

Autodesk®
Maya®

2011



Autodesk®

Getting Started

Copyright Notice

Autodesk® Maya® 2011 Software

© 2010 Autodesk, Inc. All rights reserved. Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

The following are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and other countries:

3DEC (design/logo), 3December, 3December.com, 3ds Max, Algor, Alias, Alias (swirl design/logo), AliasStudio, Alias|Wavefront (design/logo), ATC, AUGI, AutoCAD, AutoCAD Learning Assistance, AutoCAD LT, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, Autodesk, Autodesk Envision, Autodesk Intent, Autodesk Inventor, Autodesk Map, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSnap, AutoSketch, AutoTrack, Backburner, Backdraft, Built with ObjectARX (logo), Burn, Buzzsaw, CAiCE, Civil 3D, Cleaner, Cleaner Central, ClearScale, Colour Warper, Combustion, Communication Specification, Constructware, Content Explorer, Dancing Baby (image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, DesignServer, DesignStudio, Design Web Format, Discreet, DWF, DWG, DWG (logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DXF, Ecotect, Exposure, Extending the Design Team, Face Robot, FBX, Fempro, Fire, Flame, Flare, Flint, FMDesktop, Freewheel, GDX Driver, Green Building Studio, Heads-up Design, Heidi, HumanIK, IDEA Server, i-drop, ImageModeler, iMOUT, Incinerator, Inferno, Inventor, Inventor LT, Kaydara, Kaydara (design/logo), Kynapse, Kynogon, LandXplorer, Lustre, MatchMover, Maya, Mechanical Desktop, Moldflow, Moonbox, MotionBuilder, Movimento, MPA, MPA (design/logo), Moldflow Plastics Advisers, MPI, Moldflow Plastics Insight, MPX, MPX (design/logo), Moldflow Plastics Xpert, Mudbox, Multi-Master Editing, Navisworks, ObjectARX, ObjectDBX, Open Reality, Opticore, Opticore Opus, Pipeplus, PolarSnap, PortfolioWall, Powered with Autodesk Technology, Productstream, ProjectPoint, ProMaterials, RasterDWG, RealDWG, Real-time Roto, Recognize, Render Queue, Retimer, Reveal, Revit, Showcase, ShowMotion, SketchBook, Smoke, Softimage, Softimage|XSI (design/logo), Sparks, SteeringWheels, Stitcher, Stone, StudioTools, ToolClip, Topobase, Toxik, TrustedDWG, ViewCube, Visual, Visual LISP, Volo, Vtour, Wire, Wiretap, WiretapCentral, XSI, and XSI (design/logo).

ACE™, TAO™, CIAO™, and CoSMIC™ are copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright © 1993-2009, all rights reserved.

Adobe, Illustrator and Photoshop are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

Intel is a registered trademark or trademark of Intel Corporation or its subsidiaries in the United States and other countries.

mental ray is a registered trademark of mental images GmbH licensed for use by Autodesk, Inc.

OpenGL is a trademark of Silicon Graphics, Inc. in the United States and other countries. Python and the Python logo are trademarks or registered trademarks of the Python Software Foundation.

The Ravix logo is a trademark of Electric Rain, Inc.

All other brand names, product names or trademarks belong to their respective holders.

Disclaimer

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

Contents

Chapter 1	Overview	1
	Introduction	1
	About the Getting Started lessons	2
	Before you begin	3
	Installing Maya	3
	Conventions used in the lessons	4
	Using the lesson files	5
	Using the Maya Help	5
	Additional learning resources	10
	Restoring default user settings	11
Chapter 2	Maya Basics	13
	Introduction	13
	Preparing for the lessons	14
	Lesson 1: The Maya user interface	15
	Introduction	15
	Starting Maya	15
	The Maya interface	16
	Copying and setting the Maya project	25
	Saving your work	26
	Exiting Maya	27
	Beyond the lesson	27
	Lesson 2: Creating, manipulating, and viewing objects	28

Introduction	28
Creating a new scene	29
Primitive objects	30
The Toolbox: Layout shortcuts	31
The Toolbox: Transformation tools	33
The Channel Box	37
Duplicating objects	39
Save your work	40
Beyond the lesson	40
Lesson 3: Viewing the Maya 3D scene	42
Introduction	42
Camera tools	42
Workflow overview	46
Viewing objects in shaded mode	51
Grouping objects	52
The Hypergraph	53
Selection modes and masks	56
Pivot points	57
Save your work	59
Beyond the lesson	59
Lesson 4: Components and attributes	60
Introduction	60
Template display	61
Components	62
The Attribute Editor	65
Surface materials	67
Save your work	71
Beyond the lesson	72

Chapter 3 Polygonal Modeling 73

Introduction	73
Preparing for the lessons	74
Lesson 1: Modeling a polygonal mesh from a reference image	75
Introduction	75
Setting modeling preferences	76
Using 2D reference images	77
Creating a polygon primitive	80
Modeling in shaded mode	82
Model symmetry	84
Selecting components by painting	85
Selecting edge loops	86
Editing components in the orthographic views	89
Editing components in the perspective view	96
Drawing a polygon	98
Extruding polygon components	100
Bridging between edges	105

	Adding polygons to a mesh	108
	Splitting polygon faces	110
	Terminating edge loops	118
	Deleting construction history	120
	Mirror copying a mesh	122
	Working with a smoothed mesh	124
	Creasing and hardening edges on a mesh	126
	Beyond the lesson	132
	Lesson 2: Sculpting a polygon mesh	133
	Introduction	133
	Open the scene for the lesson	134
	Using Soft Select	134
	Selecting with Camera based selection	138
	Sculpting with symmetry	140
	Sculpting with Surface based falloff	144
	Selecting with Drag select	146
	Adjusting the Seam tolerance	149
	Beyond the lesson	150
Chapter 4	NURBS Modeling	153
	Introduction	153
	Preparing for the lessons	153
	Lesson 1: Revolving a curve to create a surface	154
	Introduction	154
	Creating a profile curve	155
	Creating a revolve surface	157
	Editing a revolve surface	157
	Beyond the lesson	159
	Lesson 2: Sculpting a NURBS surface	160
	Introduction	160
	Preparing a surface for sculpting	161
	Basic sculpting techniques	163
	Additional sculpting techniques	166
	Sculpting a nose	168
	Sculpting eye sockets	169
	Sculpting eyebrows	170
	Sculpting a mouth	171
	Sculpting other facial features	173
	Beyond the lesson	174
	Lesson 3: Lofting curves to create a surface	175
	Introduction	175
	Creating profile curves for a surface	176
	Duplicating curves	178
	Lofting a surface	179
	Modifying a primitive object	180
	Using the Outliner to parent objects	182

	Beyond the lesson	183
Chapter 5	Subdivision Surfaces	185
	Introduction	185
	Preparing for the lesson	185
	Lesson 1: Modeling a subdivision surface	186
	Introduction	186
	Creating a subdivision surface	187
	Splitting a surface in polygon proxy mode	189
	Extruding polygon faces	191
	Deleting polygon faces	194
	Subdivision surface levels	195
	Refining surface components	197
	Creating a crease in a subdivision surface	199
	Beyond the lesson	201
Chapter 6	Animation	203
	Introduction	203
	Preparing for the lessons	203
	Lesson 1: Keyframes and the Graph Editor	204
	Introduction	204
	Setting the playback range	205
	Setting keyframes	206
	Using the Graph Editor	209
	Changing the timing of an attribute	213
	Fine tuning an animation	214
	Deleting extra keyframes and static channels	216
	Using Playblast to playback an animation	217
	Beyond the lesson	217
	Lesson 2: Set Driven Key	219
	Introduction	219
	Lesson setup	219
	Using Set Driven Key to link attributes	220
	Viewing the results in the Graph Editor	223
	Beyond the lesson	224
	Lesson 3: Path animation	224
	Introduction	224
	Open the scene for the lesson	225
	Animating an object along a motion path	226
	Changing the timing of an object along a motion path	228
	Rotating an object along a motion path	234
	Blending keyframe and motion path animation	235
	Using Playblast to playback an animation	240
	Beyond the lesson	240
	Lesson 4: Nonlinear animation with Trax	241

	Introduction	241
	Open the first scene for the lesson	242
	Creating clips with Trax	244
	Changing the position of clips with Trax	251
	Editing the animation of clips	252
	Reusing clips within Trax	254
	Soloing and muting tracks	257
	Scaling clips within Trax	259
	Open the second scene for the lesson	260
	Creating clips from motion capture data	261
	Extending the length of motion capture data	263
	Redirecting the motion within a clip	266
	Beyond the lesson	273
	Lesson 5: Inverse kinematics	276
	Introduction	276
	Open the scene for the lesson	277
	Understanding hierarchies	278
	Viewing hierarchies using the Hypergraph	279
	Creating a skeleton hierarchy	281
	Parenting a model into a skeleton hierarchy	284
	Applying IK to a skeleton hierarchy	287
	Creating a control object for an IK system	289
	Constraining an IK system	292
	Limiting the range of motion of an IK system	296
	Simplifying the display of a hierarchy	301
	Applying parent constraints on an IK system	302
	Planning an animation for an IK system	305
	Animating an IK system	308
	Beyond the lesson	311
Chapter 7	Character Setup	313
	Introduction	313
	Preparing for the lessons	314
	Lesson 1: Skeletons and kinematics	314
	Introduction	314
	Open the scene for the lesson	315
	Creating joints	315
	Adding joints to a skeleton	322
	Creating a skeleton hierarchy	324
	Forward and inverse kinematics	325
	Posing and animating using inverse kinematics	325
	Posing and animating using forward kinematics	329
	Beyond the lesson	329
	Lesson 2: Smooth skinning	330
	Introduction	330
	Open the scene for the lesson	331

Smooth binding a skeleton	331
Skin weighting and deformations	333
Modifying skin weights	335
Influence objects	337
Beyond the lesson	340
Lesson 3: Cluster and blend shape deformers	341
Introduction	341
Open the scene for the lesson	342
Creating a target object for a blend shape	343
Creating a cluster deformer on a target object	344
Editing cluster weights	346
Creating a blend shape	350
Refining deformation effects	352
Adding target objects to an existing blend shape	353
Beyond the lesson	357

Chapter 8 Polygon Texturing 359

Introduction	359
Preparing for the lesson	360
Lesson 1: UV texture mapping	361
Introduction	361
Creating a cracker box model	362
Applying a texture map to a polygon mesh	363
Viewing UVs in the UV Texture Editor	369
Mapping UV texture coordinates	373
Working with UVs in the UV Texture Editor	377
Beyond the lesson	385
Lesson 2: UV unfolding	387
Introduction	387
Lesson Setup	388
Dividing the mesh	389
Creating a planar map	393
Unfolding a UV mesh	395
Adjusting the checker pattern	400
Outputting UVs	400
Unfolding with constraints	402
Sewing UV Edges	409
Smoothing/Relaxing a mesh interactively	414
Fixing problem areas	415
Applying textures	421
Beyond the lesson	426
Lesson 3: Normal mapping	427
Introduction	427
Lesson setup	428
Open the scene for the lesson	429
Creating a normal map	430

	Viewing a normal map	434
	Beyond the lesson	435
Chapter 9	Rendering	437
	Introduction	437
	Preparing for the lessons	439
	Lesson 1: Rendering a scene	439
	Introduction	439
	Open the scene for the lesson	440
	Creating shading materials for objects	441
	Refining shading materials for objects	445
	Maya renderers	450
	Rendering a single frame using IPR	452
	Rendering using the mental ray for Maya renderer	456
	Batch rendering a sequence of animation frames	458
	Viewing a sequence of rendered frames	461
	Beyond the lesson	462
	Lesson 2: Shading surfaces	464
	Introduction	464
	Open the scene for the lesson	465
	Assigning a shading material	466
	Using the mental ray for Maya renderer	468
	Modifying surface specularity	468
	Material types	469
	Assigning textures	471
	Using the Hypershade editor	473
	Creating a texture within the Hypershade editor	477
	Modifying a bump texture	480
	Beyond the lesson	482
	Lesson 3: Lights, shadows, and cameras	484
	Introduction	484
	Open the scene for the lesson	485
	Directional lights	486
	Spotlights	488
	Editing light attributes	491
	Shadows	494
	Creating additional cameras in a scene	495
	Animating camera moves	497
	Beyond the lesson	499
	Lesson 4: Global Illumination	501
	Introduction	501
	Open the scene for the lesson	502
	Render the scene using raytracing	503
	Render the scene using Global Illumination	508
	Beyond the Lesson	517
	Lesson 5: Caustics	519

	Introduction	519
	Open the scene for the lesson	520
	Render the scene using raytracing	521
	Render the scene using caustics	526
	Beyond the Lesson	533
Chapter 10	Dynamics	535
	Introduction	535
	Preparing for the lessons	535
	Lesson 1: Particles, emitters, and fields	536
	Introduction	536
	Creating an emitter	537
	Creating volume axis fields	539
	Adjusting the velocity of moving particles	545
	Setting the particle render type	545
	Adding dynamic attributes	546
	Adding per particle attributes	548
	Adding color to particles with a color ramp	549
	mental ray rendering particles	551
	Beyond the lesson	553
	Lesson 2: Rigid bodies and constraints	554
	Introduction	554
	Lesson setup	555
	Creating hinge constraints	557
	Running a dynamics simulation	558
	Changing an active rigid body to passive	558
	Beyond the lesson	559
Chapter 11	Painting	561
	Introduction	561
	Preparing for the lessons	562
	Lesson 1: Painting in 2D using Paint Effects	563
	Introduction	563
	Painting strokes	564
	Modifying the default brush settings	566
	Modifying the canvas	568
	Modifying the colors of a preset brush	569
	Editing strokes with tubes attributes	570
	Saving brush settings for future use	571
	Blending brushes	572
	Smearing, blurring, and erasing paint	573
	Beyond the lesson	575
	Lesson 2: Painting in 3D using Paint Effects	575
	Introduction	575
	Preparing for the lessons	577

Brushes and strokes	577
Rendering Paint Effects strokes	584
Paint Effects on 3D objects	590
Creating a surface to paint on	590
Painting on objects	594
Using turbulence with brush stroke tubes	595
Using additional preset brushes	596
Mesh brushes	599
Converting mesh strokes to polygons	602
Modifying a converted polygonal mesh	604
Beyond the lesson	608
Lesson 3: Painting textures on surfaces	610
Introduction	610
Open the scene for the lesson	610
Preparing for painting	611
Painting with an Artisan brush	613
Painting symmetrical strokes	614
Using Flood All to apply a single color	614
Brush shapes	615
Painting with a Paint Effects brush	617
Smearing and blurring	619
Painting a bump map texture	620
Beyond the lesson	622

Chapter 12 Expressions 625

Introduction	625
Preparing for the lessons	626
Lesson 1: Creating a simple expression	627
Introduction	627
Creating expressions to control a single attribute	627
Editing expressions	630
Using expressions to control multiple attributes	631
Linking multiple attributes on the same object	631
Controlling attributes in two objects	632
Beyond the lesson	632
Lesson 2: Conditional expressions	634
Introduction	634
Creating a conditional expression	634
Other conditional statement options	637
Fixing a problem in an expression	639
Using else statements	639
Simplifying expressions	640
Editing expressions to refine an animation	641
Beyond the lesson	643
Lesson 3: Controlling particle attributes	643
Introduction	643

Creating particle objects	644
Using creation expressions to set a constant color	645
Using runtime expressions	646
Modifying runtime expressions	648
Beyond the lesson	649
Chapter 13 Scripting in Maya	651
Introduction	651
Some basic concepts	652
Preparing for the lessons	655
Lesson 1: Commands in MEL	656
Introduction	656
Entering MEL commands	657
Observing script history	658
Modifying object attributes	661
Editing Objects	664
Beyond the lesson	666
Lesson 2: Saving scripts to the Shelf	666
Introduction	666
Setting up the scene	667
Recording the script history	668
Compare the rendered images	670
Saving the history as a button	671
Beyond the Lesson	674
Lesson 3: Using Variables in MEL	675
Introduction	675
Setting up the scene	675
Storing scene information	676
Create a row of barrels	678
Stacking the row of barrels	679
Using MEL built-in functions to calculate the Y offset	680
Creating dynamics with MEL commands	682
Beyond the lesson	684
Lesson 4: User interface creation and procedures	685
Introduction	685
Creating a window	685
Window naming	688
Introduction to procedures	691
Loading a script file	692
Linking the user interface	697
Saving the script	702
Using the saved script file	703
Beyond the lesson	705
Lesson 5: Using Python in Maya	705
Introduction	705
Entering Python commands	706

	Using flags in Python	709
	Using the edit flag in Python	713
	Communicating between Python and MEL	715
	Beyond the lesson	717
Chapter 14	Assets	719
	Introduction	719
	Preparing for the lessons	719
	Lesson 1: Setting up an asset	720
	Introduction	720
	Lesson setup	720
	Creating an asset	721
	Publishing attributes	722
	Publishing multiple attributes to a single published name	725
	Open the second scene for the lesson	728
	Creating a template	728
	Creating Views	730
	Creating a binding set	733
	Assigning a custom icon	734
	Setting Black Box mode	734
	Beyond the lesson	735
	Lesson 2: Using assets in a scene	736
	Introduction	736
	Lesson setup	736
	Importing assets to dress a scene	737
	Assigning an existing template to an asset	738
	Autobinding attributes	740
	Manually binding attributes	741
	Swapping assets	742
	Referencing assets in a scene	744
	Creating a proxy asset	747
	Modifying a proxy asset	749
	Dressing the rest of the scene with assets	751
	Beyond the lesson	753
Chapter 15	File Referencing	755
	Introduction	755
	Preparing for the lessons	755
	Lesson 1: Using file references	756
	Introduction	756
	Lesson setup	756
	Creating a file reference	757
	Using the Reference Editor	758
	Exporting objects in your scene	761
	Creating a proxy for a referenced object	761

	Creating a proxy for a referenced asset	763
	Creating proxy references	768
	Tweaking the zombie's walk cycle	769
	Exporting reference tweaks	773
	Applying reference tweaks	775
	Storing animation as reference edits	777
	Applying animation as reference edits	780
	Beyond the Lesson	782
Chapter 16	Hair	783
	Introduction	783
	About hair simulation	785
	Preparing for the lessons	785
	Lesson 1: Creating a basic hairstyle	786
	Introduction	786
	Lesson setup	787
	Creating hair on a surface	788
	Styling the hair	792
	Setting up hair collisions	797
	Rendering the hair	801
	Modifying hair attributes	802
	Setting up shadowing on hair	804
	Beyond the lesson	806
	Lesson 2: Creating a dynamic non-hair simulation	808
	Introduction	808
	Lesson setup	809
	Setting up the curtain scene	809
	Making the hair collide with another object	812
	Assigning a Paint Effects brush to the hair	813
	Setting up constraints	814
	Rendering the curtain scene	817
	Beyond the lesson	819
Chapter 17	Fluid Effects	821
	Introduction	821
	Preparing for the lessons	822
	Lesson 1: Creating a dynamic 2D fluid effect	823
	Introduction	823
	Creating a two-dimensional fluid container	824
	Adding a fluid emitter to a container	824
	Changing the behavior of a fluid	826
	Combining colors in a fluid	829
	Colliding a fluid with an object	831
	Beyond the lesson	832
	Lesson 2: Creating a non-dynamic 3D fluid effect	833

Introduction	833
Creating a 3D fluid container	834
Adding fluid to a container	835
Defining shader attributes for a fluid	836
Texturing the contents of a fluid container	836
Adding self shadowing to texture density	840
Beyond the lesson	841
Lesson 3: Creating a dynamic 3D effect	842
Introduction	842
Creating a 3D container	843
Painting Fuel and Density into a container	843
Painting Temperature values into a container	848
Adding color to Density and Temperature	849
Beyond the lesson	851
Lesson 4: Creating an ocean effect	852
Introduction	852
Creating an ocean plane and shader	853
Adding a preview plane to an ocean	853
Modifying ocean attributes	854
Floating objects	857
Beyond the lesson	859
Chapter 18 Fur	861
Introduction	861
Preparing for the lessons	862
Lesson 1: Assigning a fur description	862
Introduction	862
Lesson setup	863
Duplicating objects across an axis of symmetry	863
Renaming surfaces on a model	865
Assigning objects to a reference layer	866
Assigning a fur description preset to a model	868
Reversing surface normals	870
Modifying the fur direction	872
Painting fur attributes	875
Modifying the color of a fur description	882
Creating a new fur description	883
Beyond the lesson	885
Lesson 2: Rendering fur	887
Introduction	887
Lesson setup	887
Creating lights in a scene	888
Adding mental ray shadow maps to lights	890
Rendering the scene	892
Beyond the lesson	892

Chapter 19	nCloth	895
	Introduction	895
	Preparing for the lessons	896
	Lesson 1: Creating nCloth collisions	897
	Introduction	897
	Lesson setup	897
	Creating an nCloth object	898
	Making an nCloth collide with its environment	900
	Adjusting the accuracy of nCloth collisions	902
	Beyond the lesson	911
	Lesson 2: Creating nCloth constraints	912
	Introduction	912
	Lesson setup	912
	Constraining an nCloth to a passive object	913
	Changing which nCloth points are constrained	917
	Making nCloth flap in dynamic wind	923
	Beyond the lesson	926
	Lesson 3: Creating nCloth Clothing	926
	Introduction	926
	Lesson setup	927
	Making the dress into an nCloth object	927
	Making the character wear the dress	928
	Caching nCloth to speed up playback	929
	Adjusting the fit of the dress	930
	Defining the behavior of nCloth clothing	931
	Painting nCloth properties	933
	Open the second scene for the lesson	936
	Setting the initial state	936
	Constraining nCloth clothing	938
	Improving the quality of the nCloth simulation	940
	Smoothing nCloth clothing	943
	Beyond the lesson	944
Chapter 20	nParticles	945
	Introduction	945
	Preparing for the tutorials	946
	Lesson 1: Creating nParticles	947
	Introduction	947
	Lesson setup	948
	Creating an nParticle system	948
	Making nParticles collide with their environment	950
	Setting Nucleus Space Scale	952
	Adjusting nParticle size and color	953
	Adjusting nParticle collision attributes	955
	Beyond the lesson	958

Lesson 2: Creating a smoke simulation with nParticles	959
Introduction	959
Lesson setup	959
Creating an nParticle system	960
Editing nParticle Lifespan and Radius	964
Open the second scene for the lesson	968
Adding a Volume Axis field	968
Adjusting nParticle velocity	971
Adjusting nParticle Shading attributes	972
Open the third scene for the lesson	976
Fine tuning your smoke effect	977
Beyond the lesson	979
Lesson 3: Creating a liquid simulation with nParticles	981
Introduction	981
Lesson setup	982
Creating a Water style nParticle object	982
Adjusting Liquid Simulation attributes	986
Adding fluidity to the nParticles	988
Open the second scene for the lesson	990
Convert nParticles to a polygon mesh	990
Cache your nParticle simulation	994
Adding surface tension	996
Adding Motion Streak	998
Open the third scene for the lesson	1001
Render your liquid simulation	1001
Assigning material shaders	1002
Rendering a simulated frame	1005
Beyond the lesson	1006
Index	1009

Overview

1

Introduction

Welcome to Autodesk® Maya®, one of the world's leading software applications for 3D digital animation and visual effects. Maya provides a comprehensive suite of tools for your 3D content creation work ranging from modeling, animation, and dynamics through to painting and rendering to name but a few.

With Maya, you can create and edit 3D models in a variety of modeling formats and animate your models using Maya's suite of animation tools. Maya also provides a range of tools to allow you to render your animated 3D scenes to achieve photo realistic imagery and animated visual effects.

You can create convincing visual simulations using Maya dynamics and nDynamics tools. Using Maya® Fluid Effects™, you can simulate and render viscous fluids, atmospheric, pyrotechnic, and ocean effects. Maya® nCloth™ lets you create simulations of fabric and clothing, while Maya® nParticles™ can be used to simulate a wide range of effects including liquids, clouds, smoke, spray, and dust. Other Maya dynamic simulation tools include Maya® Fur™, Maya® Hair™, and Maya® Artisan™ brush tools.

The Maya software interface is fully customizable for those users who require the ability to maximize their productivity. Maya allows users to extend their functionality within Maya by providing access to MEL™ (Maya Embedded Language). With MEL, you can customize the user interface and write scripts and macros. In addition, a full Application Programmers Interface (API) is available to enhance the power and functionality of Maya. Maya also provides a Python-based Maya API for those users wishing to use it.

The content creation power of Maya is provided to users in an integrated software application that is designed to enhance user productivity and ease of use.

This section provides the following information:

- About the Getting Started lessons—Information about the 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 lessons.
- Installing Maya—Information on installing Maya.
- Using the lesson files—How to access and use the lesson files for the Getting Started with Maya lessons.
- Conventions used in the lessons—Describes the various conventions used throughout the Getting Started with Maya lessons.
- Using the Maya Help—Outlines the various help resources provided with your Maya software.
- Additional learning resources—Outlines learning resources beyond what is included with your Maya software.
- Restoring default user settings—Describes how to reset Maya to its default settings before you begin the lessons.

About the Getting Started lessons

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

If you are new to Maya, this guide gets you started on your learning path. If you are an existing user or are transitioning from another 3D software application, *Getting Started with Maya* provides a starting point for understanding features you haven't yet had time to learn.

Getting Started with Maya 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* 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.

We recommend that any new Maya 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) which introduce many fundamental concepts and skills related to the Maya user interface.

The version of Getting Started with Maya within the Maya Help also contains Apple® QuickTime® 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* lessons.
- 2 Click the tutorial you want to work through.
The Maya Help displays the associated lessons for that tutorial.

Before you begin

Before beginning Getting Started with Maya, you should have a working knowledge of your computer's operating system. You should know how to use a mouse, select menus, and enter text and commands from your keyboard. You should also know how to open and save files, copy files from a DVD to your hard drive, and be able to navigate your computer operating system's file browser.

If you require an overview or review of these 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 might 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

You must have Maya installed and licensed on your computer system to successfully complete the lessons in this guide. To operate Maya 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 check the *Maya Features and Specification* link at <http://www.autodesk.com/maya>.

Conventions used in the lessons

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

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

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

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

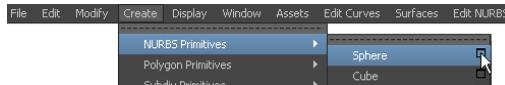
When instructed to select a menu within Maya we use the following convention:

- 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 lessons have accompanying lesson files that were created for use with the lessons. These files are included in the `GettingStarted` directory that was installed with your Maya software and can be found in the following locations:

(Windows XP, Windows Vista, and Windows 7, 32 and 64-bit)

```
<drive>:\Program Files\Autodesk\Maya2011\GettingStarted
```

(Mac OS X)

```
/Applications/Autodesk/maya2011/GettingStarted
```

(Linux 64-bit)

```
/usr/autodesk/maya2011-x64/GettingStarted
```

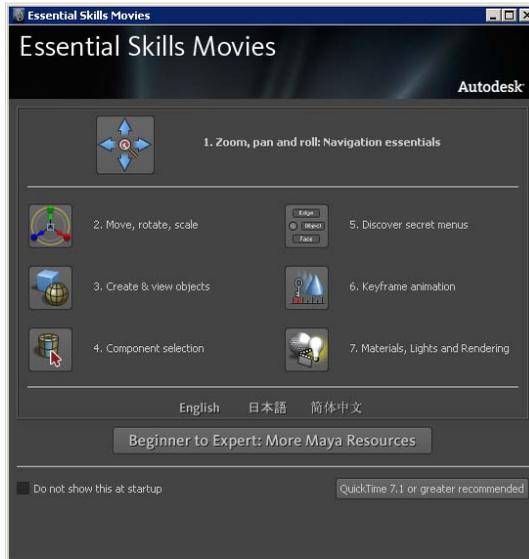
Before you use the lesson files, you need to copy the `GettingStarted` directory to your Maya `projects` folder. Then, you need to set the `GettingStarted` folder as your project directory in Maya. For more information and steps, see [Copying and setting the Maya project](#) on page 25.

Using the Maya Help

Your Maya software application comes with a comprehensive set of documentation resources.

Essential Skills Movies

The first time you start Maya, the Essential Skills Movies window displays in Maya. Watch these movies to learn about the skills you need to master when you first use Maya. The audio for the movies is available in English, French, German, Italian, Japanese, Korean, and Mandarin.



To play the Essential Skills Movies

- 1 In the Essential Skills Movies window, click the buttons to play a movie. Your computer launches the necessary multimedia player and your chosen movie begins to play.
- 2 Click your multimedia player's controls to start, stop, and pause the movie.

To close the Essential Skills Movies window or the multimedia player

- 1 To close the Essential Skills Movies window, click the close box in the upper right corner of the window.
If you do not want to have this dialog box automatically display when you start Maya, turn on the *Don't show this at startup* check box.
- 2 To close the multimedia player, select File > Exit or click the close box in the upper right corner of the window. (This instruction might vary depending upon which multimedia player is used)

If you want to watch the movies in the future

- In Maya, select Help > Learning Movies.
The Learning Movies window appears.

Maya Help

Your Maya software application comes installed with Maya technical documentation that assists you in learning the Maya software. The Maya Help is HTML-based, structured by module, fully searchable, and is displayed using your computer's web browser.

The Maya Help is topic based and displays the major functionality categories for Maya. The Maya Help can assist you in finding reference information about particular topics, how to perform specific tasks, and MEL™ command references.

To launch the Maya Help

- ▶ Select Help > Maya Help.

The Maya Help appears in a separate web browser window (depending on your user preference settings). The left hand pane of the Maya Help lets you navigate to various Maya topics.

To obtain help on a particular Maya topic

- ▶ In the Maya Help navigation pane, click the name of the Maya topic you want information about (for example, Modeling, Animation, Dynamics, MEL commands, and so on).

The Maya Help displays the associated sub-topics and categories associated with the name you selected.

Maya Index and search

You can search the Maya Help directly using the index and search capabilities. With these tools you find the Maya topic you're looking for by searching the topic word in an alphabetic list or by directly typing the topic word(s) into the search field and having the search tool find the documentation entries associated with it.

To use the Maya Index

- 1 In Maya, select Help > Maya Help.

The Maya Help appears in a separate window (depending on your user preferences). The Maya Index button appears at the top of the left navigation pane.

- 2 Click the Index button.

The navigation pane updates to display an alphabetic list at the top of the pane with the first index items listed.

- 3 Click an item/letter in the alphabetic list.

The information related to that topic appears in the right pane.

To use the Maya Help search

- 1 Select Help > Maya Help.

The Maya Help appears in a separate window (depending on your user preferences). The Maya Search button appears at the top of the left navigation pane.

- 2 Click the Search button.

The navigation pane updates to display the available search methods and options. You search a topic by typing in a word(s) that best represent the information you require.

- 3 In the text box, type a word that best represents your search topic.

By default, all of the content in the Maya Help is searched. You can narrow the search results by selecting a specific user guide from the drop down list below the text box.

- 4 Click the Search button to begin.

The search results appear in the left navigation pane, in order of relevancy.

- 5 Click the desired topic from the search results list.

The information related to that entry appears in the right pane of the Maya Help.

Popup Help

Popup Help provides you with a quick method of identifying a particular tool or icon in the Maya user interface.



To use Popup Help

- Move your mouse cursor over an icon or button.
The name or description of it appears in a popup window directly over it.

To turn on the Popup Help if it does not appear

- 1 If you're operating Maya on a Windows or Linux operating system:
 - Select Window > Settings/Preferences > Preferences.
 - In the Preferences window, click the Help category and set the Tooltips box to Enable in the Popup Help section so a check mark appears.
 - Click the Save button to close the Preferences window.
- 2 If you're operating Maya on a Mac OS X operating system:
 - Select Maya > Preferences.
 - In the Preferences window, click the Help category and click the Tooltips Enable box in the Popup Help section so a check mark appears.
 - Click the Save button to close the Preferences window.

Help Line

The Help Line at the bottom of Maya's window shows information about tools, menus, and objects. Like the Popup help, it displays descriptions when you move the mouse over icons and menu items. It also displays instructions when you select a tool. This is useful if you don't know or forget how to use a particular tool.

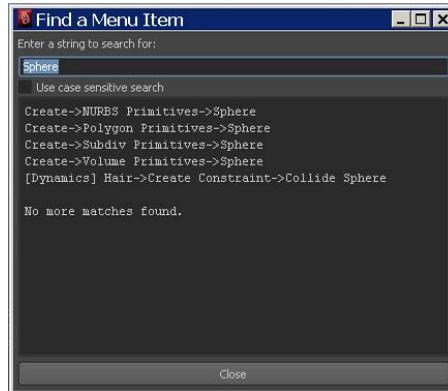


To use the Help Line

- Move your mouse cursor over an icon or button.
The icon or button name and instructions about how to use that tool appear in the Help Line.

Find Menu

The Find Menu feature lets you find the location of a particular menu item. Find Menu works only on the main menu items.



To find the location of a main menu item

- 1 Select Help > Find Menu.
The Find a Menu Item window appears.
- 2 Type the menu item you want to locate in the text field, and press Enter (Windows and Linux) or Return (Mac OS X).
The possible locations for the main menu item display in the results field of the Find Menu window.

Additional learning resources

Beyond the Maya Help resources within your Maya software, you can access the following resources to learn more about Maya or obtain technical assistance.

The Maya Learning Path

Discover the many learning resources available from Autodesk using the Maya Learning Path. For more information, see www.autodesk.com/maya-learningpath.

The Maya Web site

The Maya Web site contains a wealth of resources related to your Maya software and many other related products and services. You can view the Maya Web site at www.autodesk.com/maya using your web browser.

Autodesk Training

Autodesk provides a range of products and services to help you get the most from your Maya software. You can purchase additional self-paced learning materials or attend certified instructor led training courses at Autodesk sanctioned training facilities. For more information, see www.autodesk.com/maya-training.

Technical Support

Autodesk delivers technical support services for Maya globally through telephone and email services, as well as online eSupport services. For more information, click the *Support Center* link from the Maya Help menu or click the *Services and Support* link on the Maya Web site.

Events and Seminars

Autodesk also runs Maya seminars and short format training courses at major computing events and trade shows. For more information, see www.autodesk.com/maya-events-seminars.

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 different name; for example, *myprefs*. The *prefs* directory path is:

Windows

(Windows XP)

```
\Documents and Settings\\My  
Documents\maya\2011\en_US\prefs
```

(Windows XP 64bit)

```
\Documents and Settings\\My  
Documents\maya\2011-x64\en_US\prefs
```

(Windows Vista and Windows 7)

```
\Users\\Documents\maya\2011\en_US\prefs
```

(Windows Vista and Windows 7, 64-bit)

```
\Users\\Documents\maya\2011-x64\en_US\prefs
```

Mac OS X

```
/Users/<username>/Library/Preferences/Autodesk/maya/en_US/2011/prefs
```

Linux (64-bit)

```
-<username>/maya/2011-x64/prefs
```

NOTE If you are running the Japanese version of Maya, change *en_US* in the above directory paths to *ja_JP*.

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*.

Maya Basics

2

Introduction



Image created by: David Atkinson

Critical to learning any software application is some initial understanding of the basic concepts: how that software's world works and the fundamental skills you need to work in that world. If you have never used a three dimensional (3D) software application before, you may initially find Maya different compared to 2D applications.

If you are wondering "where do I begin?", this chapter is the best place to start. We recommend that you complete the lessons in this chapter so the essential concepts and skills presented become familiar to you.

This chapter covers some of the fundamental concepts and skills for Maya in four lessons:

- Lesson 1 The Maya user interface: [Introduction](#) on page 15
- Lesson 2 Creating, manipulating, and viewing objects: [Introduction](#) on page 28
- Lesson 3 Viewing the Maya 3D scene: [Introduction](#) on page 42
- Lesson 4 Components and attributes: [Introduction](#) on page 60

Preparing for the lessons

To ensure the lessons work as described:

- Ensure Maya is installed and licensed on your computer.
If you have not installed Maya yet, refer to the *Installation and Licensing* manual that accompanies your Maya software package. It outlines the requirements for installing Maya and procedures for installation and licensing Maya on supported hardware platforms.
- If you have never started Maya on your computer before, it will start for the first time using the default preference settings.
- If you have run Maya before, you should ensure that your Maya user preferences are reset to their default setting. This ensures that the lessons appear and work as described.
Refer to [Restoring default user settings](#) on page 11 for instructions on resetting user preferences to the default setting.
- Unless otherwise indicated, the directions in this chapter for making menu selections assume you're working from the Polygons menu set.

NOTE Before you perform the lessons in this book, 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 a check mark does not appear beside these menu items.

Lesson 1: The Maya user interface

Introduction

Just as the driver of an automobile is familiar with the dashboard of their vehicle, it is important for you to become familiar with the Maya “dashboard.”

The Maya *user interface* refers to everything that the Maya user sees and operates within Maya. The menus, icons, scene views, windows, and panels comprise the user interface.

Through the Maya user interface you access the features and operate the tools and editors that allow you to create, animate, and render your three dimensional objects, scenes, and effects within Maya.

As you spend time learning and working with Maya, your knowledge of and familiarity with the user interface will increase until it becomes second nature.

In this lesson you learn how to:

- Start Maya on your computer.
- Use the Maya interface so that you can begin to understand where and how to access the critical tools to get started with Maya.
- Select the menu and icon sets within Maya.
- Learn the names of tools related to the icons in Maya.
- Create a new scene view.

This first lesson contains additional explanations of the tools and concepts compared to many of the lessons later in this manual. We suggest you take some time to review these explanations as they lay the foundation for understanding where things are in Maya.

Starting Maya

To start Maya on Windows

- Do one of the following:
 - Double-click the Maya icon on your desktop.

- From the Windows Start menu, select All Programs > Autodesk > Autodesk Maya 2011 > Maya 2011.

To start Maya on Mac OS X

- Do one of the following:
 - Double-click the Maya icon on your desktop.
 - Click the Maya icon in the Dock.
 - From the Apple Finder menu, select Go > Applications and then browse for the Maya icon and double-click it to start Maya.

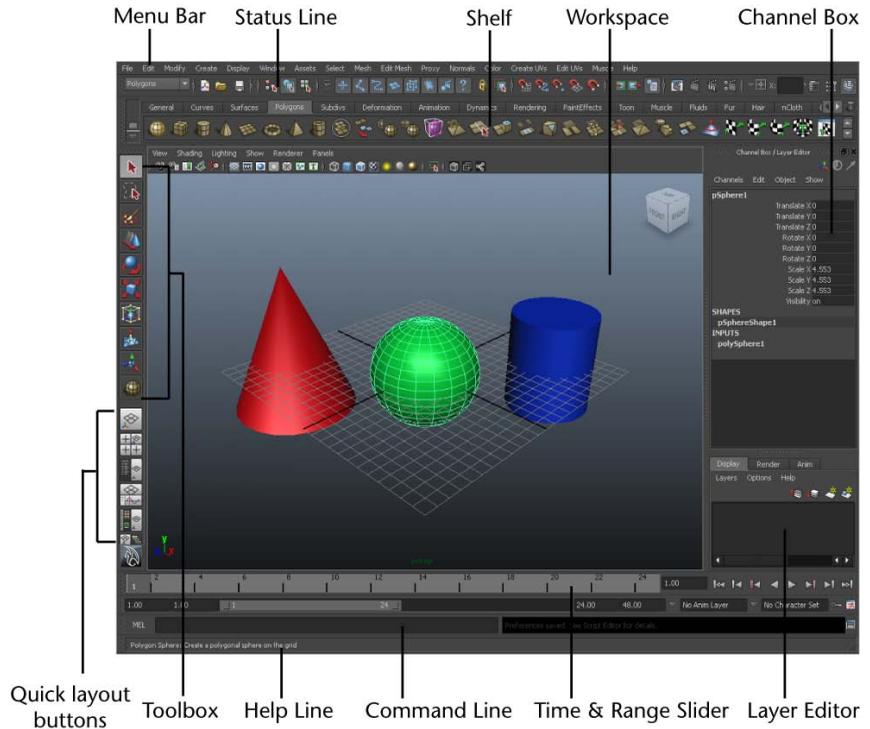
To start Maya on Linux

- Do one of the following:
 - Double-click the Maya icon on your desktop.
 - In a shell window, type: `maya`.

The Maya interface

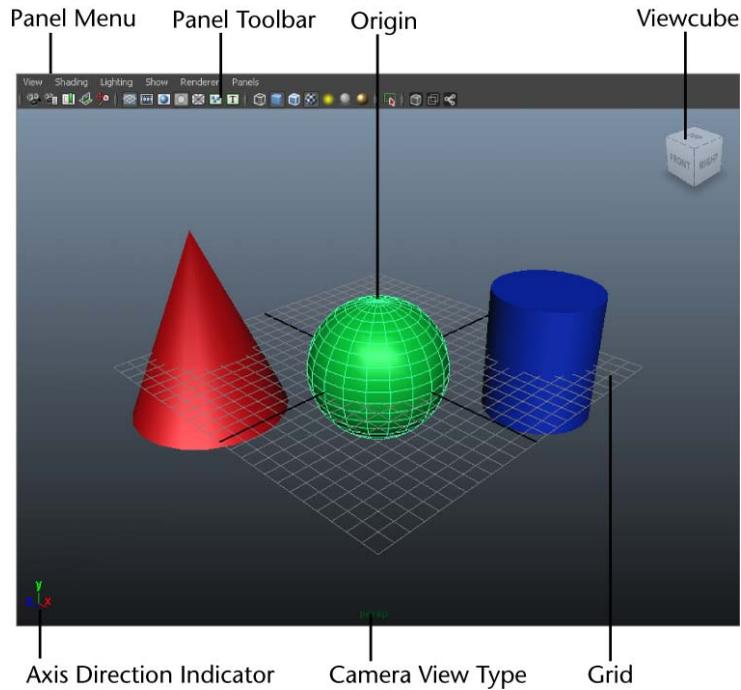
Now that Maya is running, you first need to understand what you are seeing. There are a lot of items displayed in the Maya user interface.

The best way to begin is to learn the fundamental tools and then learn additional tools as you need them. Begin by learning some of the main tools.



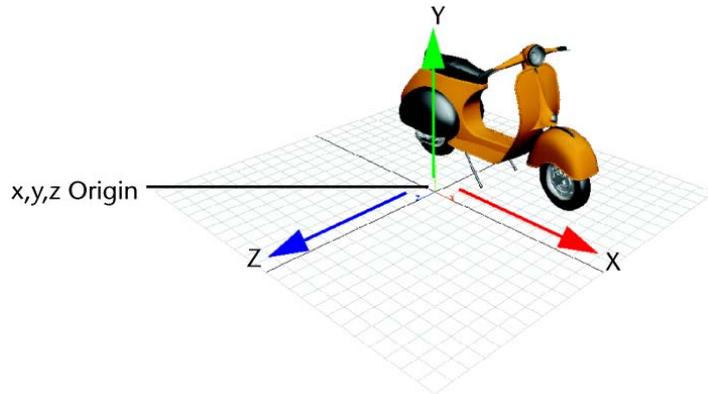
The Maya workspace

The Maya workspace is where you conduct most of your work within Maya. The workspace is the central window where your objects and most editor panels appear.



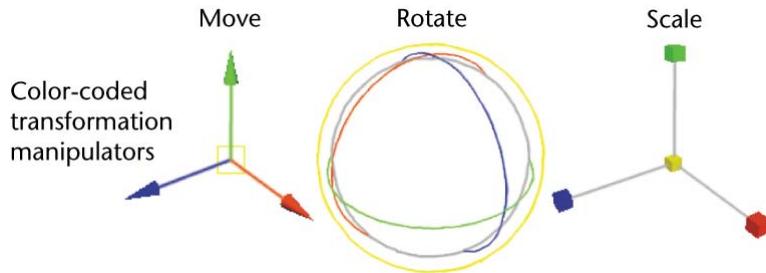
When you start Maya for the first time, the workspace displays by default in a perspective window, or *panel*. There are the other components of the default perspective view panel:

- The panel is labeled *persp* at the bottom to indicate that you are viewing the Maya scene from a perspective camera view.
- The panel has its own menu bar at the top left corner of the panel. These menus allow you to access tools and functions related to that specific panel.
- The grid is displayed with two heavy lines intersecting at the center of the Maya scene. This central location is called the *origin*. The origin is the center of Maya's 3D world, and with all object's directional values measured from this location.



In Maya, like many other 3D applications, the three dimensions are labeled as the X, Y, and Z axes. The origin is located at X, Y, Z position of 0, 0, 0. The grid also lies along the X, Z plane. We refer to this as a *plane* because you might visualize an imaginary, flat, two-dimensional square laying along this 3D position.

Maya labels the X, Y, and Z axes with a color scheme: red for X, green for Y, and blue for Z. Many tools that you use in Maya use this color scheme to indicate that you are accessing a particular item that relates to X, Y, and Z in some way.



The axis indicator shows in which direction, X, Y, or Z, you are viewing the Maya scene. The axis indicator is color coded in the red, green, and blue color scheme and appears in the lower left corner of a view panel.

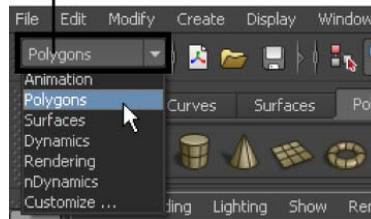
This is extremely useful if you are new to 3D, as many of the instructions in this manual and the Maya Help assume you know where you are viewing the scene in relation to the X, Y, Z axes.

Main Menu bar

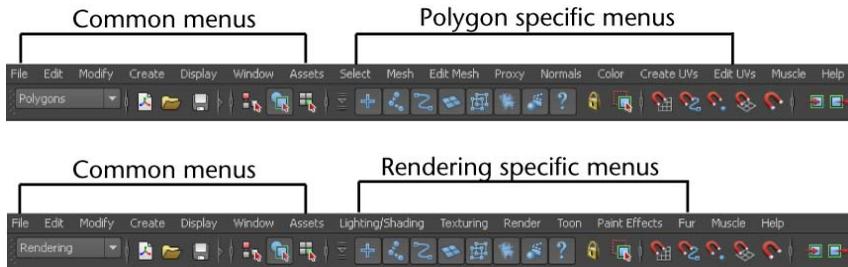
Tools and items are accessible from pull down menus located at the top of the user interface. In Maya, menus are grouped into menu sets. These menu sets are accessible from the Main Menu bar.

The Main Menu bar appears at the top of the Maya interface directly below the Maya title bar and displays the chosen menu set. Each menu set corresponds to a *module* within Maya: Animation, Polygons, Surfaces, Rendering, and Dynamics. Modules are a method for grouping related features and tools.

menu selector



You switch between menu sets by choosing the appropriate module from the menu selector on the Status Line (located directly below the File and Edit menus). As you switch between menu sets, the right-hand portion of the menus change, but the left-hand portion remains the same; the left-hand menu are common menus to all menu sets. The left-hand menus contain File, Edit, Modify, Create, Display, and Window.



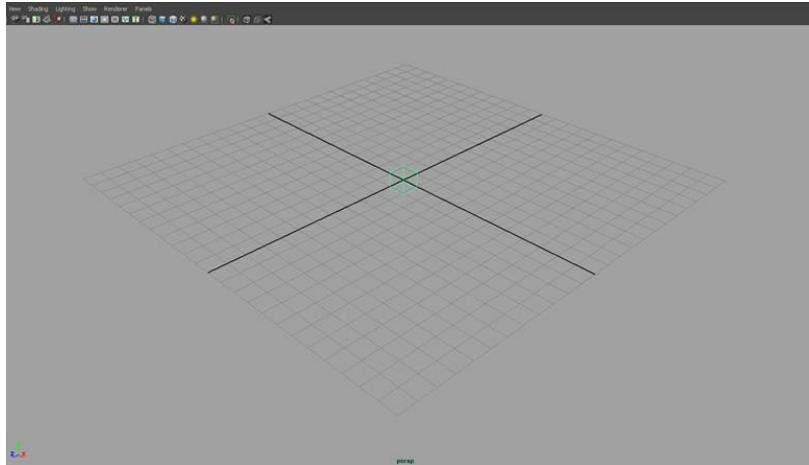
To select a specific menu set

- 1 On the Status line, select Animation from the drop-down menu.
The Main Menu changes to display the menu set that relates to the Animation module. In particular, menu titles such as Animate, Deform, Skeleton, Skin, and so on, appear.

- 2 Using the menu selector, choose Polygons from the drop-down menu. The main menu changes to display the menu set for Polygons. Menu titles such as Select, Mesh, Edit Mesh, and so on, appear. For now, leave the menu set at Polygons. You will use this set in the next step.

To create a primitive 3D object from the Polygons menu set

- 1 Select Create > Polygon Primitives > Interactive Creation and ensure that a check mark does not appear beside this item. For this lesson, you won't use this option.
- 2 From the Main Menu Bar, select Create > Polygon Primitives > Cube >. Maya creates a 3D cube primitive object and places it at the center (origin) of the Maya workspace.



Status Line

The Status Line, located directly below the Main Menu bar, contains a variety of items, most of which are used while modeling or working with objects within Maya. Many of the Status Line items are represented by a graphical icon. The icons save space in the Maya interface and allow for quick access to tools used most often.

In this lesson, you learn about some of the Status Line areas.



You've already learned the first item on the Status line: the Menu Selector used to select between menu sets.

The second group of circled icons relate to the scene and are used to create, open, and save your Maya scenes.

The third and fourth group of buttons are used to control how you can select objects and components of objects. You will learn more about selection of objects in later lessons.

The fifth group of icons are used to control the Snap Mode for objects and components. You will begin to use these tools in a later lesson in this chapter.

The last section comprise three buttons that are used to show or hide editors, including the Attribute Editor, Channel Box, Layer Editor, and Tool Settings. The default display shows the Channel Box and the Layer Editor. When you create an object, like the cube for example, information about that object displays in these editors. You will learn how to use these editors later in this chapter.

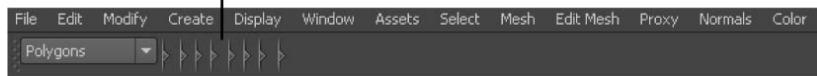


For better organization on the Status Line, all of the icon buttons are broken into groups that you can expand and collapse, as shown.

Collapse icon groups by clicking on vertical bars



Expand icon groups by clicking on vertical arrow bars



Shelf

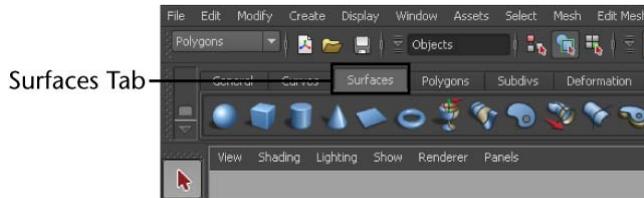
The Shelf is located directly below the Status line. The Maya Shelf is useful for storing tools and items that you use frequently or have customized for your

own use. You can keep the tools and items you use most frequently in a location that provides handy access. Maya has some of the Shelf items pre-configured for your use.



To create an object using a tool from the Shelf

- 1 From the Shelf, select the Surfaces tab in order to view the tools located on that shelf.



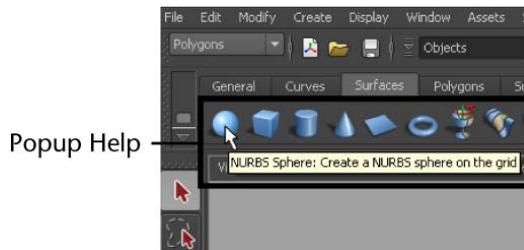
- 2 Select Create > NURBS Primitives > Interactive Creation to ensure that a check mark does not appear beside the item.

For this lesson, you won't use this option

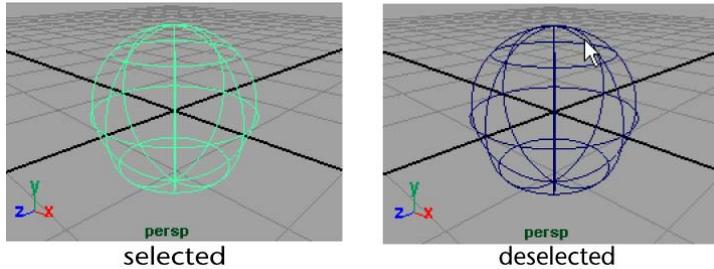
- 3 From the Shelf, select the NURBS sphere icon located at the left end by clicking on it.

Maya creates a sphere primitive object and places it at the center of the Maya workspace in the same position as the cube.

TIP You can determine if this is the correct tool prior to choosing it by first placing your mouse cursor over the icon, the name or description of it appears in a popup window directly over it.

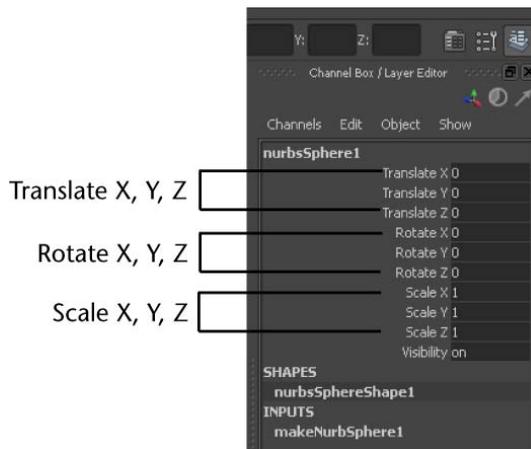


In your scene view the wireframe outline of the cube you created earlier in the lesson has changed color to navy blue, and the sphere is displayed in a bright green color. The sphere is now the *selected* object and the cube is no longer selected. In Maya, when the object displays like this, we refer to it as being *selected* or *active*.



Selection of objects and components is a way of indicating to Maya that this particular item is to be affected by the tool or action you will subsequently choose. As you work with Maya, you will be selecting and deselecting items a lot. You will learn how to select and deselect objects later in this chapter.

Some numerical information appears in the *Channel Box* editor on the right hand side of the user interface. This information relates to X, Y, and Z, translation, rotation, and scaling for the active object. The X, Y, and Z Translate numerical values are currently set to 0. This indicates that the sphere's location is at the origin. The Channel Box is useful for viewing and editing this type of basic information. You will use the Channel Box later in this chapter.



To hide or show the Channel Box

- 1 To hide the Channel Box, click the Show/Hide Channel Box icon from the right end of the Status line.

The Channel Box disappears, and the perspective scene view expands slightly. With the Channel Box hidden, you have more working area in your scene view.



- 2 To show the Channel Box, click the Show/Hide Channel Box icon on the Status line. The Channel Box appears in the scene view.

Copying and setting the Maya project

Before you can save your work, you need to set a project in Maya.

A project is a file directory that stores and organizes all of the files (scenes, images, materials, textures, etc.) related to a particular scene. In Maya, you create and work with a variety of file types and formats. The project directory allows you to keep these different file types in their unique subdirectory locations within the project directory.

You can create a new project directory by choosing File > Project > New. Alternatively, you can choose an existing folder to act as the project directory if you have prepared a folder in advance.

For these lessons, the project directory will be the `GettingStarted` folder that we have created for you. Here, you will find all the template files that you need for the lessons, and you will save your work as you learn. You need to copy the `GettingStarted` folder to your `Maya projects` folder, and then you need to set the `GettingStarted` folder as the project directory in Maya.

To copy and set the `GettingStarted` folder as your Maya project

- 1 Navigate to the `GettingStarted` folder that was installed with your Maya software. The location of this folder depends on the operating system that you are using.

(Windows XP, Windows Vista and Windows 7, 32 or 64-bit)

```
<drive>:\Program Files\Autodesk\Maya2011\GettingStarted
```

(Mac OS X)

```
/Applications/Autodesk/maya2011/GettingStarted
```

(Linux 64-bit)

```
/usr/autodesk/maya2011-x64/GettingStarted
```

- 2 Copy the `GettingStarted` folder.
- 3 Paste the `GettingStarted` folder in one of the following locations, depending on the operating system that you are using:
(Windows XP, 32 or 64-bit)

```
<drive>:\Documents and Settings\\My  
Documents\maya\projects
```

(Windows Vista or Windows 7, 32 or 64-bit)

```
<drive>:\Users\\My Documents\maya\projects
```

(Mac OS X)

```
/Users/<username>/Documents/maya/projects
```

(Linux 64-bit)

```
<home directory>/maya/projects
```

- 4 In Maya, choose **File > Project > Set**.
- 5 Browse for the `GettingStarted` folder that you pasted in your Maya `projects` directory in step 3.
- 6 Select the folder and click **OK**.

Saving your work

Make it a habit to save your work often when working on your Maya projects. In that way, you can always open an earlier version of your work should you make a mistake.

Maya refers to everything you've created in your workspace as the *scene*. This includes any objects, lights, cameras and materials associated with your working session.

To save your Maya scene

- 1 Select **File > Save Scene**.
A file browser appears, listing the `GettingStarted` project directory where you can save your scene.

If the `GettingStarted` directory does not appear, you need to copy the `GettingStarted` folder and set it as your Maya project. See [Copying and setting the Maya project](#) on page 25.

- 2 Within the `GettingStarted` directory, double-click the `scenes` folder to open it.
- 3 Type: `Lesson1` in the file name text box.
- 4 Click Save.

Maya saves your file to the `scenes` directory within your `GettingStarted` project directory. Maya automatically saves the file with a `.mb` file extension. The `.mb` file extension indicates that the scene was saved as a Maya binary file: the default file type for a Maya scene.

Exiting Maya

Before you exit Maya, ensure you save any work that you want to retrieve and continue with at a later time.

To exit Maya

- ▶ Select File > Exit from the main menu.
Maya does one of the two following actions:
 - If you have saved your scene immediately preceding the Exit command, Maya exits.
 - If you have not recently saved your scene, a message prompt appears on the screen asking if you want to save your changes. Click either Save, Don't Save, or Cancel.

Beyond the lesson

In this lesson you began your orientation to Maya by learning:

- How to start Maya on your computer.
- The Maya workspace, and how it shows three dimensional space (X, Y, Z).
- How Maya color-codes items and tools related to X, Y, Z.

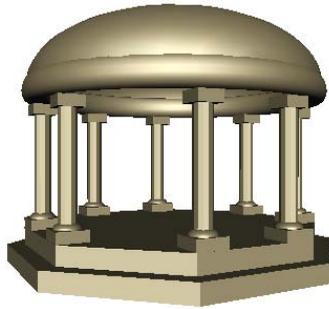
- The location of the main menus for the various modules within Maya.
- How to create a three-dimensional object from the Polygons menu.
- The location of the Status Line and how items are displayed as icons.
- About the Shelf how to create a three dimensional object from the Shelf.
- How to hide and show the Channel Box and that basic transformation, scaling, and rotational information for an object can viewed in the Channel Box.
- How to save your work.
- How to exit Maya.

As you proceed through Getting Started with Maya you should be familiar with the fundamental concepts and skills covered in this first chapter.

If you want to learn more about a particular tool or feature that has been presented in this lesson, refer to the Maya Help.

Lesson 2: Creating, manipulating, and viewing objects

Introduction



Using primitive objects to model 3D forms is a great place to continue learning about Maya. You can create many types of 3D objects using Maya and then move, scale, and rotate them to create more complex forms in your scene.

In this lesson, you begin to construct a classic temple using the primitive object creation tools in Maya. The project is not very complex and provides you with experience in using some of the important object manipulation and viewing tools.

As you continue to work with Maya, you'll learn how to visualize more complex forms using these basic objects. Maya has many advanced tools and options for modeling complex forms, as you will learn in later chapters.

In this lesson, you learn how to:

- Create 3D primitive objects.
- Select objects for manipulation and editing purposes.
- Move and rotate objects using your mouse.
- Move, rotate, and scale objects using numeric input.
- Duplicate objects.
- Change the viewing panels in Maya using a variety of methods so you can view your objects from different points of view.
- Undo actions when you need to undo a particular task or step.

Creating a new scene

You begin your temple project by creating a new empty scene.

To create a new scene

- 1 Start Maya (if it is not already running).
When Maya starts, it automatically creates a new scene.
If Maya was previously running, follow steps 2 and 3.
- 2 From the main menu, select File > New Scene.
Maya displays the following prompt.



- 3 Click No.

Maya creates a new scene and delete everything that was in the previous scene.

Primitive objects

Maya provides many types of primitive types and shapes such as cubes, spheres, cylinders, and planes.



Primitive objects can be used as a starting point for a wide variety of shapes and forms. The most common workflow when using primitive objects is:

- Set the construction options for the primitive when you initially create it so that it appears in the Maya scene roughly in the size and shape that you require.
- Move, scale, and rotate the primitive object into its final position either by direct manipulation (the move, scale, and rotate tools), or by entering numeric values through an editor.
- Duplicate the primitive objects to create multiple copies of the original or create different variations from your original primitive object.

In this section, you construct the base for the temple using a polygonal cylinder primitive. The octagonal shape is created by modifying the creation options for the cylinder tool before you create the object. If you did not modify the cylinder options you would create a round cylinder.

To create a polygonal cylinder for the base

- 1 Select the Polygons menu set.

NOTE Unless otherwise indicated, the directions in this lesson for making menu selections assume you've already selected the Polygons menu set.

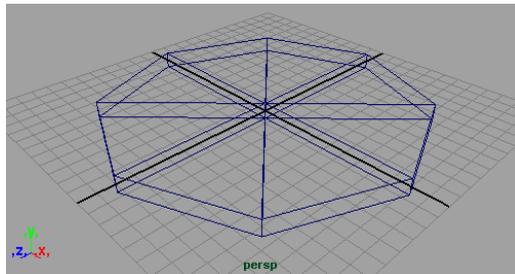
You should also ensure that the Interactive Creation option for primitives is first turned off by selecting Create > Polygon Primitives > Interactive Creation to ensure the check mark does not appear beside the item.

- 2 From the main menu, select Create > Polygon Primitives > Cylinder > 

An option window appears.

- 3 In the Polygon Cylinder Options window, select Edit > Reset Settings and then set the following options:
 - Radius: 10
 - Height: 1
 - Axis divisions: 8
 - Height divisions: 1
 - Cap divisions: 1
 - Axis: Y
- 4 In the Polygon Cylinder Options window, click Create.

Maya creates a cylinder primitive object that is octagonal in shape and positioned at the center of the Maya workspace. This cylinder is 20 units wide by one unit high, and has eight faceted sides.



NOTE You were instructed to reset the option settings as a precaution in case they had been set differently. This is a good habit to practice when working with tool options to avoid getting a result that was different from what you expected.

The Toolbox: Layout shortcuts

The Toolbox is located on the left hand side of the Maya user interface. It contains icons that open tools for transforming your objects within Maya

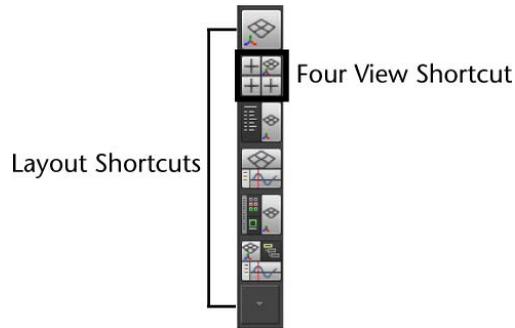
(selection, move, rotate, scale) as well as layout shortcuts for changing the views and panel layouts.

The Quick Layout buttons shortcuts allow you to select a different panel or switch to another layout.

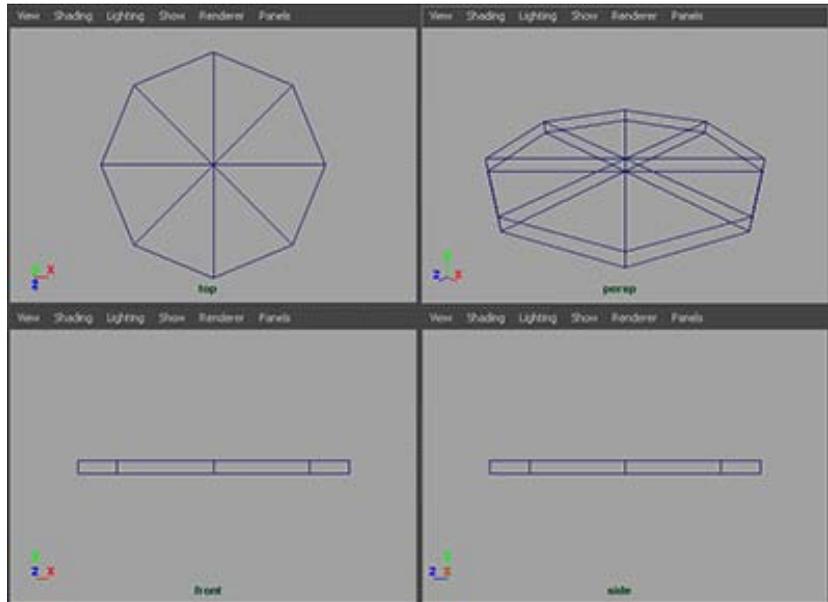
You need to finish positioning the cylinder. To do this you need to see the object from a side view to make sure it is sitting exactly on the ground plane.

To change the panel layout to view the base from a side view

- 1 From the Toolbox, click the Four View layout shortcut.



The workspace changes to a four-view layout. The perspective view is located in the top right corner and the other views show the object from the top, front and side. The layout shortcuts have other options that you will learn later in this tutorial.



It's now possible to see the base from the side view, but it would be easier to determine the position of the base if the side view were enlarged to a full view.

- 2 To enlarge the side view, position the mouse cursor in the side view, and tap the spacebar of your keyboard.

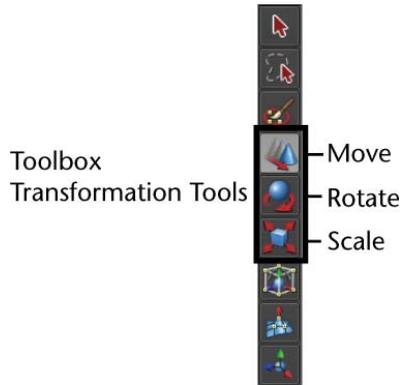
The workspace changes to a single view layout with the side view in an enlarged view. It is easier to view the position of the base from this side view. Notice that the base lies slightly above and below the ground plane (X, Z).

TIP You can position your mouse cursor in any scene view and tap the spacebar once to toggle the view. If the view is a full panel view, it will change to a four panel view and vice versa.

The Toolbox: Transformation tools

You need to move the base slightly upwards in the Y direction so it is positioned on the X, Z plane. To do this you use the Move transformation tool located in the Toolbox.

The upper half of the Toolbox contains the tools for transforming objects (selection, move, rotate, scale) within Maya. When you move your mouse cursor over any transformation tool icon you see the name of the tool appear next to the mouse cursor.

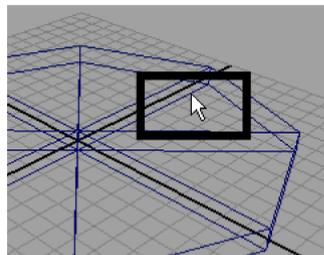


The tool's name also appears in the Help Line at the bottom of the Maya window. The Help Line has an additional purpose: it displays summary instructions as you use tools that require several steps.

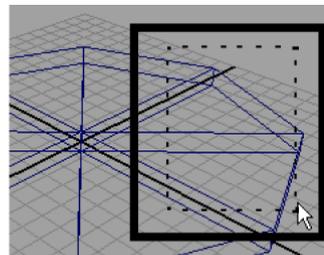


Selection and de-selection of objects

Before you can transform an object, you must ensure it is selected. You can select objects by clicking them directly, or by dragging a rectangular bounding box around some portion of the object to indicate what you want selected. To deselect an object, you simply click somewhere off of the selected object.



click on wireframe object to select



Bounding box selection

To select the base primitive object in the scene view

- Do one of the following:
 - With your left mouse button, click the object's wireframe outline in the scene view.
 - With your left mouse button, drag a bounding box around one corner or edge of the object's wireframe.
The object is selected when its wireframe outline color displays in a bright green color. If it is not selected, its display color is navy blue.

To use the Move Tool to adjust the position of the base

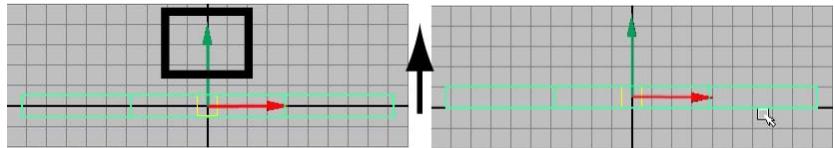
- 1 Select the Move Tool from the Toolbox.

A move manipulator icon appears over the primitive cylinder in the scene view.

The Move Tool Manipulator has handles that point in the direction of the three fundamental axis directions of 3D space: X, Y, Z. The handles are colored red, green, and blue based on their function related to the X, Y, Z axes and control the direction of the movement along an axis.

When you click a specific handle, it indicates that the move is constrained to that particular axis direction.

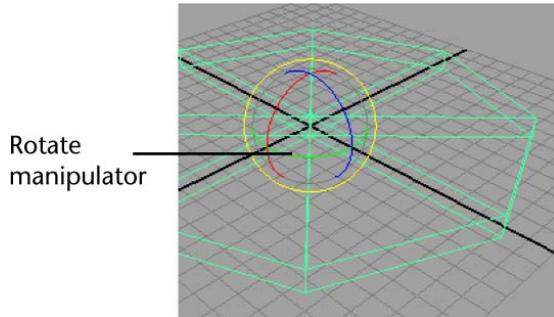
- 2 In the side view, drag the green Y manipulator handle to move the primitive cylinder upwards in the Y direction. Move it upwards enough so that the bottom of the base cylinder is aligned with the X axis (the thick dark line of the grid)



The base cylinder now needs to be rotated slightly so the front of the base is parallel to a grid line. Since each facet of the octagon represents 45 degrees of a circle, you need to rotate the object approximately half of that amount or 22.5 degrees.

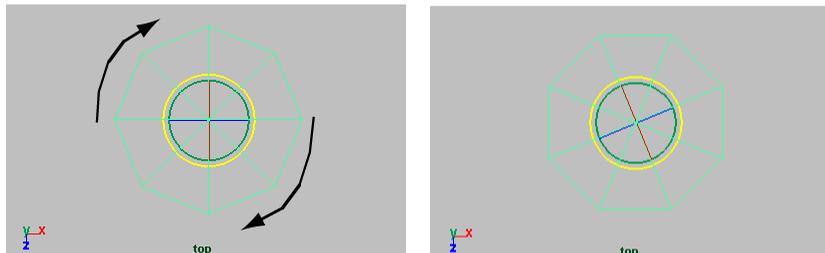
To use the Rotate Tool to adjust the position of the base

- 1 Display all four views by positioning the mouse cursor in the view and tapping the spacebar of your keyboard.
The four view panel appears.
- 2 Position the mouse cursor in the top view and tap the spacebar once.
The top view appears in the workspace.
- 3 With the base cylinder selected, choose the Rotate tool from the Toolbox.
A rotate manipulator icon appears over the primitive cylinder in the scene view.



The Rotate Tool manipulator consists of three rings (handles), plus a virtual sphere enclosed by the rings. The colors of the handles correspond to the X, Y, and Z axes. The handles are colored red, green, and blue based on their function related to the X, Y, Z axes and control the direction of the rotation around an axis.

- 4 In the top view, drag the green Y manipulator ring to rotate the primitive cylinder so that one of the facets of the base cylinder is aligned with the grid as shown in the image below.
You are rotating the cylinder around its Y axis.



You may be asking yourself the question “How do I know if I’ve rotated the base exactly 22.5 degrees?” You can check the accuracy of the rotation by viewing the Channel Box. Rotate Y should be close to 22.5 degrees.

TIP You can undo and redo the last action you performed. Undo reverses the last action you performed on a selected object. It also reverses any action you performed from the Edit Menu.

To undo an action select, Edit > Undo, Redo, Repeat. Maya allows you to perform multiple undos.

The Channel Box

The Channel Box is an editing panel that provides you access to an object’s transformation information and much more. It provides information on three distinct areas for any type of object: The transform node, shape node, and input node.

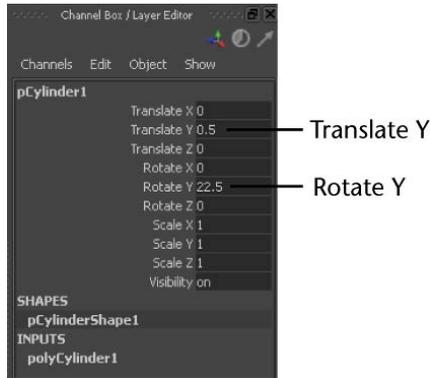
Nodes are where information about object types are kept track of within Maya. Nodes are comprised of attributes. *Attributes* refer to information related to what the node is designed to accomplish. In this case, information about the primitive cylinder’s Y axis rotation is referred to as the Rotate Y *attribute*. You will learn more about nodes later in this tutorial.

When you moved and rotated the cylinder primitive using the Move Tool, you were doing this by your own visual judgement. This will usually be sufficient for many of your creative applications.

If you need to control the attribute of an object with more accuracy you can do this by entering the precise values into the appropriate attribute field of the Channel Box.

To move and rotate the base using the Channel Box

- 1 With the base cylinder selected, view the Transformation attributes in the Channel Box. Specifically, view the values for Translate Y, and Rotate Y.

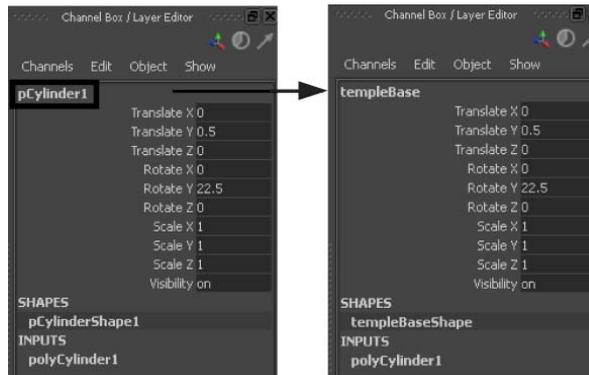


- 2 In the Channel Box, adjust the attribute values so they match the above image by clicking in the field and entering the correct numerical values. This accurately positions the base in your Maya scene.

Maya named the cylinder primitive when it was first created. Rename the cylinder to something more meaningful to your project.

To rename the cylinder primitive using the Channel Box

- 1 In the Channel Box, click in the field with the name *pCylinder1*.
- 2 Rename the primitive object by typing the new name: `templeBase` and then pressing Enter.



Duplicating objects

Duplicating an existing object is a useful way to make an exact copy of it without having to start over. When you duplicate an item the copy takes on the characteristics of the original. Using the Duplicate Tool you can additionally apply transformations to the copy (move, rotate, scale).

Return to a four view layout to view what you've accomplished to this point.

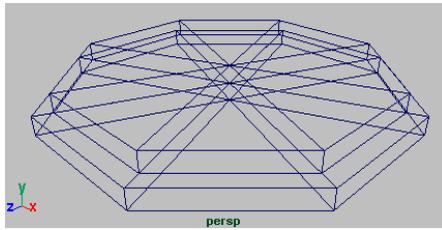
To change the panel layout to a four view layout

- From the Toolbox, click the Four View layout shortcut.
The workspace changes to a four view layout. It is easier to view the work from this four view layout.
The base for the temple is constructed of two levels and appears stepped. You duplicate and scale the templeBase object using the duplicate tool.

To duplicate the temple base

- 1 Display all four views by positioning the mouse cursor in the view and tapping the spacebar. Then click in the perspective view to display the base in this view.
- 2 With templeBase selected, choose Edit > Duplicate Special >  from the main menu.
The Duplicate Special Options window appears.
- 3 In the Duplicate Special Options window, select Edit > Reset Settings and then set the following options:
 - Translate: 0 1.0 0
 - Rotate: 0 0 0
 - Scale: 0.9 1.0 0.9
 - Geometry Type: Copy
 - Group under: Parent
- 4 In the Duplicate Options window, click Duplicate Special.
Maya creates a duplicate of the templeBase object that is scaled to 0.9 of the original in the X, Z axes, and is one unit above *templeBase*. As a result of the scale operation, the base for the temple now appears stepped.

Maya keeps track of the name of the duplicated object based on the name of original and renames the duplicated object *templeBase1*.



Save your work

In the last lesson we recommended that you save your work at regular intervals. An example of this is when you have just completed a major task such as constructing the base for the temple. With this strategy, if you ever make a mistake, you can always open the previously saved version of your work and begin from there.

To save your Maya scene

- 1 Select File > Save Scene.

A file browser appears showing the `GettingStarted` project directory where you can save your scene.

If you do not see the `GettingStarted` project directory, you have not yet copied the `GettingStarted` folder and set it as your Maya project. For more information, see [Copying and setting the Maya project](#) on page 25.

- 2 Within the `GettingStarted` project directory, ensure that you are saving in the `scenes` sub-directory.
- 3 Type `Lesson2Base` in the file name text box.
- 4 Click Save.

Beyond the lesson

In this lesson you continued with the fundamental tools and skills to successfully learn:

- An overall workflow for constructing forms using primitive objects.

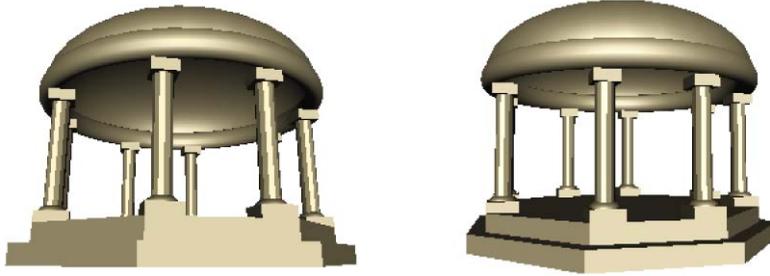
- Where the primitive object tools are located in the main menu.
- How to create a primitive object as well as reset and edit its creation options.
- How to change between a single view and four view panel layout using layout shortcuts and by tapping the spacebar of your keyboard.
- How to select objects by clicking them with your mouse.
- How to move and rotate objects using the transformation tools in the Toolbox.
- That tool manipulators can constrain a transformation to the X, Y, or Z axes.
- How to edit an object's transformation node attributes (move, rotate, scale) accurately using the Channel Box.
- How to rename objects using the Channel Box.
- How to duplicate objects and apply transformations while doing so.

We suggest you additionally practice the following tasks on your own:

- Creating other primitive object types, with various options so that you can understand the variations that are possible.
- Try using the ViewCube™ located in the upper-right corner of the active scene view to change the camera's viewing angle in relation to the objects in the scene.
- Practice moving, rotating, and scaling objects, and changing between the various scene views (single perspective, four view, single side, single top etc.)

Lesson 3: Viewing the Maya 3D scene

Introduction



In the previous lesson you learned how to view your 3D scene by changing between single and four view layouts. It is important for you to learn how to change your views in a more interactive manner so that you can: view your objects close up or far away, select objects more accurately, or view objects from different angles in your perspective view.

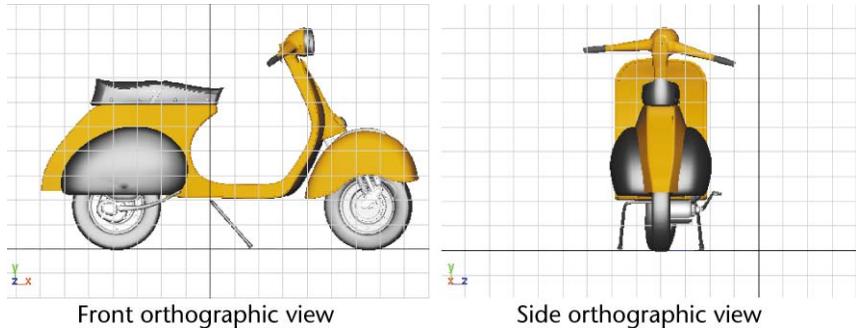
In this lesson you learn how to:

- Understand the difference between moving objects in the scene and moving the point of view on the scene.
- Use the dolly, track, and tumble camera tools to change the view of your scene in both the orthographic and perspective views.
- Rotate objects using the transformation tools in the Toolbox.
- Select objects using a variety of techniques.
- Group objects together so they can be transformed as a unit.
- Display objects in both wireframe and shaded modes.
- Use additional primitive objects and options.

Camera tools

In the lessons so far, when you looked at an object from the top, front, or side views you have been viewing the scene through an *orthographic view*. Orthographic views appear two-dimensional because the object is displayed

using parallel projections of only two axes at a time. (Scooter images courtesy of *The Art of Maya*)



When you view the scene through the perspective view, you are viewing the scene in a three-dimensional manner. The perspective view simulates what your scene would look like from a camera's point of view.

Perspective View

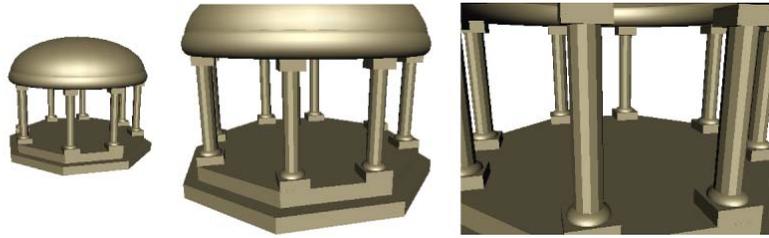


In Maya, you view the scene through a set of virtual cameras. These cameras are either orthographic or perspective in nature. You can adjust how these cameras view the scene using the Camera Tools.

The three primary methods for manipulating the camera view are *dolly*, *tumble*, and *track*.

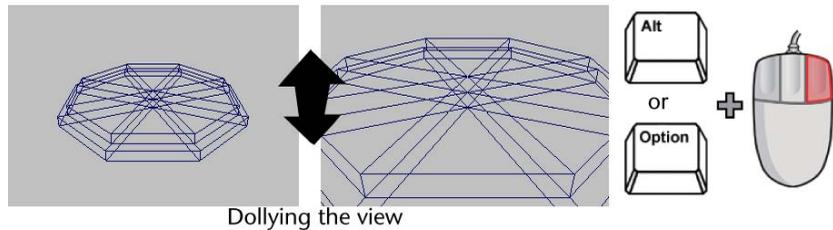
Dolly Tool

The Dolly Tool gets its name from filmmaking where a camera, mounted on a wheeled tripod, is moved towards or away from the scene. In Maya, dollying allows you to view the items in your scene either close up or from further back.



To dolly the perspective view

- 1 Enlarge the scene view to a single perspective view.
- 2 Do one of the following:
 - (Windows & Linux) Press the Alt key and drag the mouse to the right while holding down the right button on your mouse.
 - (Mac OS X) Press the Option key and drag the mouse to the right while holding down the right button on your mouse.



- 3 To dolly the camera outwards from the subject in the scene you can perform the same key and mouse combinations as described above but drag the mouse to the left.

Dolly works in both the perspective and orthographic views.

TIP If you make an error when adjusting your camera view of the scene, you can reset the camera to its default home setting.

To reset the camera view for a particular orthographic or perspective view:

From the panel menu, select View > Default Home.

Tumble Tool

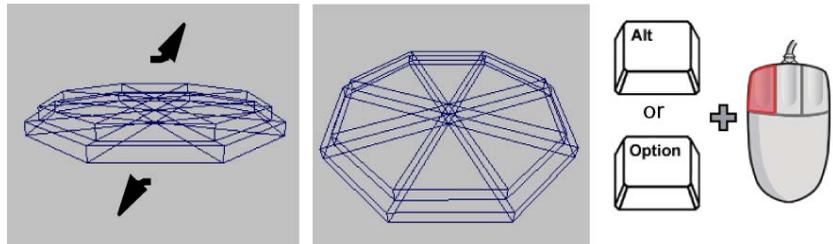
The Tumble Tool allows you to tumble or rotate the camera's view around a particular center of interest to achieve either a higher or lower vantage point, or a different side angle.



To tumble the perspective view

- Press the Alt key (Windows & Linux) or the Option key (Mac OS X) and drag the mouse either left or right, or up or down, while holding down the left button on your mouse.

Tumbling the view revolves the camera around the center of the scene view, in whichever direction you drag (left, right, up or down). The Tumble Tool does not work in the orthographic views.



Tumbling the perspective view

Track Tool

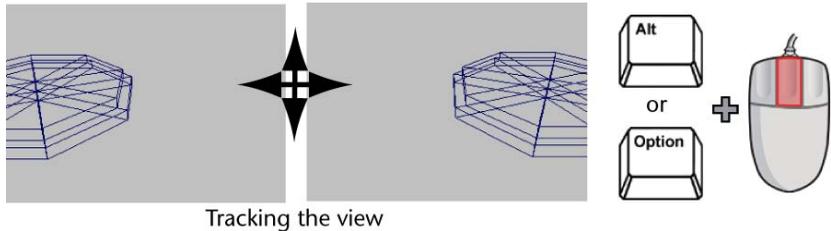
The Track Tool allows you to move the camera up, down, or sideways in relation to the scene.



To track the perspective view

- Press the Alt key (Windows & Linux) or the Option key (Mac OS X) and drag the mouse in any direction, while holding down the middle button on your mouse.

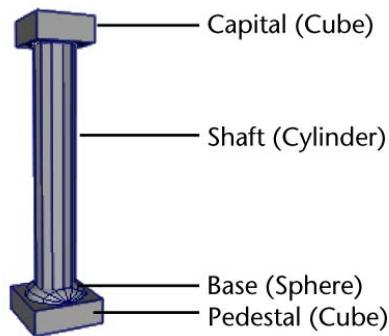
The Track Tool works for both orthographic and perspective views.



NOTE Even though the objects appear to move across the screen when operating any of these camera tools, it is the viewing camera that is actually moved in relation to the scene, not the objects.

Workflow overview

The columns are made up of multiple primitives that are moved, scaled, and rotated into position. Once the first column is created, with each component named and accurately positioned, you will group and duplicate it to create others.



To create a polygonal cube for the pedestal

- 1 From the Main Menu, select Create > Polygon Primitives > Cube >  .

- 2 In the Polygon Cube Options window, select Edit > Reset Settings and then set the following options:
 - Width: 1.75
 - Height: 0.6
 - Depth: 1.75

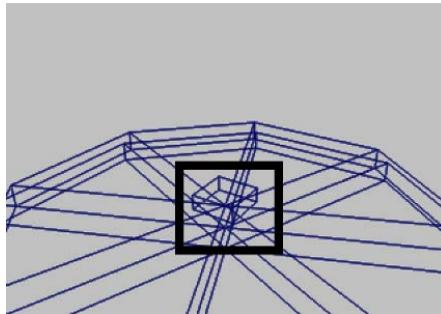
Leave the other options at their default settings.

NOTE If the Polygon Cube Options window does not appear, ensure that the Interactive Creation option for primitives is turned off by first selecting Create > Polygon Primitives > Interactive Creation so that a check mark does not appear beside this menu item.

- 3 In the Polygon Cube Options window, click Create.
Maya creates a cube primitive and positions it at the origin.
- 4 In the side view, move the cube upwards (Y axis) so it rests on the top surface of the temple base.

You can do this using the Move Tool or with the Channel Box. If you use the Channel Box, enter a Translate Y value of 2.3.

You may find it useful to dolly or tumble the scene view to obtain a better viewpoint.



- 5 In the Channel Box, rename the cube *columnPedestal*.

To create a polygonal cylinder for the shaft

- 1 From the main menu, select Create > Polygon Primitives > Cylinder >

2 In the Polygon Cylinder Options window, select Edit > Reset Settings and then set the following options:

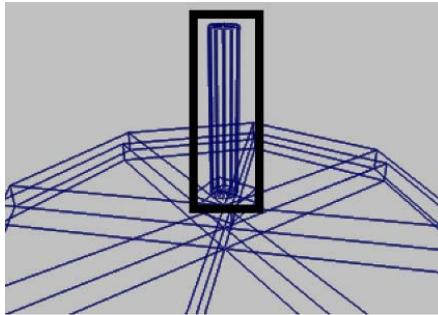
- Radius: 0.5
- Height: 6
- Axis divisions: 12

Leave the other options at their default settings

3 In the Polygon Cylinder Options window, click Create.
Maya creates the cylinder primitive at the origin.

4 In side view, move the cylinder upwards (Y axis) so it rests on the top surface of `columnPedestal`.

You can do this using the Move Tool or with the Channel Box. If you use the Channel Box, enter a Translate Y value of 5.6.



5 In the Channel Box, rename the cylinder `columnShaft`.

The capital for the column rests on top of the column and is very similar to the pedestal. You duplicate the pedestal and position the duplicate at the top of the column.

To duplicate the pedestal to create the capital

1 With only `columnPedestal` selected, select Edit > Duplicate Special > from the main menu.

The Duplicate Special Options window appears.

2 In the Duplicate Special Options window, select Edit > Reset Settings and then set the following options:

- Translate: 0 6.6 0

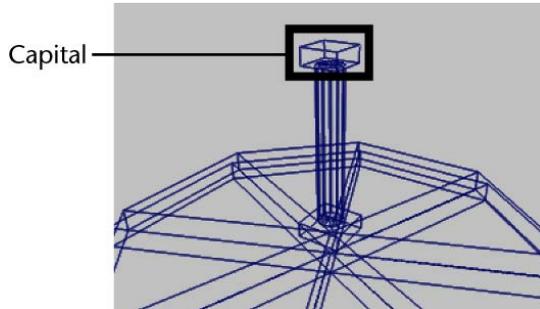
- Scale: 0.8 1.0 0.8

Leave the other options at their default settings.

- 3 In the Duplicate Special Options window, click Duplicate Special.
Maya creates a duplicate of the columnPedestal object and moves and scales it based on the options you set.

NOTE If you positioned the geometry for the column using the Transform Tools and your mouse, the Y translate values may be incorrect for your particular model. You may want to continue positioning the objects by visual reference using your mouse.

Using the duplicate options is an alternate method for positioning and scaling duplicated objects when you can anticipate its final location.



- 4 In the Channel Box, rename the duplicated cube *columnCapital*.

The base for the column rests on top of the pedestal. You will create the base using one half of a NURBS sphere primitive and then move and rotate it into position. You will do this by modifying the creation options for the sphere primitive.

To create a NURBS sphere for the column base

- 1 Select Create > NURBS Primitives > Sphere > .
The NURBS Sphere Options window appears.

NOTE If the NURBS Sphere Options window does not appear, make sure you have Create > NURBS Primitive > Interactive Creation turned off and then try again.

2 In the NURBS Sphere Options window, select Edit > Reset Settings and then set the following options:

- Start Sweep Angle: 0
- End Sweep Angle: 180
- Radius: 0.75
- Number of Sections: 8
- Number of Spans: 4

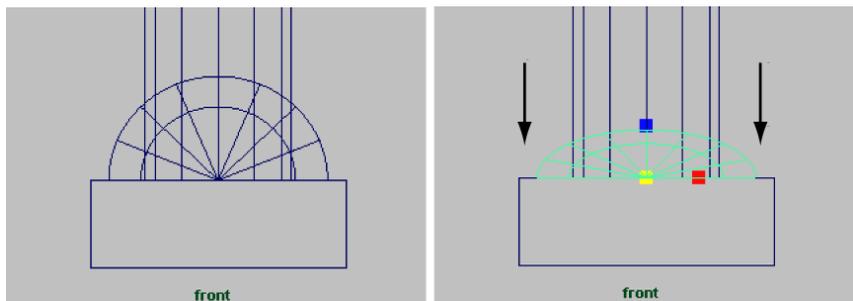
Leave the other options at their default settings.

3 In the NURBS Sphere Options window, click Create.
Maya creates a half-sphere primitive at the origin.

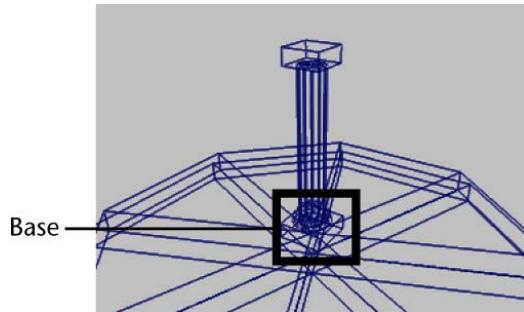
The sphere needs to be rotated 90 degrees and then positioned on top of the pedestal.

To rotate and position the sphere on the pedestal

- 1 In side view, rotate the sphere so that the dome part is pointing up. This is accomplished by either of the following methods:
 - Rotating the sphere about the X axis using the Rotate Tool's manipulator handle.
 - Using the Channel Box to change the Rotate X value to -90.
- 2 Move the sphere so it rests on the top surface of columnPedestal (Translate Y = 2.6, if you have been inputting values into the Channel Box).
- 3 Using the Scale Tool, scale the sphere along its Z axis (blue manipulator handle) so that the sphere becomes slightly squashed in appearance.



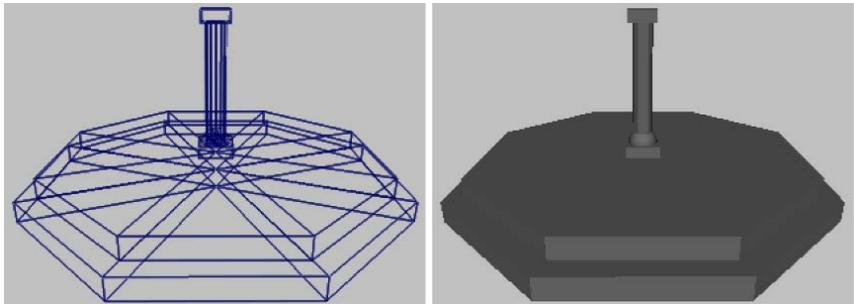
When you scale an object non-uniformly along one of its axes, you are scaling it *non-proportionally*.



- 4 In the Channel Box, rename the sphere *columnBase*.

Viewing objects in shaded mode

Up to this point, you have been viewing your objects in the default wireframe mode. In wireframe mode, objects appear transparent except for the simple wire outline that indicates their position and general shape. Maya provides several options for displaying objects in a shaded manner.



Change the display of your scene so that the objects display as shaded objects.

To display the objects in smooth shaded mode

- 1 Enlarge your perspective view, and dolly and tumble the scene so you can easily view what you've completed so far.
- 2 From the panel menu, select Shading > Smooth Shade All.

The objects in your scene display in an opaque dark gray color. You can work with objects in either shaded or unshaded mode for the balance of this lesson.

Grouping objects

When you need to move, scale or rotate multiple objects as one unit it is easier if they have been grouped together so that they transform as one unit.

Many primitive objects in Maya are grouped objects. For example, the NURBS cube primitive is comprised of 6 flat squares or *planes* that have been grouped together as one unit. When the plane objects are grouped together they create a *hierarchy*.

A hierarchy is a collection of nodes or objects that are connected together to form a unit for some purpose. Hierarchies are useful for describing how the objects within them share similar characteristics or *attributes*; move, scale, rotate for example.

To group the objects for the column

- 1 Select the four objects that comprise the column simultaneously by doing one of the following:
 - With your left mouse button, shift-click each object in turn until the four objects are selected in the scene view.
 - With your left mouse button, drag one large bounding box around the column objects in an orthographic view.

It is important that you do not select any of the templeBase objects as part of your selection. If you accidentally select any of the base objects, deselect them.

- 2 From the main menu, select Edit > Group >  .
- 3 In the Group Options window, select Edit > Reset Settings and then set the following options:
 - Group Under: Parent

Leave the other options at their default settings

- 4 In the Group Options window, click Group.
Maya groups the objects together in a *hierarchy*. You will learn more about hierarchies in the steps that follow.

The Hypergraph

The Hypergraph is a window that shows how the nodes and their connections are organized in your scene. You view object hierarchies and dependencies in the Hypergraph. Use the Hypergraph to view what happens when you group an object.

To view the Hypergraph

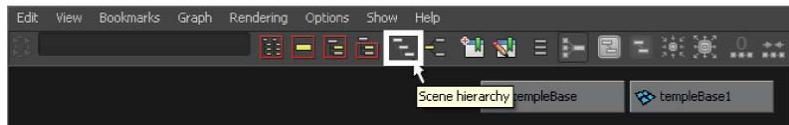
- 1 From the view menu, select Panels > Layouts > Two Panes Stacked.

The scene view splits into two viewing panels - each has their own separate view menu. You will set these to view the scene in the upper view and the Hypergraph in the other.

- 2 From the lower pane menu, select Panels > Hypergraph Panel > Hypergraph Hierarchy.

The Hypergraph panel will display below the scene view panel.

- 3 At the top of the Hypergraph panel, select the Scene Hierarchy icon to ensure the Hypergraph is displaying the scene hierarchy.



- 4 In the Hypergraph panel, select View > Frame All.

The Hypergraph displays the hierarchy for all of the objects in the scene. This approach to viewing the entities in the scene provides a very graphical approach to viewing all of the various nodes in your scene.



In the Hypergraph, each node is represented as a rectangle labelled with an icon that denotes the type of information it represents (for example, surface, shading, and so on). Each node has a unique name assigned to it when it is first created. When you rename your objects, you are actually renaming the node associated with that object.

Some nodes display with a line connecting them. This denotes that they are in a hierarchy and have a dependency structure based on how they were originally grouped.

For the temple's column objects, the hierarchy displays each of the named objects under a node labelled *group1*. Group1 is the *parent node* for this hierarchy of objects.

In Maya, when the *parent node* (sometimes referred to as the root node) is moved, rotated, or scaled in any way, the child nodes (sometimes referred to as the leaf nodes) underneath are also affected.

When you select objects at the top level of a hierarchy and move them, the objects within the hierarchy or group follow.

NOTE This system of nodes, attributes, and hierarchies may initially appear somewhat complex, but it is one of the most powerful features of Maya. The node based architecture provides flexibility and power to create complex models, shaders, and animations.

To rename the parent node in the Hypergraph

- 1 In the Hypergraph, click on the *group1* node so it becomes active.
In the scene view, all of the objects in the column group become selected as a result of selecting the group at the top (parent) level of the hierarchy.
- 2 In the Hypergraph, right-click the top node representing *group1* and select Rename from the pop-up menu.
A small text box appears in the node.
- 3 Enter `Column` as the new name.

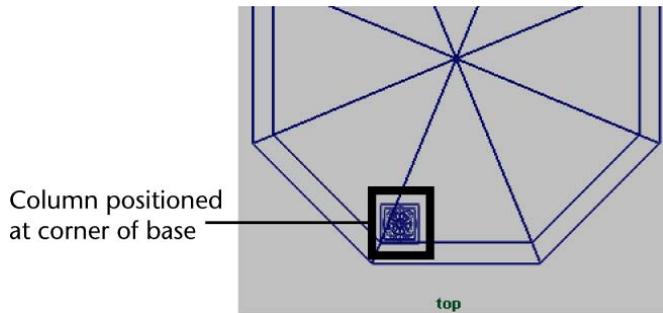
Now that the column is grouped, you need to position it at one corner of the temple base.

To position the column on the temple base

- 1 Change the display of objects to wireframe mode by tapping the 4 key on your keyboard.
This is a keyboard shortcut. Instead of selecting the item from the menu you can use a single key to implement the command.

TIP Many of the tools and features in Maya can be accessed using keyboard shortcuts. In Maya, these shortcuts are called *Hotkeys*. Some Hotkeys are displayed directly beside the menu item, others are listed in the Hotkey Editor. For a complete listing of available hotkeys, go to Hotkey Editor.

- 2 In the Hypergraph, select the *Column* at the top node so that *Column* becomes active in the scene view.
- 3 In the scene view, use the Move Tool to position *Column* at the front corner of the temple base as shown below.



With your first column in position, you can now create a copy of the column and position it on the adjacent corner of the base.

To create a duplicate copy of the column

- 1 With *Column* still selected in the Hypergraph, select Edit > Duplicate Special >  from the main menu. The Duplicate Special Options window appears.
- 2 In the Duplicate Special Options window, select Edit > Reset Settings and then set the following options:
 - Number of Copies: 1Leave the other options at their default settings.
- 3 In the Duplicate Special Options window, click Duplicate Special. The Hypergraph view updates to show an additional column object in the scene. The copy also takes on the prefix name of the original group and is now labelled *Column1*.

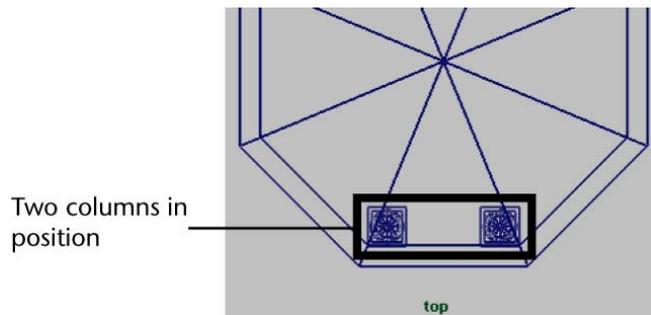
In the scene view it appears that nothing was actually duplicated. When an object is duplicated without any transformations the copy is positioned

in the same position as the original. The two objects are on top of each other.

You need to move the column into position on the adjacent corner of the temple base.

To move the duplicate column into position on the base

- 1 In the Hypergraph, ensure *Column1* is selected by clicking on its top node so it becomes selected in the scene view.
- 2 In the scene view, use the Move Tool to position *Column1* on the adjacent corner of the temple base as shown below.



- 3 From the Toolbox, click the Four View layout shortcut.
The workspace changes to a four view layout and the Hypergraph is no longer displayed.

Selection modes and masks

It isn't always efficient to have the Hypergraph window open when you want to select an object at a particular level within its hierarchy. Maya allows you to select items in different selection modes depending upon your specific needs.

There are three main types of selection modes: Hierarchy, Object and Component. You use these modes in order to *mask* or limit the selection of other objects in order to select only the types of items you want. When you use a selection mask you are filtering out or masking items you don't want to be chosen as part of the selection.

The icons for the three modes appear on the Status Line.



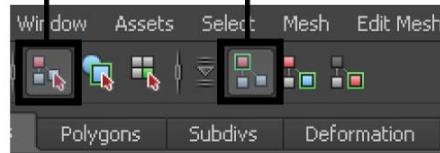
When you first start Maya, the default selection mode is set to Objects. This is useful for much of your selection work with Maya, with a few exceptions. When you want to select items that have been grouped, set the selection mode to Hierarchy.

TIP If you set the selection mask, it will remain that way until you change it again. If an item won't select for you in Maya, you should check the selection mask setting to see if it is set correctly.

To use the Hierarchy and Combinations selection mask

- 1 On the Status Line, choose the Select by Hierarchy and Combinations icon
The Selection Mask icons update to display the three selection choices.
- 2 On the Status Line, choose the Select by hierarchy: Root icon.

Hierarchy and Combinations Select by hierarchy



Select by hierarchy: Root ensures that when you select items they are selected at their parent or root node.

- 3 In the scene view, shift-click *Column* and *Column1* so they are selected simultaneously. (Do not be concerned that they highlight in different colors.)

Pivot points

A pivot point is a specific position in 3D space that is used as a reference for the transformations of objects. All objects (curves, surfaces, groups) have pivot points.

When you group objects in Maya, a new node called a *parent node* is created for the group of objects. The pivot point for the group's parent node is placed at the origin (0, 0, 0). This is useful if you later want to duplicate and rotate the objects radially (that is, in a circular fashion around the pivot).

To group the two columns

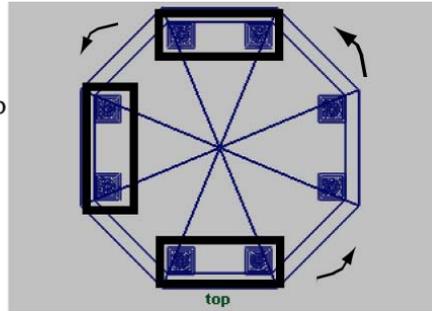
- 1 Ensure *Column* and *Column1* are selected.
- 2 From the main menu, select Edit > Group > .
- 3 In the Group Options window, select Edit > Reset Settings. Set the following options:
 - Group Under: Parent
- 4 In the Group Options window, click Group.
Maya groups the objects together in a hierarchy and the pivot point is positioned at the origin. (When the pivot point is relocated to the origin the Move Tool manipulator for the selected group appears at the origin.)

With the two columns grouped, you then duplicate the columns with a rotation option, the groups will duplicate and rotate around the pivot point at the origin (0, 0, 0).

To duplicate and rotate the group

- 1 With column group selected, select Edit > Duplicate Special >  from the main menu.
- 2 In the Duplicate Special Options window, select Edit > Reset Settings and then set the following options:
 - Rotate: 0 90 0
 - Number of Copies: 3
- 3 In the Duplicate Special Options window, click Duplicate Special.
The columns are duplicated and rotated by 90 degrees with each copy.

Each column group is duplicated and copied 90 degrees



Save your work

Your temple is taking shape! Save your work before proceeding to the next lesson.

To save your Maya scene

- 1 To save your Maya scene, select File > Save Scene As.
- 2 Type `Lesson3Columns` in the file browser area reserved for file names.
- 3 Click Save.

Beyond the lesson

In this lesson, you explored additional tools and skills within Maya as you continued with the construction of the classic temple. In this lesson you learned how to:

- Use your mouse to change the view of your scene using the dolly, track, and tumble camera tools in both the orthographic and perspective views.
- Rotate objects using the transformation tools in the Toolbox and the Channel Box.
- Group objects so they can be transformed as a unit.
- Display objects in both wire frame and shaded modes.
- Access the Hypergraph to view nodes and hierarchies. You are also learning that many tasks in Maya can be completed or approached using multiple techniques.

- Rename nodes within the Hypergraph.
- Select objects in your scene using the Hypergraph.
- Use the group pivot point to your advantage when duplicating objects.

Before proceeding with the next lesson you may want to review the material presented in this lesson so you are familiar with the concepts and skills associated with them. Some suggested tasks you can try on your own include:

- Practice using the view camera tools (dolly, track, and tumble) so you can navigate within the scene views efficiently.
- Practice moving and rotating objects, and changing between the various scene views (single perspective, four view, single side, single top, and so on).
- View objects in the Hypergraph so you can understand their relationships and hierarchy.

Lesson 4: Components and attributes

Introduction

Working with components is an important part of the workflow when working in Maya. Components describe objects at a more detailed level. As you edit the components of your classic temple, you will learn more about what is possible in Maya.

In this lesson you learn how to:

- Template objects in the scene.
- Understand the difference between objects and components.
- Edit objects at their component level.
- Assign surface material attributes
- Access the Attribute editor

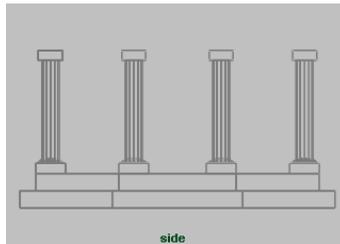
Template display

When your scene becomes complex, templating the display of some of the objects in your scene allows you to more easily select only the objects you want. When you template the display of an object, its wireframe changes to a gray color. The object(s) can still be seen, but not easily selected. This helps to prevent you from selecting or modifying it accidentally. You can change the display of objects to a template and use them as a modeling reference (the way a grid is used).

You need to change the display of the base and columns to template mode so that you can more easily work on the temple roof.

To template the base and columns

- 1 In the side view, with the Selection Mask set to Hierarchy, select all the objects in your scene.
- 2 From the main menu, select Display > Object Display > Template. The selected objects become templated.



For your classic temple, you need to create the entablature using a torus primitive. An entablature is a structure that lies horizontally upon the columns of a temple and supports the roof.

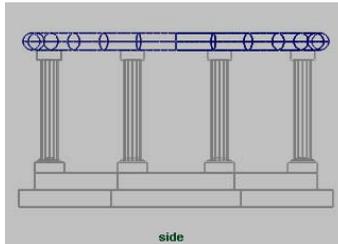
To create and position a torus primitive for the entablature

- 1 From the main menu, select Create > NURBS Primitives > Torus >  .
- 2 In the NURBS Torus Options window, select Edit > Reset Settings and then set the following options:
 - Radius: 8.5
 - Minor Radius: 0.5
 - Number of Sections: 24

- 3 In the NURBS Torus option window, click Create.

NOTE If the NURBS Primitives Options window does not appear, ensure that a check mark does not appear beside the Interactive Creation menu item before selecting Create > NURBS Primitives > Torus > by first selecting Create > NURBS Primitives > Interactive Creation.

- 4 In the Channel Box, rename the torus primitive *Entablature*.
- 5 Move the entablature vertically in the scene so it rests on top of the columns (Translate Y = 9.7).



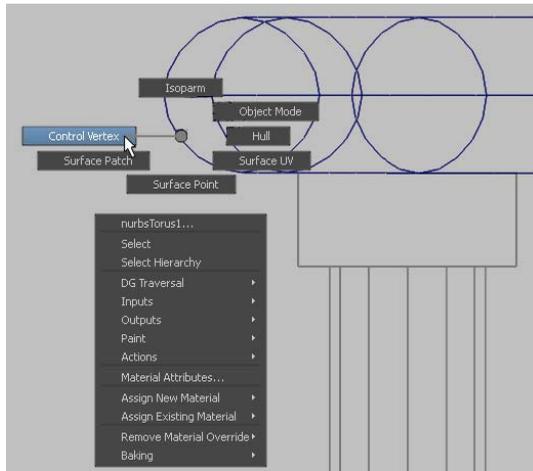
Components

All objects in Maya have a transform and a shape node. Geometric shapes, like the primitives in this tutorial, have smaller parts called *components*. A few examples of components in Maya are control vertices, faces, and hulls. Components allow you to work with objects at a finer level and allow you to edit them in creative ways.

In order to change the shape of the entablature beyond the basic scale transformations, you need to modify its component information.

To select components of the entablature

- 1 In the side view, dolly in for a closer view of *Entablature*.
- 2 Set the Selection Mode to Component mode.
- 3 Right-click the wireframe of Entablature and select Control Vertex from the pop-up menu.

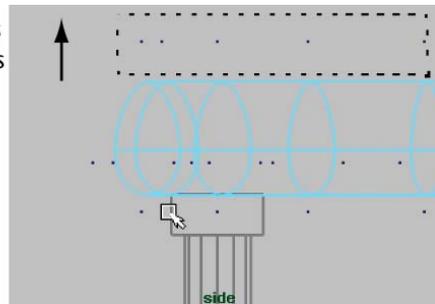


The menu that pops up is a *marking menu* for quickly selecting operations relevant to the object where you right-click the mouse. In this case, the choices pertain to the display of the entablature's components.

A set of small blue squares appear on Entablature called control vertices. Control Vertices (CVs) describe the shape of an object based on their position in space. If you move any combination of these vertices, you change the shape of the object.

- 4 Drag a selection box around the top row of vertices so they become selected.
- 5 Using the Move Tool, move the vertices up vertically as shown below to change the shape of *Entablature*.

Control Vertices
moved upwards



- 6 To cancel the display of the CVs, right-click the wireframe of Entablature once again and select Object Mode from the pop-up menu.

The roof for the temple rests on top of the entablature. The roof for the temple is created using one half of a sphere primitive.

To create a roof for the temple

- 1 Select Create > NURBS Primitives > Sphere >  .
- 2 In the NURBS Sphere Options window, select Edit > Reset Settings and then set the following options:
 - Start Sweep Angle: 0
 - End Sweep Angle: 180
 - Radius: 8.75
 - Number of Sections: 8
 - Number of Spans: 4

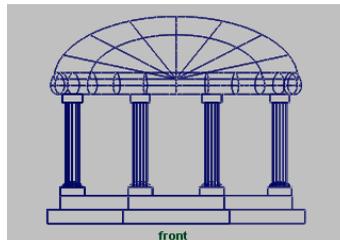
In the NURBS Sphere Options window, click Create.

- 3 Rename the half-sphere *templeRoof*.

The roof needs to be rotated -90 degrees about the X axis and positioned on top of the entablature.

To rotate and position the roof on the entablature

- 1 In side view, rotate the roof so that the dome part is pointing up.
- 2 Move the roof so it is positioned close to the top edge of *Entablature*.
- 3 Scale the roof along its Z axis (blue manipulator handle) so that the sphere becomes slightly squashed in appearance.



Now that the roof is complete, you can untemplate the templated objects.

To untemplate objects

- 1 On the Status Line, choose the Select by Hierarchy and Combinations button.
- 2 On the Status Line, choose the Select by hierarchy: template button.

Hierarchy and Combinations Select by template



Select by hierarchy: template ensures that only templated objects will be affected by a selection.

- 3 In the scene view, drag a selection box around all the objects in the scene so that the templated objects are selected simultaneously.
- 4 From the main menu, select Display > Object Display > Untemplate.
- 5 On the Status Line, choose the Select by hierarchy: root button.

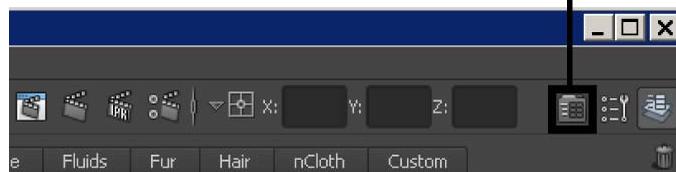
The Attribute Editor

The Attribute Editor provides information about the various nodes and attributes for the objects and materials in your scene. Like the Channel Box, you can view and edit the basic transform information and many other keyable attributes. However, the Attribute Editor provides a more detailed display of all attributes for a selected object.

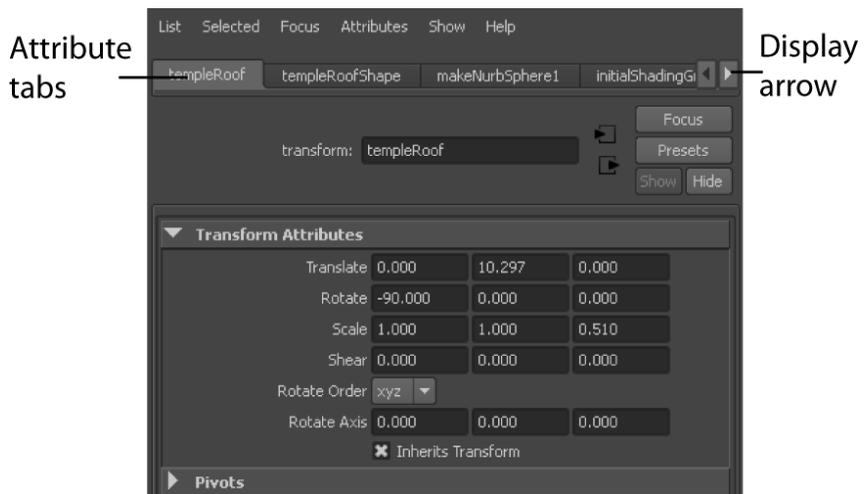
To view object attributes using the Attribute Editor

- 1 In the scene view, select templeRoof so it becomes the selected object.
- 2 To view the Attribute Editor, click the Show/Hide icon on the Status Line.

Show/Hide Attribute Editor



The Attribute Editor displays the attributes for templeRoof. The various attributes for the templeRoof object appear under various tabs. Each tab represents a node.



- 3 Click the templeRoof tab to see its attributes.

This tab is known as the transform node, because the most important attributes on this tab control templeRoof’s transformation. Every visible object in Maya has a transform node, including cameras and lights.

- 4 Click the templeRoofShape tab to see its attributes.

This tab is called the shape node because the attributes establish the object’s geometric shape or physical properties when the object is first created. Most objects have shape nodes, some do not, such as the group for the column objects. The shape node also includes other types of attributes, such as object display attributes.

- 5 Click the makeNurbSphere tab to see its attributes.

This is an input node that includes attributes related to the object’s construction history. The attributes of an input node are passed to another node subsequent in the construction history for the object—in this case, to the templeRoofShape node.

- 6 The last two nodes are initialShadingGroup and lambert1. If you can’t see them, click the display arrow.

The initialShadingGroup and lambert1 nodes are default nodes that relate to the default shading material for an object. Maya uses them to establish the initial color of objects and other settings related to shading. If you

create your own shading materials for the temple, as you will in the following steps, these nodes are replaced by the new shading nodes you create.

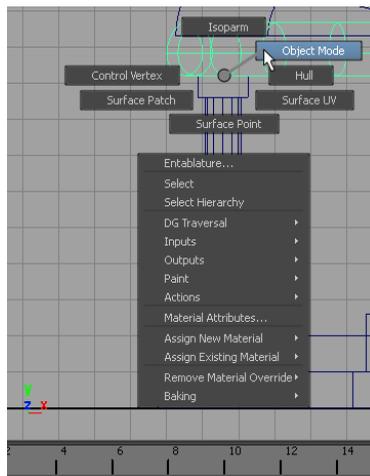
Surface materials

The color, shininess, and reflectivity attributes of an object are controlled by its surface material (sometimes referred to as a shader, or shading material). Material attributes relate to how the object simulates a natural reaction to light in Maya's 3D computer world.

Maya assigns a default shading material to all objects when they are first created. In this section, you learn how to assign a new material to your objects.

To assign a new material to the temple objects

- 1 Right-click the wireframe of Entablature and select Object Mode from the pop-up menu.

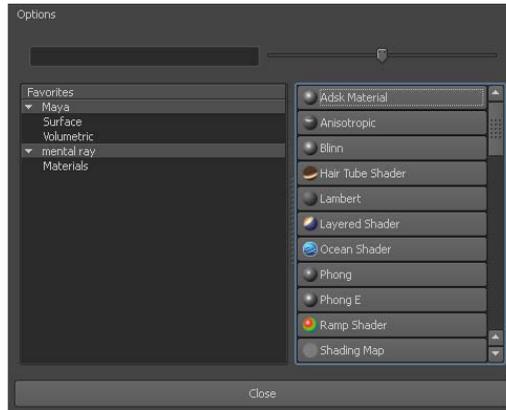


This changes the selection mask back to objects so you can select objects in the scene. It is a shortcut to access this feature.

- 2 Drag a selection box around all the objects in your scene to select them.

TIP If objects won't select in the Maya scene, check that your selection mask is set correctly on the Status Line.

- 3 From the Status Line, select the Rendering menu set using the menu selector.
The main menu changes to display the menu set for Rendering.
- 4 From the main menu, select Lighting/Shading > Assign New Material.
The Assign New Material window appears.

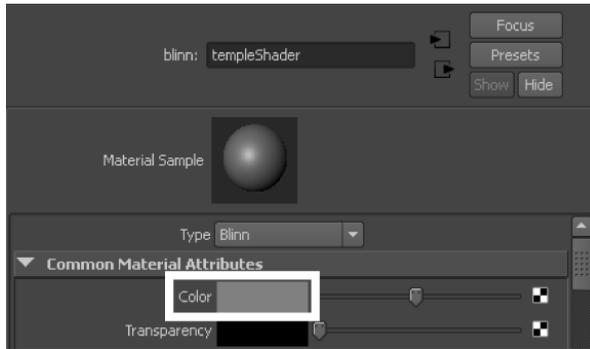


- 5 In the materials list on the right, select Blinn.
A Blinn shading material is assigned to all the objects in the scene and the Attribute Editor updates. Blinn shading materials (named after the inventor of this shading algorithm) provide for high-quality specular highlights on surfaces.
- 6 In the Attribute Editor, rename the blinn1 shading material to *templeShader*.

With a shading material assigned to all the objects, you need to edit the color attributes of the *templeShader* material.

To edit the shading material's attributes

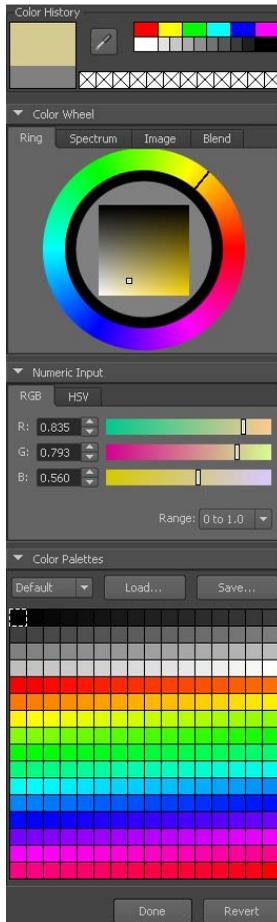
- 1 In the Attribute Editor, click in the gray box to the right of the word Color.



The mini Color Chooser appears.



- 2 Click the ring between yellow and orange to achieve a sand color. You can further refine this color by clicking a shade in the square palette. The exact color is unimportant for this lesson. However, if you'd like more options for choosing your color, you can double-click the the gray box to display the full Color Chooser.



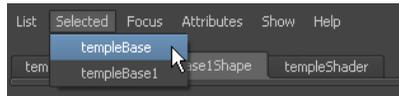
In either case, as you adjust the color wheel indicator, the temple objects become the same color you select in the Color Chooser.

- 3 Click Accept to close the chooser.

You will learn more about Maya's shading and texturing capabilities in future lessons.



- 4 In the Attribute Editor menu, choose the Selected menu item, and select `templeBase` from the list.



The attributes for `templeBase` display in the Attribute Editor.

- 5 Click the `templeShader` node tab to see its attributes.
If you can't see this tab, click the display arrow to the right of the tabs. These are the same attributes you edited when you assigned the `templeShader` shader to all the objects in the scene. When you first create an object, you see two default nodes for shading, `initialShadingGroup` and `lambert1`. When you assign a shading material, the two default nodes are replaced by the attribute node for the assigned shading material.
You will encounter transform, shape, input, and shading nodes throughout your work with Maya. There are other types of nodes that you'll learn about as you continue learning about Maya.
- 6 Close the Attribute Editor using the Show/Hide icon.

Save your work

You have completed this lesson. Save your work before proceeding.

To save your Maya scene

- 1 Select `File > Save Scene As`.
- 2 Type `Lesson4Final` in the file browser area reserved for file names, and then click `Save`.

Beyond the lesson

In this lesson, you completed the construction of the classic temple and learned how to:

- Display objects in template mode.
- Select objects at their component level using selection masks and the right mouse button.
- Assign and edit shading materials for your objects in the scene.
- Access the Attribute editor and view the various node types for objects.

As you proceed through Getting Started with Maya, we assume you are familiar with the fundamental concepts and skills covered in this first chapter.

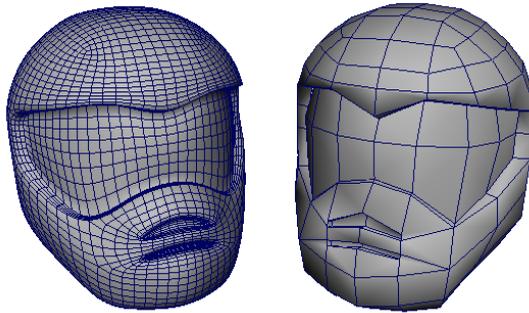
Before proceeding to the next chapter you may want to review the material presented in this lesson so you are familiar with the concepts and skills associated with them. Some suggested tasks you may want to do on your own:

- Practice construction of additional forms using primitive shapes by creating additional details for your temple scene (stairs, roof details, etc.).
- Learn more about the tools and options that have been presented in this lesson, by referring to the Maya Help. To view the Maya Help, select Help > Maya Help.

Polygonal Modeling

3

Introduction



In Maya, *modeling* refers to the process of creating virtual 3D surfaces for the characters and objects in the Maya scene. Surfaces are vital for creating a convincing 3D image.

Modeling is a process requiring keen visual skills and mastery of the modeling tools. The more accurate you are when modeling your forms in terms of size, shape, detail, and proportion, the more convincing your final scene will be.

There are three modeling surface types in Maya:

- Polygons
- NURBS
- Subdivision surfaces.

Each surface type has particular characteristics and benefits.

Polygon surfaces are a network of three-or-more sided flat surfaces called *faces* that get connected together to create a *poly mesh*. Polygon meshes are comprised of vertices, faces, and edges.

The wireframe lines on the mesh represent the *edges* of each face. The regions bounded by the edges are *faces*. Where the edges intersect each other is the location of a point called a *vertex*.

When a polygon mesh is rendered, its edges can be set to appear hard or smooth. As a result, polygons can easily represent both flat as well as curved 3D forms. You'll work with these component types continuously when modeling with polygons.

Polygonal surfaces have a wide range of applications and are the preferred surface type for many 3D applications including interactive games and web development applications.

Polygonal surfaces can be described with the smallest amount of data of all the 3D surface types, and therefore, can be rendered quickly, delivering increased speed and interactive performance to the end user in games and other applications.

Preparing for the lessons

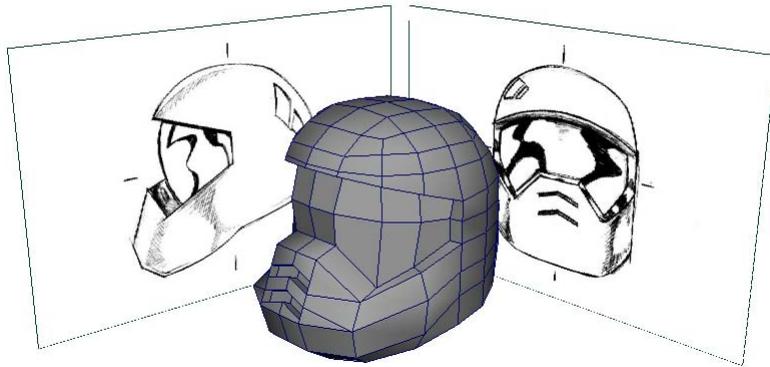
To ensure the lesson works as described, do these steps before beginning:

- 1 Select File > New Scene.
- 2 Make sure Construction History (below the menu bar) is on: . (If it is turned off, the icon has a large X across it).
- 3 Select the Polygons menu set.
Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the Polygons menu set.
- 4 Make sure Display > UI Elements > Help Line is turned on.
You will use the Help Line while modeling.
- 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](#) on page 25.

- 6 As you work through these lessons, remember to save your work frequently and increment the name of the file every so often (`filename1`, `filename2`, and so on).
In this way you'll have earlier versions of your work to return to should the need arise.
- 7 Make sure that Soft Selection is turned off by opening the Tool Settings Editor with the Select Tool active and unchecking the box marked Soft Select.

Lesson 1: Modeling a polygonal mesh from a reference image

Introduction



In this lesson, you model a helmet for an interactive game character, using polygonal surface modeling techniques.

In this lesson you'll be introduced to some of the polygon modeling tools Maya provides by learning how to:

- Use 2D image planes as a reference for constructing 3D models
- Use 3D primitives as the basis for creating more complex models
- Work with the components of a polygon mesh (faces, edges, and vertices)
- Smooth a polygon mesh

- Select the faces, edges, and vertices of polygonal meshes
- Create polygon faces by placing vertices
- Scale and extrude faces on a polygon mesh
- Move and rotate extruded polygonal meshes
- Split vertices and subdivide polygonal faces
- Combine separate meshes into one mesh
- Bridge between meshes
- Add faces to an existing mesh
- Use Snap to Grid
- Preview a smoothed high resolution version of a polygon mesh
- Harden and soften polygon edges

Setting modeling preferences

Before you create your polygon model change some of the default settings to better work through this lesson.

- 1 Select Window > Settings/Preferences > Preferences.
The Maya user preferences window appears.
In the Categories section of the Preferences window select Polygons.
- 2 In the Polygon Display settings, set the following:
 - Border Edges: On
 - Edge Width: 4

This displays border edges prominently on polygon meshes. Displaying border edges distinctively lets you differentiate them from other edges on your models and troubleshoot issues.

- 3 In the Categories section of the Preferences window, choose Selection.
- 4 In the Polygon Selection settings, set the following:
 - Select faces with: Whole face

The Whole face setting lets you select polygon faces by clicking anywhere on the face (the Center setting requires you to click on the face's center

dot). In addition, when a face is selected in the scene, the entire face appears highlighted.

- 5 Click the Save button to close the settings window.
- 6 Turn off the Interactive Creation option for primitives: Select Create > Polygon Primitives > Interactive Creation (no check mark).

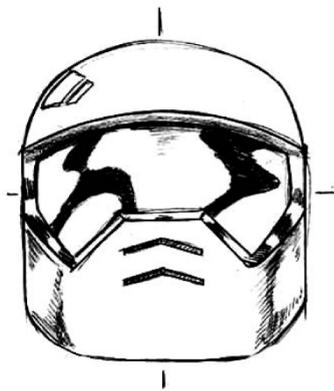
Using 2D reference images

You can use front, side, and top views from drawings, sketches, or photographs to help visualize your 3D model in Maya, much like an architect or engineer creates their designs from the plan and elevation views of a blueprint.

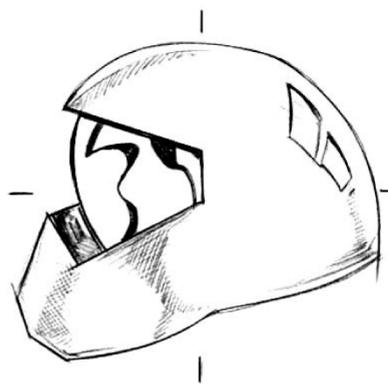
You can import 2D images into your orthographic camera views as *image planes*. An *image plane* is a 2D object (plane) that places an image file in the scene view. By default, an image plane only appears in the camera to which the image plane is connected. Image planes are also used to create backgrounds and environments when rendering.

When you load an image into an image plane, it appears in your selected orthographic view at the origin along an axis that is perpendicular to the selected orthographic view. You can refer to the image in the orthographic view to define the silhouette and character lines of your model. You can move the image plane, change its transparency, or turn it off.

For this lesson you load two images we've provided for the lesson into image planes in the front and side orthographic views of your scene. You'll refer to them frequently while you model the helmet.



Poly_HelmetFront.jpg



Poly_HelmetSide.jpg

To load reference images into the front and side orthographic views

- 1 In the Toolbox, click the Four View shortcut from the Layout Shortcuts bar.
The perspective view is located in the top right corner and the other views show your scene from the top, front, and side.
- 2 In the front view panel menu, select View > Image Plane > Import Image.
- 3 Select the image file named `HelmetFront.jpg`.
This image can be found in the `GettingStarted` directory that you set as your Maya project:
`GettingStarted/PolygonModeling/sourceimages`
- 4 In the browser, click the Open button.
`HelmetFront.jpg` appears in the front view and depicts a drawing of the helmet.
- 5 In the side view panel menu, select View > Image Plane > Import Image.
- 6 Select the image file named `HelmetSide.jpg`.
This image can be found in the `GettingStarted` project directory:
`GettingStarted/PolygonModeling/sourceimages`
- 7 In the browser, click the Open button.
`HelmetSide.jpg` appears in the side view.

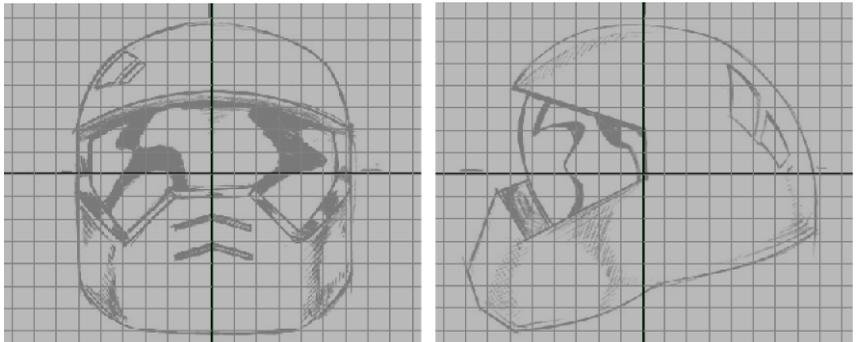
NOTE The front and side images for this lesson were created so that the two views are aligned and the height of each image is identical. When you import them as image planes, they appear at the same scale. These are important considerations when you construct your own reference images in the future. Otherwise, your reference images may not align or may be at different scales between the two orthographic views.

By referring to the image planes in the orthographic views as you work, you can correlate how a feature in one view, appears in another. While a top view reference image is useful in many cases, it isn't critical for this lesson.

The image planes can be set so they appear partially transparent to allow you to work with the polygonal components more easily. To do this you'll select the front and side orthographic cameras and modify the transparency of the images.

To modify the transparency of the reference images

- 1 In the front view panel menu, select View > Select Camera.
The camera for the front orthographic view is selected.
- 2 Display the Channel Box by clicking the Show/Hide Channel Box icon on the Status Line.
The Channel Box displays the keyable attributes for the Front orthographic camera.
- 3 In the Inputs section, click the ImagePlane1 name to display the attributes for the image plane.
- 4 Set the Alpha Gain attribute to a value of 0.25.
The image appears partially transparent.
- 5 In the side view panel menu, select the camera and change the Alpha Gain setting for the side view image plane exactly as you changed the front view image plane.



Front and side images in the orthographic views with Alpha Gain settings reduced.

TIP You can change the Alpha Gain at any time if you want the image planes to appear more or less transparent.

- 6 In the perspective view's panel menu, select Show > Cameras to temporarily turn off the display of the image planes in the perspective view.

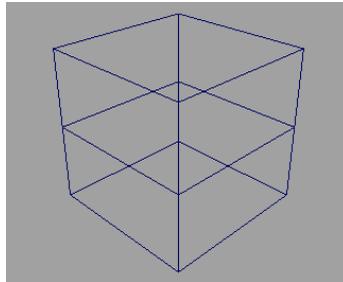
Creating a polygon primitive

You'll create the top portion of the helmet mesh from a cube primitive using the image planes as a reference. Primitive objects are one method for starting 3D meshes because they can be modified to create other forms.

To create a cube primitive for the helmet mesh

- 1 Ensure you are working in the perspective view.
- 2 Select Create > Polygon Primitives > Cube >  .
- 3 In the options window, select Edit > Reset Settings, and then set the following:
 - Width: 14
 - Height: 14
 - Depth: 14
 - Width divisions: 1
 - Height divisions: 2
 - Depth divisions: 1
- 4 Click the Create button.

A cube primitive is created at the origin with the dimensions you specified and one subdivision around the middle. This cube primitive is comprised of four-sided polygon faces called *quads*. Quad polygons are used frequently with 3D character models as they are easily smoothed and deform well when bound into a skeleton.



You can smooth the cube so it appears more rounded on the corners and subdivide it into smaller faces using the Smooth feature.

To smooth and subdivide the cube primitive

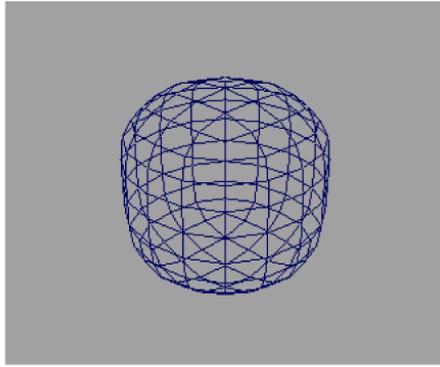
1 With the cube still selected in the scene view, select Mesh > Smooth > , and set the following options in the window that appears:

■ Add divisions: Exponentially

■ Division levels: 2

2 Click the Smooth button

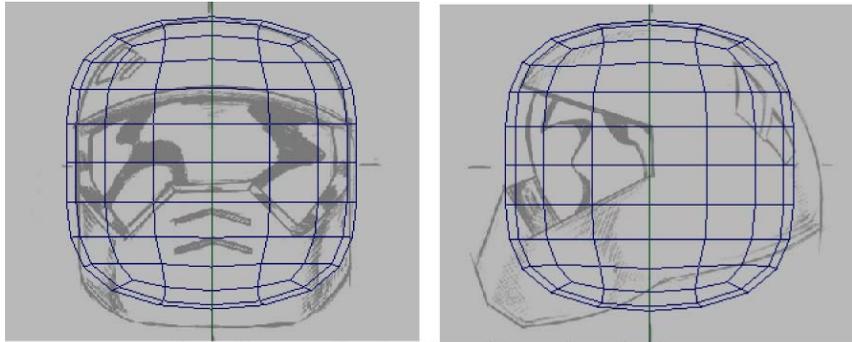
The cube primitive is smoothed and is rounded at its corners. The cube has also been subdivided into smaller faces.



Cube primitive after smoothing has been applied.

The polygon faces are still four-sided even though their shape and position have been modified by the smooth operation.

NOTE If the top, domed portion of the smoothed cube doesn't roughly match the helmet sketches as indicated in the image below, scale the cube with the Scale Tool to make it match.



Scale the smoothed version of the cube primitive so it roughly matches the reference sketches.

To rename the polygon mesh

- With the cube still selected, rename the cube primitive using the Channel Box from `pCube1` to `helmetmesh`.

Modeling in shaded mode

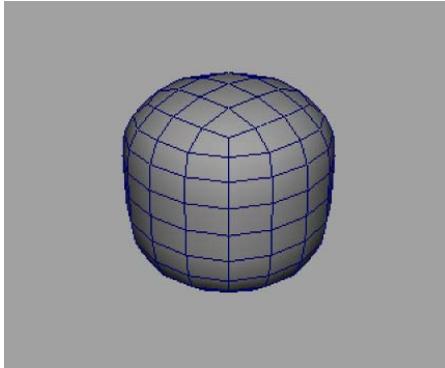
Modeling a polygon mesh in shaded mode gives you a better sense of the 3D volume that the model occupies as well as how any surface details appear.

You can control the shading of an object separately in each view. For example, you can display the object as shaded in the perspective view only, and set the orthographic views to display objects in wireframe mode. You can also set the display settings to show both a shaded and wireframe display simultaneously.

To display the helmet mesh in Wireframe on Shaded mode

- 1 In the perspective view, right-click on the helmet mesh and select Object Mode from the marking menu that appears.
This changes the selection mode to Objects.
- 2 Select the helmet mesh.
- 3 In the perspective view, select Shading > Smooth Shade All from the panel menu.
The helmet mesh displays in smooth shaded mode.
- 4 From the panel menu, select Shading > Wireframe on Shaded.

The wireframe appears on the mesh as well as the shading (whether or not the mesh is selected).



Smoothed primitive in shaded mode.

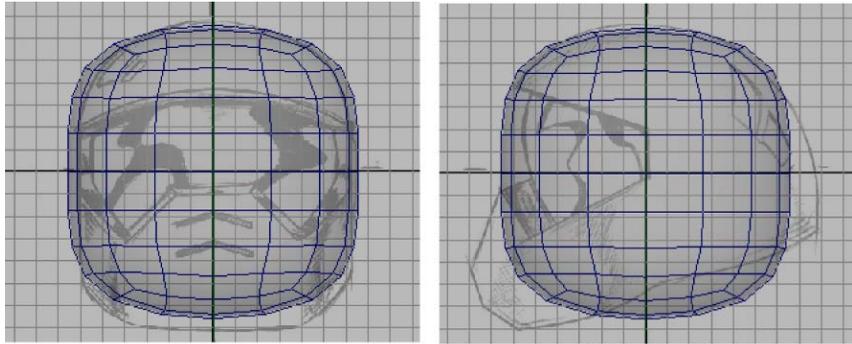
The opacity of the smooth shading in the side and front views prevents you from seeing the image planes behind the object. X-Ray display mode solves this problem by making the smooth shaded surface appear semitransparent.

To display the helmet mesh in shaded X-Ray mode

- 1 With the helmet still selected, choose the following from the side view's panel menu:
 - Shading > Smooth Shade All
 - Shading > Wireframe on Shaded
 - Shading > X-Ray.

The helmet mesh updates to a semitransparent shaded display with the wireframe. You can see the 2D image behind the mesh.

- 2 Repeat the previous step for the front view as well.



Cube primitive with shading set to Smooth Shade All, Wireframe on Shaded, and X-ray.

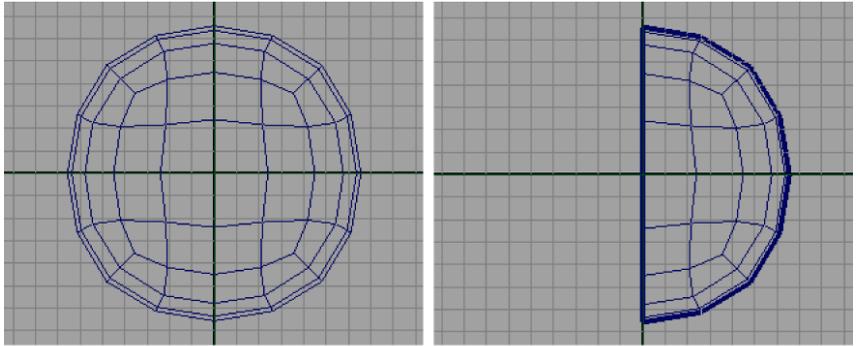
After setting the shading for the objects in the various views, you may wish to adjust the Alpha Gain setting on your image planes to change their transparency. As you gain experience with Maya, you'll develop your own personal preferences for modeling in wireframe or shaded mode and switching between the various shading modes.

Model symmetry

Whenever you model an object, you should take advantage of any symmetry that the form provides. It can save you time and effort if you work on only one half of the model and then later copy it across the model's axis of symmetry. The sketch of the helmet is symmetrical along its center line (YZ plane).

To delete the faces on the left half of the mesh

- 1 In the top view, right-click on the mesh and select Face from the marking menu that appears.
- 2 Drag a bounding box around the faces on the left half of the helmet mesh to select them.
- 3 Press the delete key.
The selected faces are deleted.



Delete the left half of the cube primitive by selecting those faces from the top orthographic view.

You now have half as many components in your model. When you near the completion point of the model, you will copy the finished half across the axis of symmetry to get the complete model.

Selecting components by painting

You can delete some of the faces that are not needed on the front and lower section of the helmet mesh using the Paint Selection Tool. This is useful when you need to select components that are in an irregular region and are not easily selected using the bounding box.

To select faces for deletion using the Paint Selection Tool

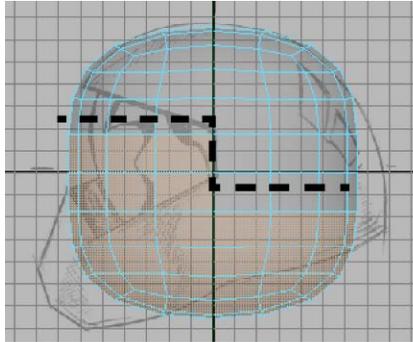
- 1 In the Toolbox, click the Paint Selection Tool icon.



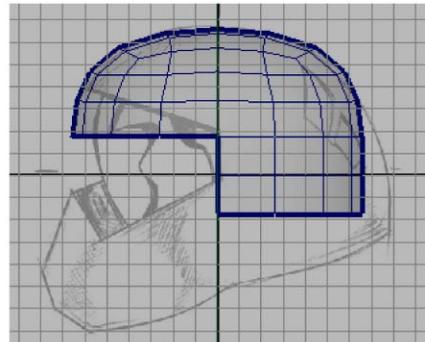
Click the Paint Selection Tool icon and paint select faces using Face selection mode

- 2 In the side view, right-click the helmet mesh and select Face from the marking menu that appears.
- 3 In the side view, paint select the faces that appear along the front and lower areas of the mesh (see image) by dragging the mouse over the faces.

NOTE The Paint Selection Tool only selects components on the mesh that face towards the camera.



Paint select the faces along the front and lower half of the mesh.



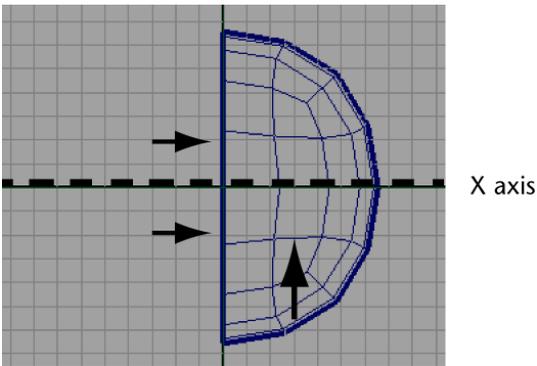
Delete the selected faces.

- 4 Press the delete key to delete the selected faces.
- 5 Tumble the perspective view to review your work so far.

The edges that reside along the outside perimeter of a polygon mesh, called *border edges*, are displayed with a thicker line than the interior edges on the mesh. You set this display setting early in the lesson so you could differentiate this edge type from other edges.

Selecting edge loops

In the top view, the rows of edges immediately above and below the X axis, and the row to the right of the Z axis are not straight. You'll want to align the edges in these rows (referred to as *edge loops*) by snapping them to the grid using the Move Tool. An *edge loop* is a path of polygon edges that are connected in sequence by their shared vertices. Edge loop selections are very useful when you model using polygons.



To align edges on the helmet mesh

- 1 Right-click the mesh and select Edge from the marking menu that appears. Maya's component selection mode is set to edges.
- 2 Double-click one edge in the edge-loop directly below the X axis. The edge loop is selected.
- 3 On the Status Line, turn on the Snap to Grids feature by clicking the icon.



Snap to Grids lets you move selected components to the closest grid line or grid intersection point. When many components are selected, you can align them to each other via the grid simultaneously.

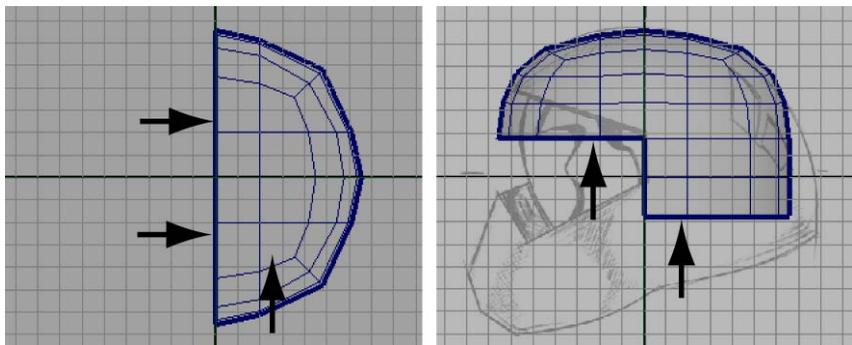
- 4 In the Toolbox, double-click the Move Tool to display its tool settings.
- 5 In the Move Snap Settings, turn off the Retain component spacing setting.



Double-click the Move Tool icon and in the Move Tool settings turn off Retain component spacing.



- 6 In the top view, click-drag the blue arrow on the Move Tool manipulator downwards a small distance.
The edge loop immediately snaps to a lower grid line. It is not positioned at the location you want, but all of the edges in the loop you selected are now aligned to each other.
- 7 On the Status Line, turn off the Snap to Grids feature by clicking its icon.
- 8 With the Move Tool still active in the top view, drag the manipulator arrow upwards a short distance so the edge loop is positioned roughly at its original location (which should be at the second grid line below the X axis).
- 9 Using the steps described above, align the other edge loop that appears above the X axis and then align the edge loop directly to the right of the Z axis. (Remember to turn on and off Snap to Grids as required.)
When you have finished, the edge loops nearest the X and Z axes are aligned. (see image below)



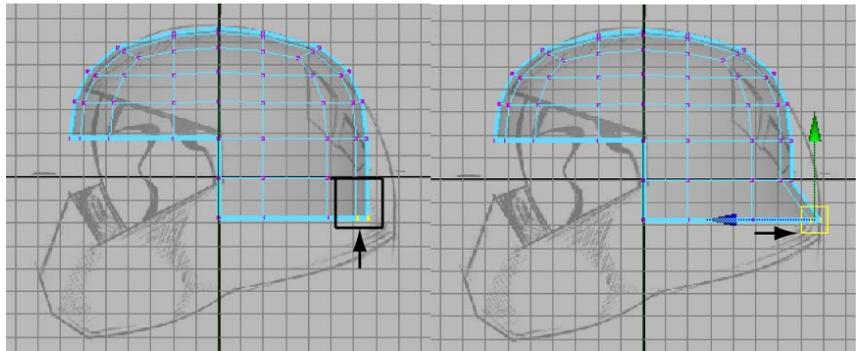
Edge loops are aligned to the grid.

Editing components in the orthographic views

Component selection and transformation is one fundamental method for editing the shape of a polygon mesh. As you model, you'll find yourself frequently examining and then refining the position of the polygon components (vertices, edges, and faces) while working in the various scene views so they match the reference images on the image planes.

To manually reposition the vertices on the rear of the helmet

- 1 Right-click the helmet and select Vertex from the marking menu that appears.
- 2 In the side view, select the pair of vertices at the rear lower edge (see image) by dragging a bounding box around them.
- 3 In the Toolbox, click the Move Tool.
- 4 In the side view, click-drag the blue arrow on the Move Tool manipulator towards the right until the vertices are repositioned so that your helmet matches the reference sketch on the image plane.



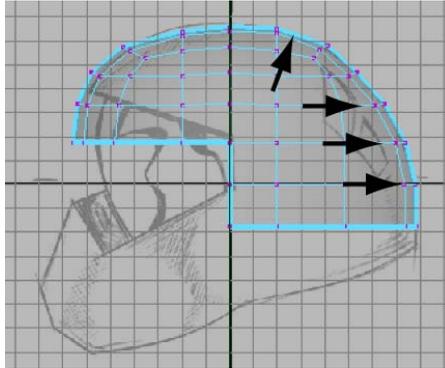
Select the pair of vertices at the bottom rear edge.

Move the vertices horizontally to match the reference sketch.

By moving the vertex as well as the vertex adjacent to it along the axis of symmetry, you ensure that the symmetrical shape of the helmet is maintained. If you move one vertex independently of the other it may result in an unwanted bump or valley in the mesh. These types of anomalies will become more apparent when you copy the completed half of the mesh across the axis of symmetry.

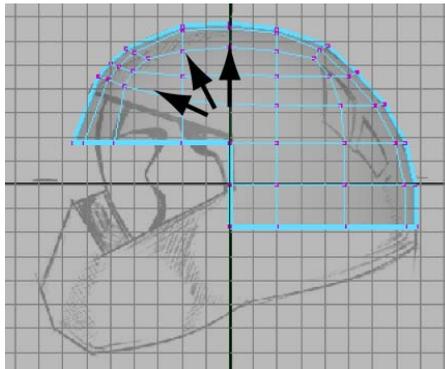
- 5 Repeat steps 2 through 4 for the other pairs of vertices on the rear of the helmet. Ensure that the edge loops appear smooth in relation to each other.

When you have finished, the back region of the helmet should closely match the reference image on your image plane.



- 6 In the side view, select other pairs of vertices along the top and front of the helmet and move them in a similar fashion so they match the reference image. Do not reposition the vertices for the top of the face shield yet.

TIP You can click in the center of the Move Tool manipulator to drag a vertex selection freely.

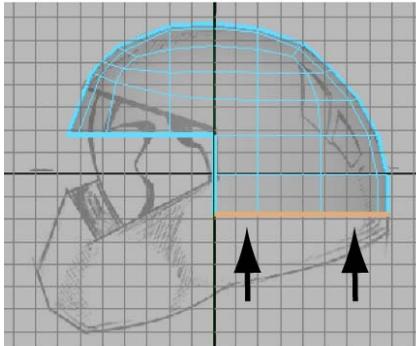


Reposition pairs of vertices along the top and front.

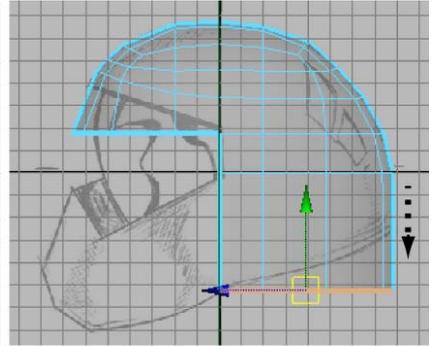
Next, you'll reposition the border edges that lie along the bottom edge of the mesh. You can select these edge types using the Select Border Edge Tool.

To reposition the lower border edges on the helmet

- 1 In the side view, select the lowest horizontal edge loop on the helmet by choosing Select > Select Border Edge Tool from the main menu, and then clicking the first and then the last edge on the loop as indicated in the image below.
- 2 Using the Move Tool, drag the selected edge loop downwards until the left hand vertex roughly matches the lower edge indicated in the reference sketch.

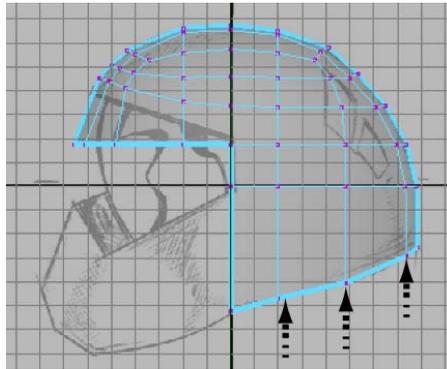


Select the border edges.



Move the border edges

- 3 In the side view, select and reposition the remaining vertices individually on the edge loop using the Move Tool so they match the reference sketch.

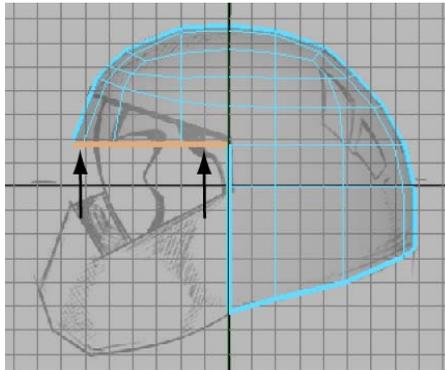


Move the vertices individually.

NOTE Up to this point in the lesson, you've been instructed to reposition the vertices on the helmet mesh only within the side view (Y, Z plane). Once you achieve the shape you want in the side view you will then concentrate on how the model appears when viewed from the front and perspective views.

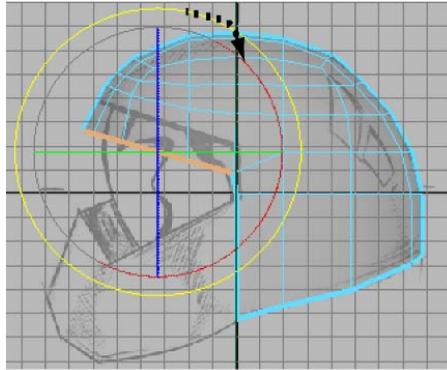
To edit the border edges on the upper edge of the face shield

- 1 From the Select menu, choose Select Border Edge Tool.
- 2 In the side view, click the first border edge that will be used as the upper edge of the face shield, then click the last border edge (see image).
The border edges in between are selected.



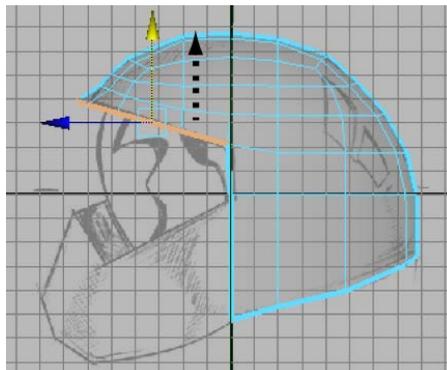
Select the border edges

- 3 In the Toolbox, select the Rotate Tool by clicking its icon.
- 4 In the side view, click-drag the rotate manipulator in a clockwise direction until the border edges are rotated at roughly the same angle as the corresponding edge in the reference sketch.



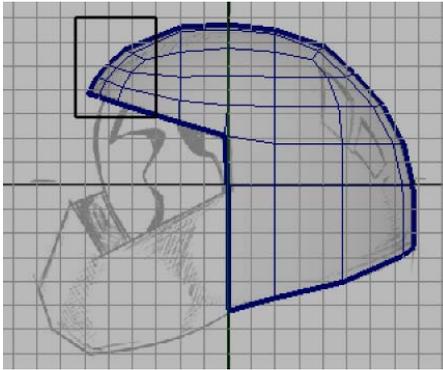
Rotate the border edges.

- 5 With the border edges still selected, click the Move Tool again and drag the green manipulator upwards to match the location of the border edges in the reference sketch.



Move the border edges.

- 6 Reposition the pairs of vertices on the upper front of the helmet to match the reference sketch.

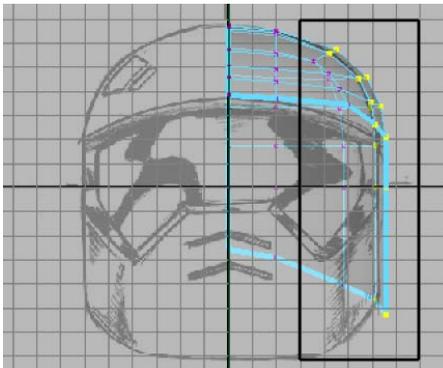


At this point in the lesson, the outline of your helmet should roughly match the helmet in the side view reference image. If it doesn't, review the earlier steps in this lesson and make any adjustments to your polygonal mesh as required.

If you view your helmet in the front orthographic view, you'll notice that the helmet shows a wider profile from this view than the reference sketch. In the next steps you'll correct this using the front and top orthographic views of the helmet for reference.

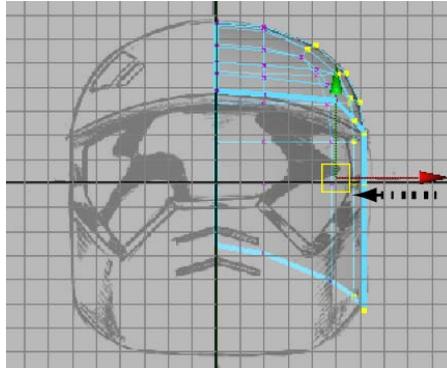
To reposition vertices on the side of the helmet to match the sketch

- 1 Display the front view.
- 2 In the front view, select the vertices that extend beyond the outline of the helmet as shown in the reference sketch (see image).



Select the side vertices that extend beyond the sketch.

- 3 In the front view, move these vertices to the left until they match the widest area of the helmet in the reference sketch (see image).



Move the side vertices
to match the sketch.

When you view your helmet from the top orthographic view the region between the side and rear of the helmet appears a bit flat in relation to the other areas. To correct this you can move the other vertices in this region outwards in a similar fashion so the curvature in this area appears fuller and more rounded. However, viewing only from the orthographic views can be limiting, and you should also use the perspective view to examine the mesh.

To examine the mesh using the perspective view

- 1 Enlarge the perspective view.
- 2 Dolly and tumble the perspective view while you closely examine the helmet mesh.

As you examine the vertices along any particular edge loop, the vertices on the mesh should appear to cascade in a smooth gradual fashion to create the curvature of the mesh with no undesirable spikes or dips.

Ensuring that the mesh appears relatively smooth at various stages throughout the modeling process will reduce the possibility for issues when you create a high resolution version of the mesh later on.

If you find areas where one vertex (or more) appears to protrude outwards (or recedes) on the mesh in relation to neighboring vertices, you can correct these protruding regions by repositioning the affected vertices in the perspective view.

Editing components in the perspective view

Repositioning polygon components in the perspective view can be more challenging when compared to the orthographic views because your frame of reference changes as you track, dolly, or tumble the 3D view.

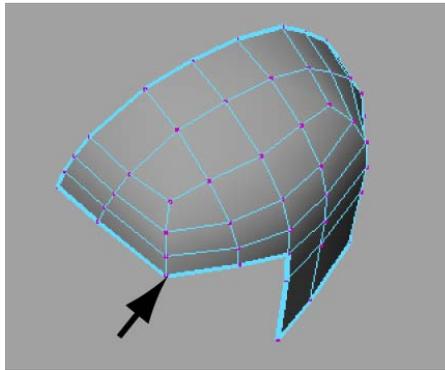
By default, the Move Tool lets you reposition components in relation to *world space coordinates*. That is, the movement of a component is referenced to a direction based on the center of the 3D scene and the X, Y, or Z axes.

It is also possible to set the Move Tool to move objects and components based on other coordinate systems, such as *object space* and *local space*.

In addition, you can also move a polygon component in relation to its surrounding mesh. For example, you can select and move a vertex in a direction that is perpendicular or *normal* to its surrounding surface mesh. This is a useful technique for correcting any protruding vertices on the mesh.

To move a vertex on the mesh in a direction normal to the mesh

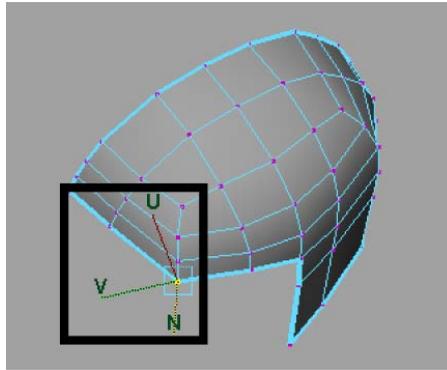
- 1 Dolly and tumble the perspective view while examining the mesh until you find vertices that protrude outwards from the mesh in an unwanted fashion.



- 2 In the Toolbox, double-click the Move Tool to display the Move Settings editor.
- 3 In the Move Settings editor, in the Move section, turn on the Normal option.

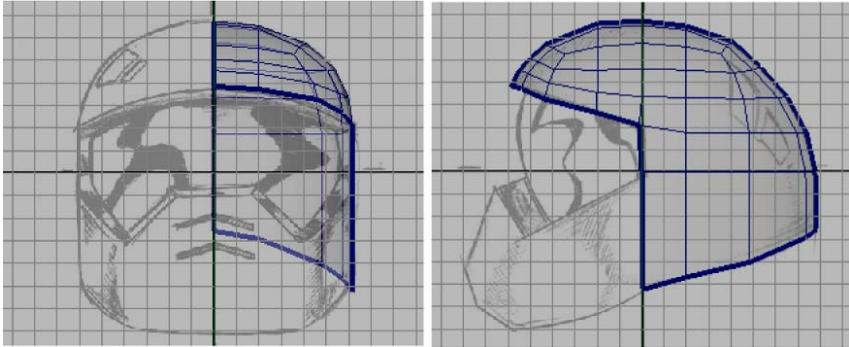


- 4 In the perspective view, click the vertex you wish to reposition. The vertex highlights and the Move manipulator appears to indicate the three directions of movement that are possible in this mode.



- The U and V handles slide the vertex in relation to its associated edges, while the N handle moves the vertex either away or towards the mesh depending on the direction you move your mouse.
- 5 Drag the N handle outwards or inwards depending on what's required for that particular vertex. Dolly or tumble the view as required to examine the mesh after you've done this and make any modifications as necessary.
 - 6 Repeat the above steps for any other protruding or receding vertices on the mesh until you are satisfied that the mesh appears smooth.
 - 7 Before you proceed to the next section, double-click on the Move Tool and return the Move setting to the default World setting.

If you examine your model in the perspective view your helmet mesh should now match the reference sketches as they appear in both the front and side image planes (see image below). The mesh should also have a relatively uniform distribution of polygon faces on the mesh and the edge loops should flow smoothly.



Drawing a polygon

To create the lower front region of the helmet (that is, the region that would protect the mouth and jaw of the wearer) you'll create a polygon for the cross section of the lower front region by manually placing vertices using the Create Polygon Tool (Mesh > Create Polygon Tool).

To place vertices for a polygon

- 1 In the side view, select Mesh > Create Polygon Tool > .

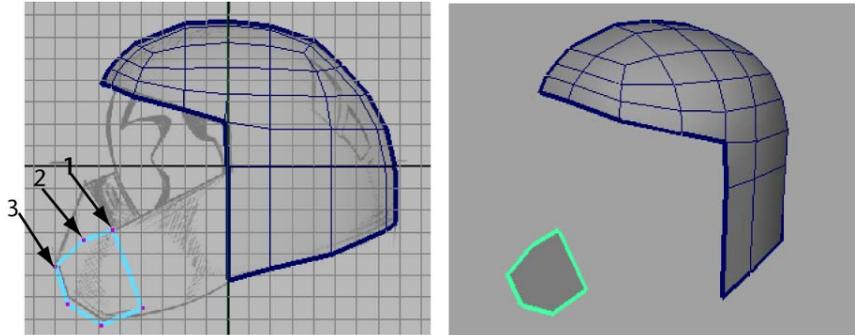
The Create Polygon Tool settings editor appears.

- 2 In the tool settings editor, set the following options:

- Limit the number of points: On
- Limit points to: 6

These settings specify that a closed polygon face will be created once you have placed six vertices in the scene.

- 3 In the side view, place six points for the profile as it appears in cross section of the lower front region of the helmet (see image below). Ensure that you place the vertices in a counter-clockwise direction for this step. The new polygon face is created along the YZ plane (the axis of symmetry for the model).



Place six vertices to create the polygon face.

- 4 Press the q key to quit the Create Polygon Tool when the polygon face is complete.

The new polygon component remains selected in Object Mode.

NOTE If the polygon face doesn't appear shaded in the side view but does appear shaded in the perspective view, it means that the polygon face and the image plane lie on exactly the same plane. You can correct this by moving the image plane backwards in the side view so it doesn't lie directly on the YZ plane, but this step is not critical to the success of the steps that follow.

You'll extrude this polygon face to create the remainder of the lower front region of the helmet. Before you can extrude the new face it needs to be selected in Face mode. You can do this quickly by converting the existing selection to Faces.

To convert the object selection to face selection

- With the face still active in Object Mode, choose Select > Convert Selection > To Faces.

The selection is converted to a face selection.

TIP You can also press Ctrl + F11 or Ctrl + right-click on the selected object and choose To Faces > To Faces from the marking menu that appears.

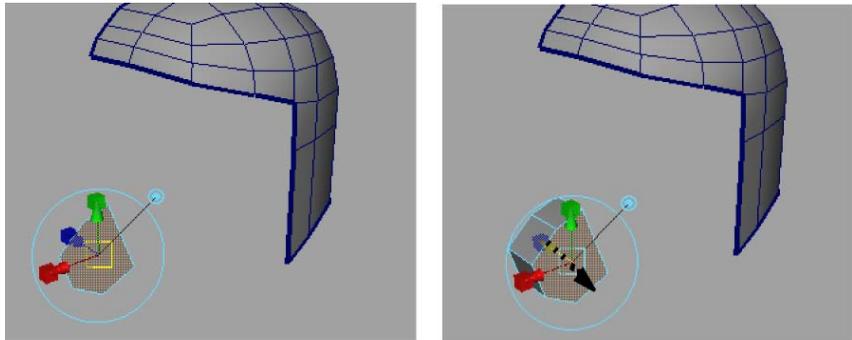
Extruding polygon components

You can create new polygon components from existing ones using the Extrude feature (Edit Mesh > Extrude). When you extrude a polygon component (for example, a face, edge, or vertex), you create additional polygon components from the ones you selected. Using the Extrude feature you will:

- Create the lower front region of the helmet by extruding the polygon face you created in the last section of the lesson.
- Extrude edges around the face shield and along the lower bottom edge.

To extrude the polygon face for the lower front region

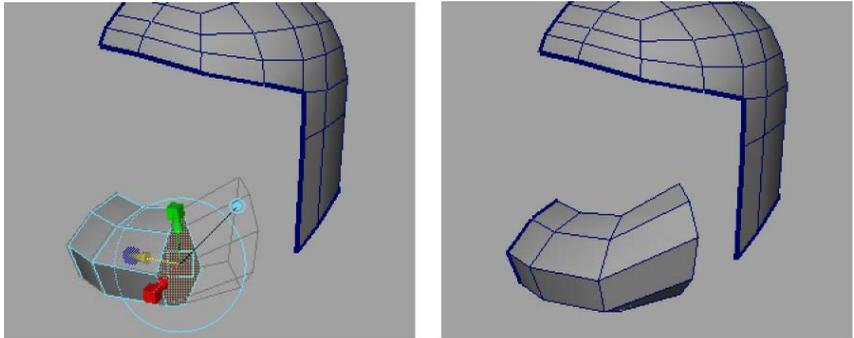
- 1 Enlarge the scene view to a single perspective view.
- 2 With the polygon face still selected, choose Edit Mesh > Extrude. The extrude manipulator appears on the selected face.
- 3 In the perspective view, drag the blue arrow on the manipulator to extrude a section of mesh out from the face (positive X) a distance of approximately one half a grid unit (see image).



Drag the blue arrow to extrude the selected face.

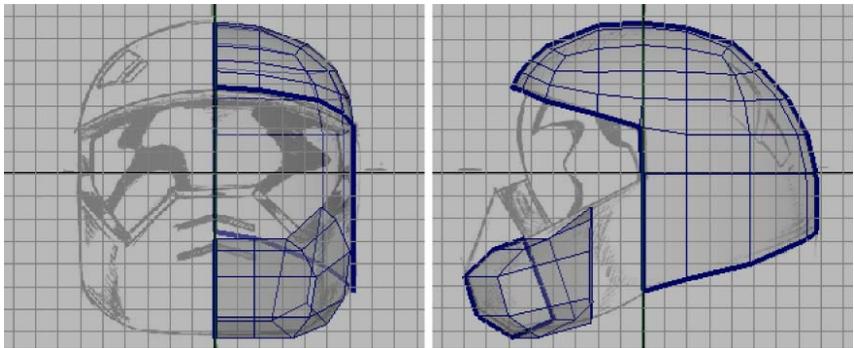
- 4 Press the g key to extrude again.
- 5 Click the large circle that surrounds the manipulator to display the rotate manipulators and then drag the green circular manipulator to rotate the angle of the extrusion to match the angle in the reference sketch (see image) and then drag the arrow manipulator a second time to extrude a second section of mesh.

TIP As you change the angle of rotation and extrude the mesh you can also momentarily switch between the move or scale manipulators to fine tune the position and scale of each section as you extrude it.



Extrude three segments, changing the extrude angle as you proceed.

- 6 Press the g key once again and create a third extruded region using the manipulator to move, rotate, or scale the extruded segment of the mesh so you position it correctly when compared to the reference sketch (see image). You may also want to view the extrusion from either the top or front view to ensure your extrusion doesn't extend outwards more than the side region.

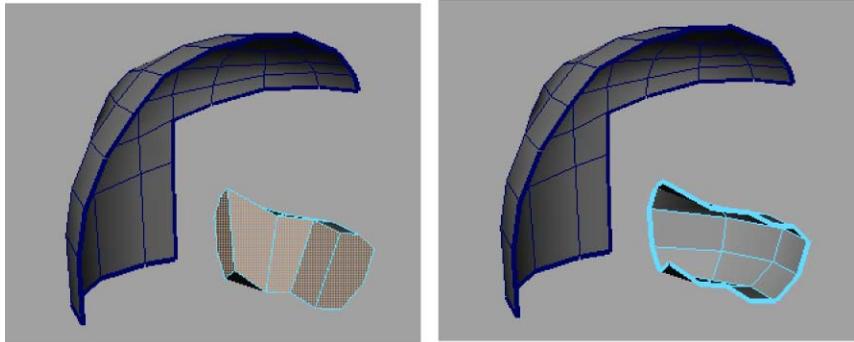


- 7 Save your work.

To delete unwanted faces on the extruded mesh

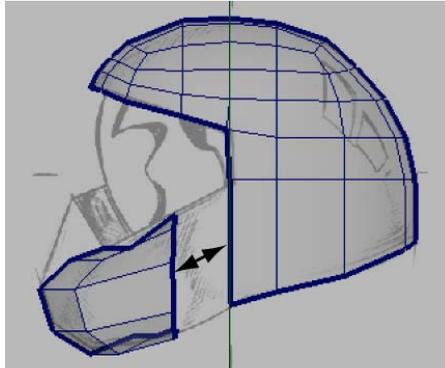
- 1 Tumble the perspective view until you can view the inside of the lower front region (see image below).

- 2 Select the faces that appear on the inside of the mesh you just extruded, including the faces on either end of the extrusion. These faces were required for creating the extruded portions of the lower region but are not needed beyond this point.
- 3 Press the Delete key to delete the selected faces.



Select and delete the interior faces.

When you are finished, a gap will exist between the last extruded segment of the lower region and the helmet mesh.

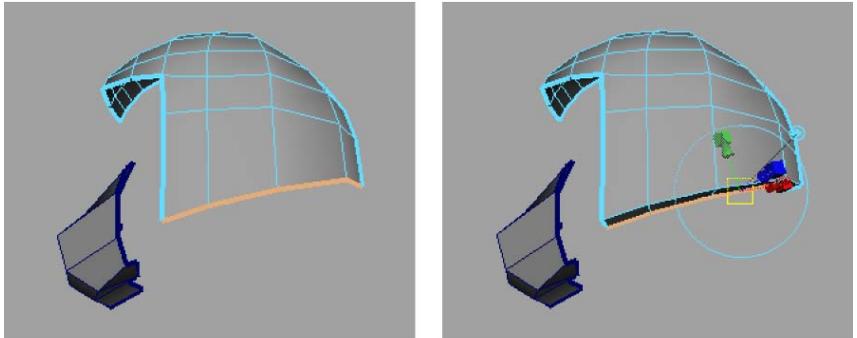


A gap exists between the two meshes.

You'll combine these separate meshes together and then create a mesh that bridges between them in the next section of the lesson. To prepare for the bridge, you need to extrude more edges on the helmet mesh so that the number of edges match when you create the bridge.

To extrude the bottom edges of the helmet mesh

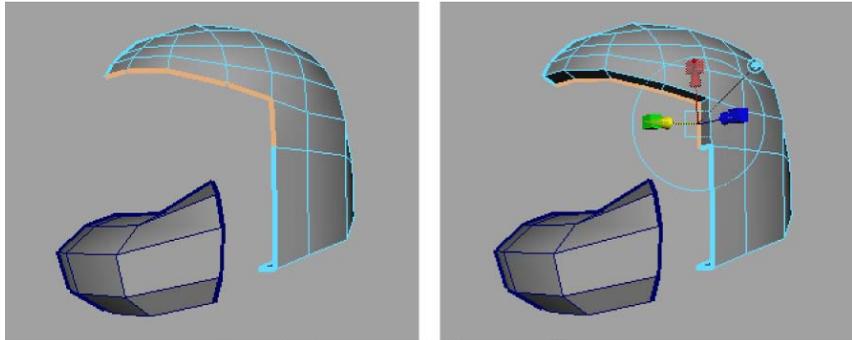
- 1 In the perspective view, select the lower edges of the helmet mesh using Select > Select Border Edge Tool.
- 2 Select Edit Mesh > Extrude, then drag the blue arrow manipulator in a direction towards the inside of the helmet to create a row of edges that are perpendicular to the selected bottom edges. Extrude these edges a distance that is approximately one grid unit in depth (see image below).



Extrude the top and side edges inwards.

To extrude top and side edges for the face shield

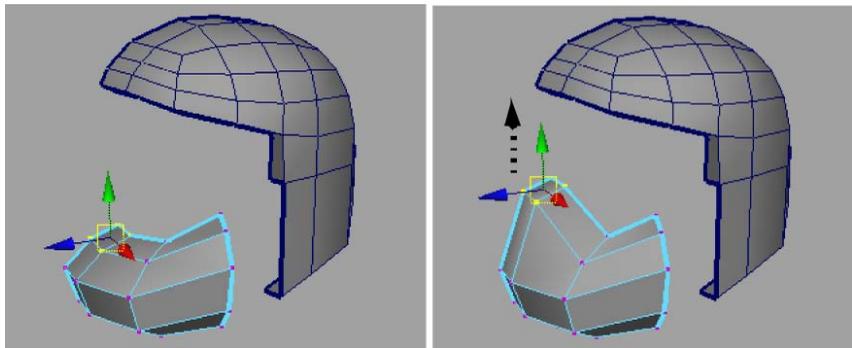
- 1 In the perspective view, select the upper and side edges on the helmet mesh using Select > Select Border Edge Tool (see image below).
- 2 Select Edit Mesh > Extrude and then drag the blue arrow on the extrude manipulator in a direction towards the inside of the helmet to create edges that are perpendicular to the top and side edges of the face shield. Extrude these edges a distance that is approximately one grid unit in depth (see image below).



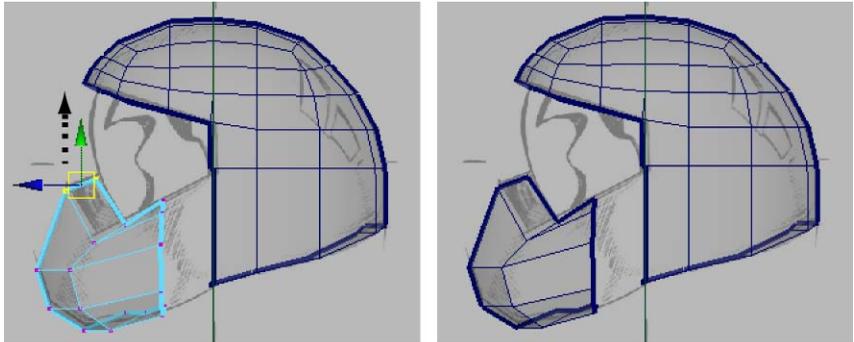
Extrude the top and side edges inwards.

To move vertices on the lower front region to match the reference sketch

- 1 In the perspective view, right-click on the helmet and select vertex mode to change the selection type to vertices.
- 2 Select the four vertices on the lower front region of the helmet that are near the axis of symmetry (see image below) and using the Move Tool, move the selected vertices upwards by dragging the green arrow on the Move Tool manipulator so that the lower front region of the helmet matches the reference sketches as they appear in the various orthographic views.



- 3 Adjust any vertices on the lower region that may require minor repositioning by selecting and moving them as necessary.



Select and move the vertices upwards to match the reference sketch.

Bridging between edges

Next you'll connect the lower front region to the helmet mesh. You can create meshes that bridge between one or more border edges of a mesh using the Bridge feature (Edit Mesh > Bridge).

When using the bridge feature you must ensure that:

- the edges to be bridged are in the same polygon mesh. That is, you must combine the two meshes into one using the Combine feature, before you perform the bridge.
- you select an equal number of border edges on either side of the region to be bridged.

To combine the two meshes into one

- 1 In the perspective view, right-click the mesh and select Object Mode from the marking menu that appears.
- 2 Select the helmet mesh, and then shift-select the lower front mesh, so both meshes are selected.
- 3 Select Mesh > Combine.

The two meshes are combined into one. When you select either object now, the other is also selected because they are in the same mesh.

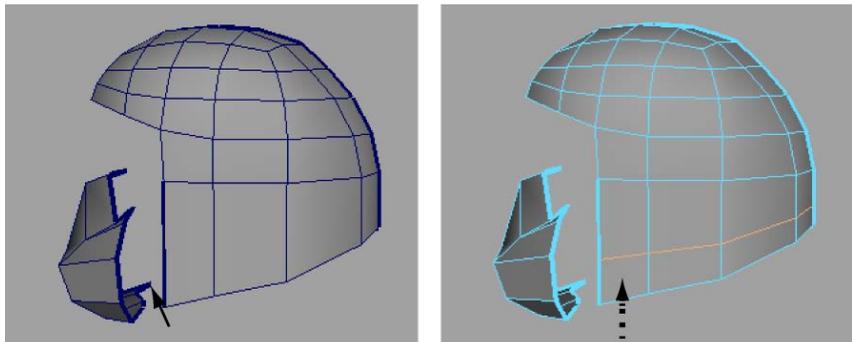
The edge of the mesh on the lower front region of the helmet is comprised of five edges. The corresponding region on the side region of the helmet only has three. You can increase the number of edges on the lower side region by

inserting two edge loops across the mesh. Inserting two edge loops in this region of the mesh also divides the large faces so they better match the size of the other faces on the rest of the mesh.

To insert edge loops on the side region of the helmet

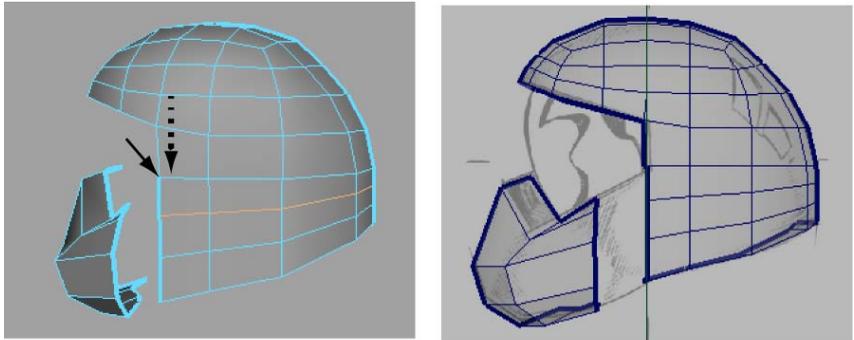
- 1 Select Edit Mesh > Insert Edge Loop Tool >  .
The Insert Edge Loop Tool settings editor appears.
- 2 In the tool settings editor, set the following options:
 - Maintain Position: Relative distance from edge
- 3 In the perspective view, click-drag the edge on the side region of the helmet near the bottom of the edge as indicated in the image below. Without releasing the mouse button, drag the mouse upwards about one third of the distance along the edge and then release the mouse button to insert the edge loop.

NOTE Click-dragging an edge when using the Relative option inserts an edge loop that closely matches the existing edge layout on the mesh. That is, the lower region of the mesh is much wider near the front than at the rear. The Relative setting adjusts the position of the edge loop locator based on this topology so is ideal in these situations. When you click-drag using the Relative option, remember to click near the edge whose layout you want the edge loop to match.



Click-drag the side edge near the bottom to insert the first edge loop.

- 4 With the Insert Edge Loop Tool still active, click-drag the side region of the helmet near the top of the same edge and insert a second edge loop about two thirds of the distance along the edge.



Click-drag the side edge near the top to insert the second edge loop.

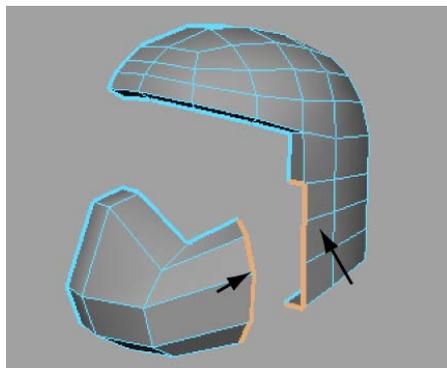
- 5 Press the q key to return to selection mode, and click anywhere off the mesh to unselect the edges.

Now that you've inserted the necessary edges, you can proceed with creating the bridge.

To bridge between the lower front and side region of the helmet

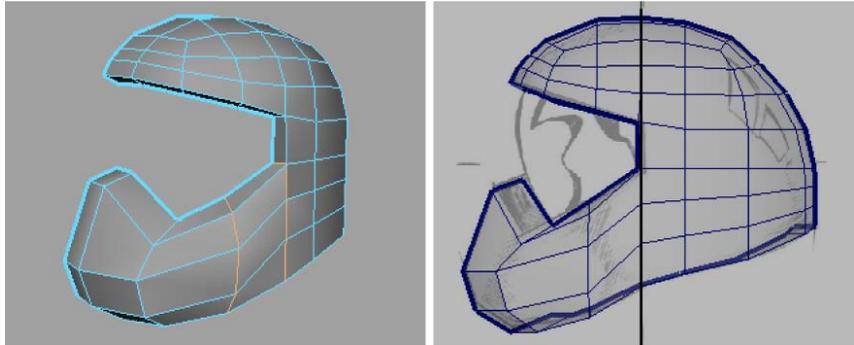
- 1 Choose Select > Select Border Edge Tool and then click the border edges on both the lower front as well as the side region of the mesh where you want the bridging mesh to be constructed. (You should have five edges selected on either side).

TIP When you select border edges using the Select Border Edge Tool you can click on the first and last edge in a series on one side of the bridge and the tool will select the edges in between.



Use the Select Border Edge Tool to select the edges for the bridge.

- 2 Select Edit Mesh > Bridge >  .
- 3 In the Bridge Options window, set the Divisions to 0, then click the Bridge button to create the bridge.



- 4 Press the q key to return to select mode, and click anywhere off the mesh to unselect the edges.

NOTE If your bridge appears to twist or cross over itself it indicates that the two meshes have their surface normals mismatched. In this particular case it likely indicates that you did not create the original profile shape for the lower front by placing the six vertices in a counter-clockwise direction.

If this occurs you must undo your steps to the point immediately before you combined the two meshes and then reverse the surface normals on only the lower front section by selecting it and then selecting Normals > Reverse. You can then redo the instructions in this section as required.

Adding polygons to a mesh

For the protective face shield on the helmet, you'll create one large multi-sided polygon using the Append to Polygon Tool and then split it into multiple four-sided polygons (quads) so that the new polygons match the existing four-sided topology of the helmet mesh.

To create the face shield using the Append to Polygon Tool

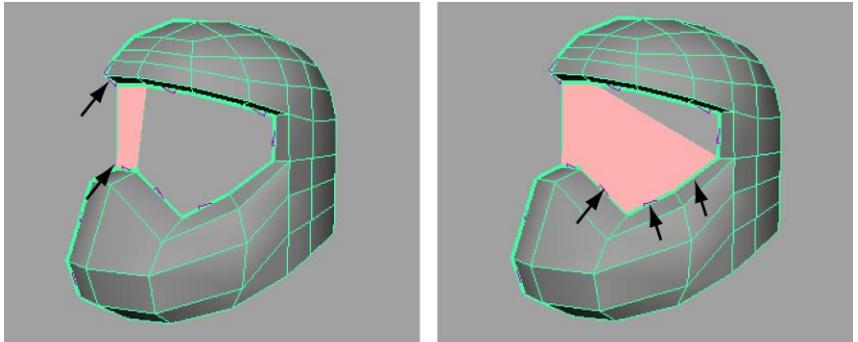
- 1 Tumble the perspective view so you can see all of the edges surrounding the area for the face shield.

- 2 Select Edit Mesh > Append to Polygon Tool > .
- 3 In the Append to Polygon Tool settings window, set the following:
 - Keep new faces planar: Off

If the planar option is on, the Append to Polygon Tool will not create a multi-sided polygon as the edges you'll select do not form a planar region.

- 4 In the perspective view, click once on the top inner border edge of the helmet mesh to indicate which mesh you are appending to, and then click the same edge a second time to begin the append operation (see image below).
- 5 Click the bottom edge of the face shield that is opposite the first edge you just clicked.

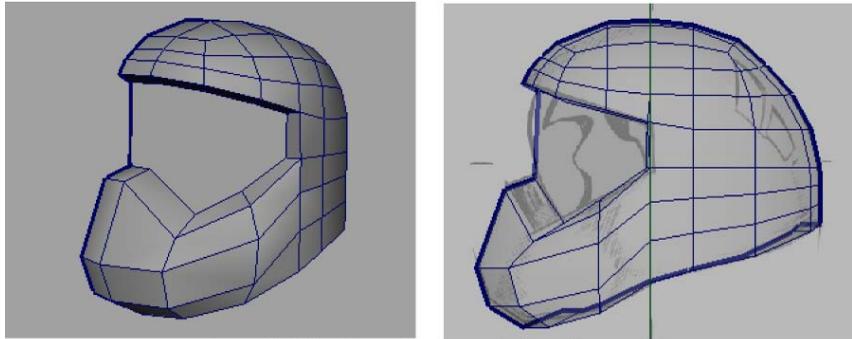
A polygon appears that spans the two edges.



Click the edges in a counter-clockwise direction to append a polygon face onto the mesh.

- 6 Continue to click the remaining border edges in a counter-clockwise direction around the opening for the face shield (see image above) until you reach the last edge.
- 7 When you are finished clicking the perimeter border edges for the face shield, press the q key to quit the tool.

The multi-sided polygon is completed and added to the existing mesh. Multi-sided polygons are also referred to as *n-gons*.



A multi-sided face is appended to the mesh.

Ideally, all faces should be four-sided to match the other faces on the helmet mesh. In the next section, you'll split the n-gon vertically and horizontally into several smaller four-sided polygons using the Split Polygon Tool.

Splitting polygon faces

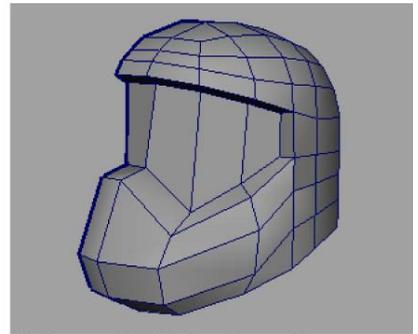
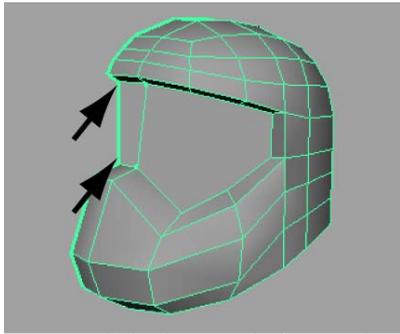
Earlier in the lesson you split the helmet mesh by inserting edge loops across the mesh. You can split localized areas of a mesh using the Split Polygon Tool. When using the tool you draw a line across the faces to indicate the location for the split. You'll begin by vertically splitting the face shield.

To split the face shield vertically

- 1 Select Edit Mesh > Split Polygon Tool >  .
The Split Polygon Tool settings editor appears.
- 2 Set the following options:
 - Split only from edges: On
 - Use snapping points along the edge: On
 - Snapping tolerance: 100

These settings let you begin and end your split at a location that corresponds exactly with an edge and lets you snap to the midpoint and ends of the edge you select. (The number of points setting of 1 ensures this). These settings will help to ensure that the faces you split align exactly and that the four-sided topology of the mesh is maintained.

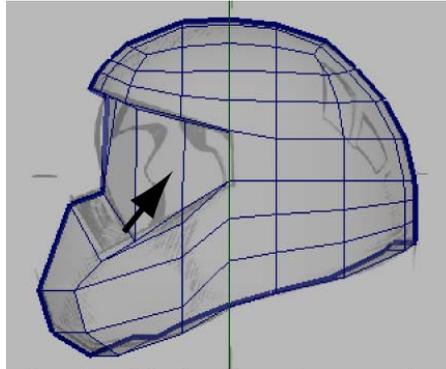
- 3 Tumble the camera in the perspective view so you can view both the upper inner edges of the face shield as well as the lower inner edges.
- 4 Click-drag the top inner edge of the face shield to indicate the start of the split (see image below). Drag the mouse to position the vertex until it stops at the right side of the edge.
- 5 Click-drag on the lower inner edge of the helmet mesh to indicate the end of the split (see image). Drag the mouse to the right until the vertex stops at the right side of the edge.
- 6 Press the y key to split the face.



Click-drag on the inner edges to split the multi-sided polygon face.

- 7 Press the g key to select the Split Polygon Tool again, and then continue to split the face shield vertically at the other locations specified in the image below. Remember to press the y key after each split and then the g key to select the tool again.
When you are finished the n-gon for the face shield will be split into four or five new polygons (depending on how you've constructed your mesh).

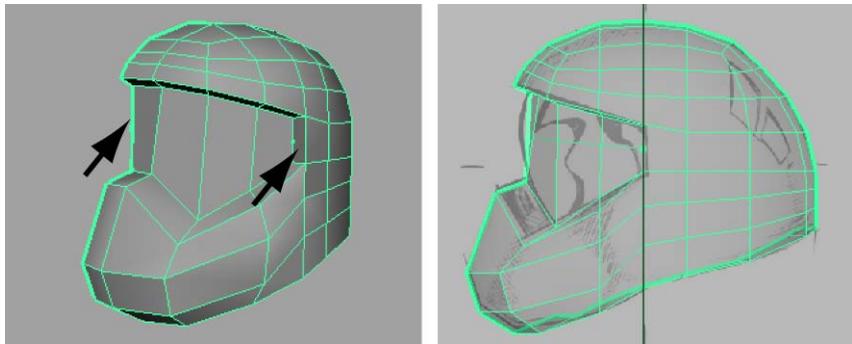
One polygon on the side of the face shield will still be five-sided. To correct this, you'll also split the face shield horizontally. Splitting the face shield horizontally will also let you modify its shape afterwards.



One five-sided polygon remains.

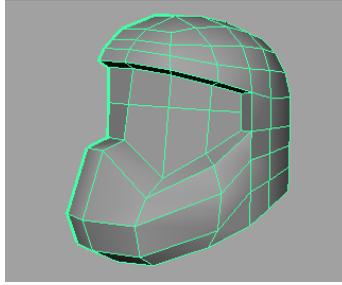
To split the face shield horizontally

- 1 In the perspective view, with the Split Polygon Tool still active, click-drag the inner side edge (see image) to indicate the start location for the horizontal split.
- 2 In the side view, click-drag the front vertical border edge of the face shield (this border edge lies on the axis of symmetry) and release the mouse button at the mid-point along the edge where the vertex naturally snaps (as if a magnet were attracting it towards that location).



Click-drag the inner side edge then the front vertical edge to split across the faces.

- 3 Press the y key to split the faces.



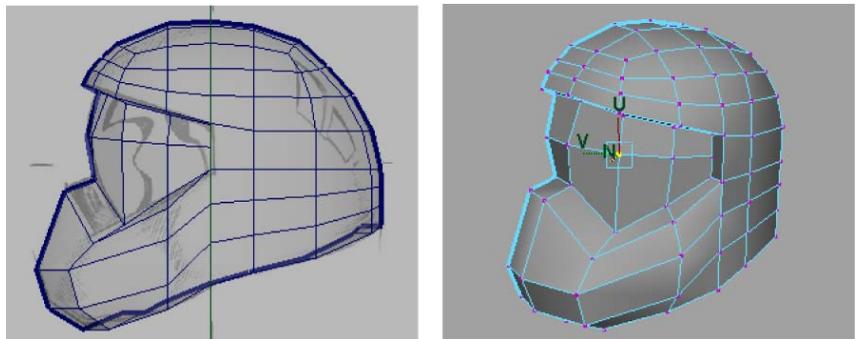
NOTE When you split across multiple faces at the same time you only need to click an edge to indicate the start point for the split and on a second edge to indicate the end point. The Split Polygon Tool automatically splits the edges in between.

- 4 Press the q key to quit the Split Polygon Tool.

In the next steps you'll reposition some of the vertices along the horizontal split to make the face shield protrude outwards a small amount.

To adjust the shape of the face shield

- 1 In the side view, select the middle two vertices at the front of the face shield and use the Move Tool to move the vertices outwards (+Z) a small amount (see image).
- 2 In the perspective view, select the remaining middle vertices on the face shield one at a time and reposition them outwards a small distance using the Move Tool with the Move Setting option set to Normal.



Select and move the middle vertices on the face shield so it appears to protrude outwards a small amount.

TIP Tumble the perspective view so you can see the relationship between the vertices as you move them outwards.

- 3 Reset the Move Tool's Move Settings to World before continuing to the next steps.

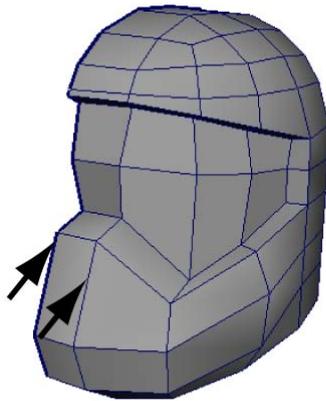
To create the diagonal grill vents on the lower front of the helmet you'll insert edges on the face, reposition some of the vertices, and then extrude some of the faces.

To insert multiple edges for the diagonal grill vents

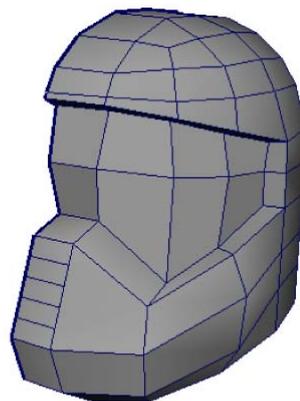
- 1 Select Edit Mesh > Insert Edge Loop Tool > .
- 2 In the Insert Edge Loop Tool Options window, set the following:
 - Multiple Edge Loops: On
 - Number of edge loops: 4
 - Auto Complete: Off

These settings let you insert four evenly spaced edges on the face where you want the grill vents to appear.

- 3 In the perspective view, click the border edge of the face where the grill vents will appear and then click the edge directly opposite that border edge (see image).



Select the border edge then the edge opposite it.



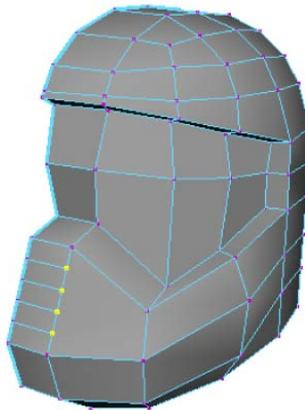
Four edges are inserted on the face.

- 4 Press the y key to complete the edge loop insertion
- 5 Click off the mesh to deselect the edges.
- 6 Press the q key to exit the tool and return to selection mode.

To make the grill vents appear diagonally you'll select the vertices on the right side and then slide them downwards using the Move Tool.

To move vertices along an edge using the Move Tool

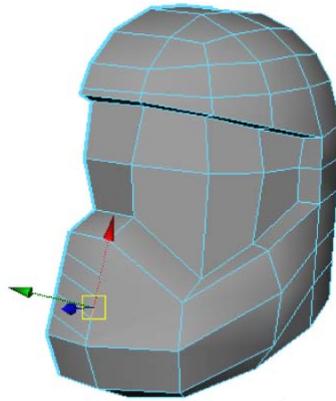
- 1 Right-click the helmet mesh and change the selection mode to Vertex.
- 2 Select the vertices on the right side of the grill feature (see image below).



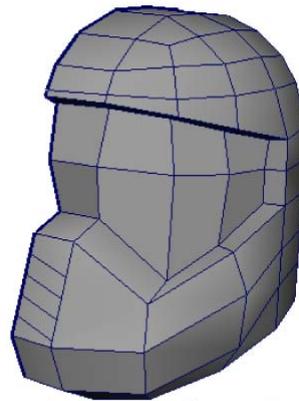
Select the vertices on the right side of the grill feature.

- 3 Double-click the Move Tool icon to display the Move Tool's settings editor.
- 4 In the Move Tools' settings editor, click the Set to Edge button.
The vertices appear unselected temporarily. The Move Tool expects you to select an edge it will reference for the axis of movement.
- 5 Click an edge that is on the same line of the edges as the vertices you selected.
The Move Tool manipulator appears and is aligned to the edge you selected. The vertices appear selected again indicating that the Move Tool is now set to move those vertices along the axis defined by the edge you selected.

- 6 Drag the red arrow on the Move manipulator downwards to move the vertices so that the shape of the faces for the grill are more diagonal (see image).



Drag the red arrow on the manipulator downwards.



The edges appear diagonal as a result.

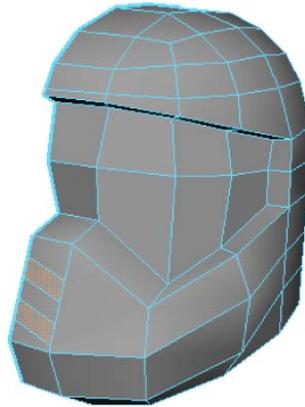
NOTE Make sure you do not move the vertices so that the lowest vertex touches the corner vertices or you'll create an edge that has zero length.

- 7 Click off the mesh to deselect the vertices.
- 8 Before proceeding to the next section, double-click the Move Tool and reset the tool settings by clicking the Reset Tool button.

To make the grill vents three-dimensional, you'll extrude some of the thin diagonal faces inwards.

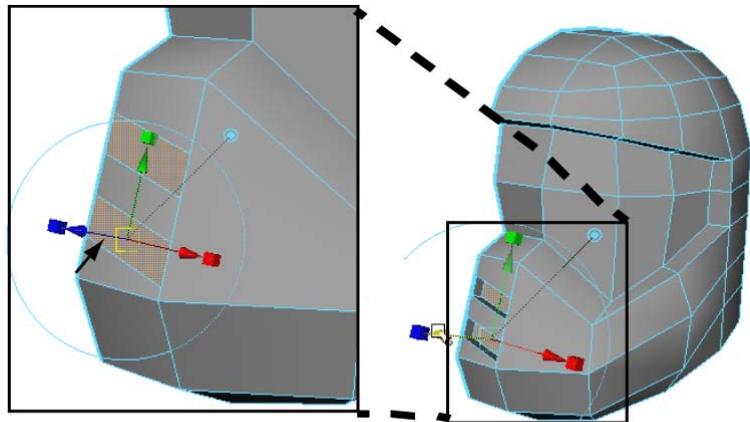
To extrude the faces for the grill feature

- 1 Right-click the helmet mesh and change the selection mode to Face.
- 2 Beginning at the bottom of the grill feature, shift-select the two diagonal faces as indicated in the image below.



Shift-select the thin diagonal faces.

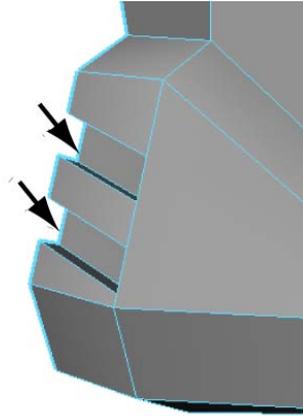
- 3 Select Edit Mesh > Extrude.
The Extrude manipulator appears.
- 4 Drag the blue arrow on the Extrude manipulator towards the helmet a short distance to create the two recessed vents for the grill.



Drag the blue arrow on the extrude manipulator towards the helmet to create the two recessed vents for the grill.

- 5 Press the q key to quit the Extrude feature.

- 6 Shift-select the two side faces on the grill vents that lie on the axis of symmetry and delete them (see image below). These faces will not be required when you create the opposite half of the helmet.



Select and delete the two small faces on the side.

- 7 Save your work before proceeding to the next section.

Terminating edge loops

When you inserted the edges for the grill vents the adjacent face was changed from four-sided to eight-sided as a result of the splitting that occurred along the shared edge.

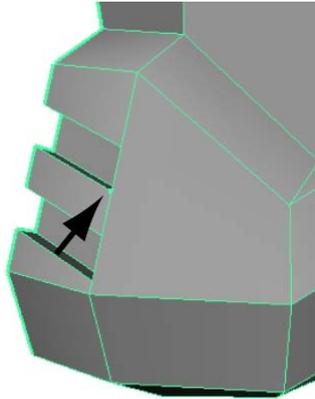
This situation can easily occur when you are splitting polygons in a localized region of a mesh, and should be anticipated if maintaining four-sided polygons is a constraint for your polygon models.

When this occurs you'll need to find a way to gradually divide the n-gons into four-sided regions by *terminating the edge loops*. Terminating edge loops is when you split a multi-sided n-gon gradually into multiple four-sided polygons.

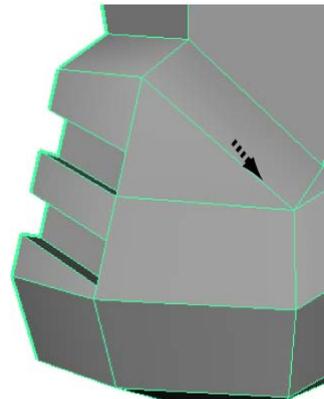
When you do this, a single vertex may have five or more shared edges coming out of it as a result. This process lets you create regions on a mesh that have more detail while maintaining a particular mesh type—four-sided polygons in this case.

To manually split the multi-sided polygon into three four-sided polygons

- 1 Select Edit Mesh > Split Polygon Tool.
- 2 In the perspective view, click-drag on the top side edge of the top grill vent. Drag your mouse so the vertex is positioned at the bottom of the edge (see image).

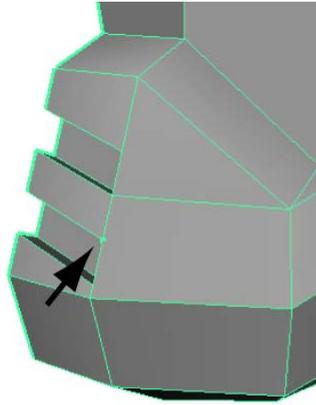


Click-drag on the top side edge of the top grill vent.

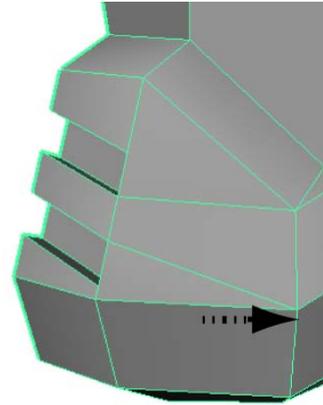


Click-drag on the top edge of the n-gon.

- 3 Click-drag on the top edge of the multi-sided polygon and slide the vertex to the right end of the edge (see image).
- 4 Press the y key to split the face.
The region above the split edge appears like a polygon but is actually four-sided because of the edges on the grill vent.
- 5 Split the side edge on the bottom grill vent in a similar fashion by click-dragging your mouse so the vertex is positioned at the top of the edge (see image below).



Click-drag along the top side edge of the bottom grill vent.



Click-drag along the bottom edge of the n-gon.

- 6 Click-drag the bottom edge of the multi-sided polygon and slide the vertex to the right end of the edge (see image).
- 7 Press the y key to split the face.
Your multi-sided face is now split into three four-sided faces, maintaining the overall quad topology on the helmet mesh.
- 8 Press the q key to exit the Split Polygon Tool.

Deleting construction history

Maya keeps track of the options, attribute settings, and transformations made to an object via its construction history. Construction history is useful when you're working on a surface and want to edit an attribute from an earlier stage of the model's development.

The simplest way to access the construction history for an object is via its various input nodes in the Channel Box or via the Attribute Editor. Selecting an input node displays the attributes associated with that particular modification to the object. You then edit the attributes on that node as required and the surface updates.

For example, the input nodes for the helmet mesh include: `polyCreateFace`, `polyCube`, `polyExtrudeFace`, `polySmoothFace`, `polyMoveVertex`, `polyUnite`, and so on. You can associate the steps in creating the helmet mesh with nodes that are related to a particular stage of its construction history.

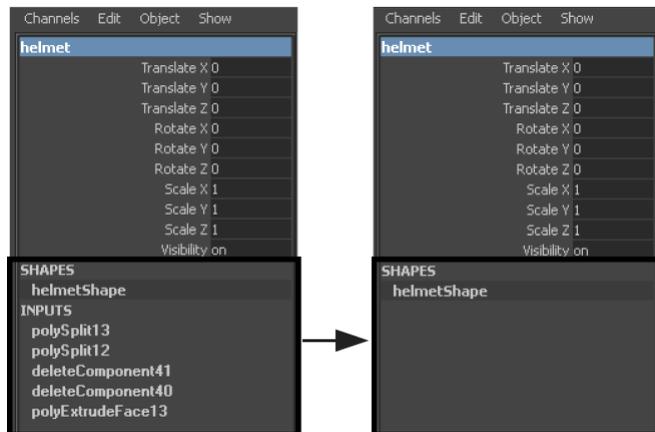
As you near the completion of a model, these construction history nodes are not required. If your production work entails having another person texture or animate the model, you'll want to delete these history nodes from the objects in your scene altogether so another user doesn't accidentally change your work.

Deleting the construction history is usually done only when a model is completed to a particular milestone stage and the next major phase of production with it needs to occur. For example, skinning, animating, texturing, and so on.

To delete the construction history on the helmet mesh

- 1 Right-click the mesh and change the selection mode to Object, and select the helmet mesh.
- 2 Display the Channel Box if it isn't already, and notice the list of history nodes that appear in the Inputs section.
- 3 Select Edit > Delete by Type > History.

The construction history is deleted on the helmet mesh. Notice the list of items under Inputs has only one item remaining—the polySurfaceShape node.



- 4 Save your work before proceeding to the next section.

Mirror copying a mesh

Once you've finalized one half of your polygon model (and deleted its construction history) you'll want to create the opposite half by copying it across the axis of symmetry so you have the complete model. You can produce a mirrored copy of a polygon mesh using Mirror Geometry (Mesh > Mirror Geometry).

Before copying one half across the axis of symmetry, you should check that all of the border edges lie along the axis of symmetry. If any edges do not lie along this axis, the original half will be copied based on the vertices that create the widest point and a gap between the two halves may result.

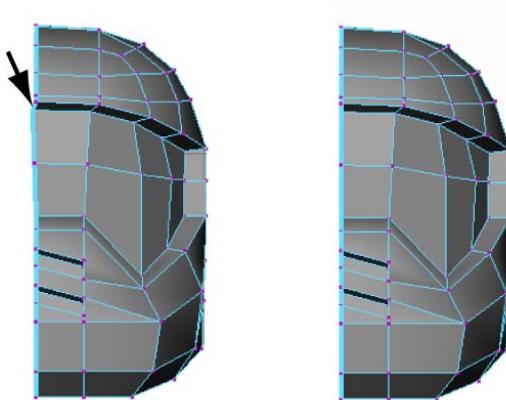
To ensure that the border vertices lie along the axis of symmetry

- 1 Enlarge the front view so you can view the vertices that lie along the axis of symmetry.

If any vertices on the mesh protrude across the Y axis, you'll need to snap these to the Y axis using the Move Tool in combination with the Snap to grids feature.

NOTE Any vertices that protrude across the Y axis likely occurred as a result of an earlier extrude operation. Because the extrude feature extrudes components based on the face normal, a few vertices may have moved across the axis of symmetry.

- 2 In the front view, select all of the vertices that should lie along the axis of symmetry (Y axis) using a bounding box selection.
- 3 On the Status Line, turn on Snap to Grids.
- 4 In the Toolbox, double-click the Move Tool to display its tool settings and ensure that the Retain component spacing setting is turned off.
- 5 In the front view, click-drag the red arrow on the Move Tool manipulator to the right a small distance.
The vertices immediately snap to the grid line to the right. They initially will not be positioned at the location you want, but you'll notice that all of the vertices you selected are now aligned to each other.
- 6 Drag the manipulator towards the left until the vertices snap to the Y axis.



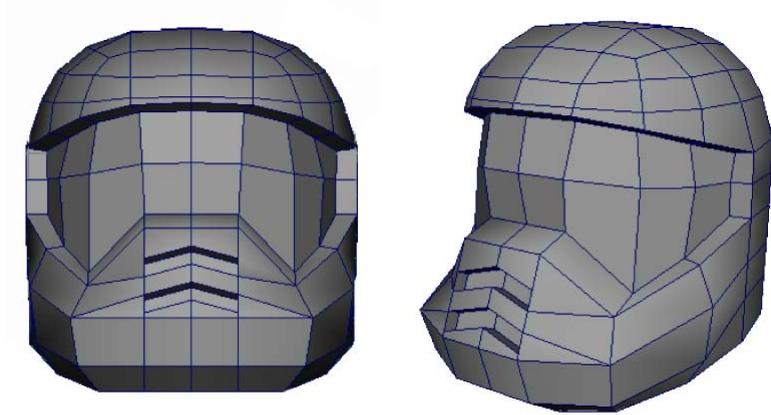
Any vertices that should lie along the axis of symmetry should be aligned using snap to

- 7 On the Status Line, turn off Snap to Grids.
- 8 Click off the mesh to unselect the vertices.

To mirror copy the polygon mesh

- 1 Right-click the mesh, change the selection mode to Object, and then select the helmet mesh.
- 2 Select Mesh > Mirror Geometry > , set the following options, and then click the Mirror button.
 - Mirror Direction: -X
 - Merge with original: On
 - Merge vertices: On

The mesh is copied along -X to create the opposite half of the model. The two meshes are combined into one and vertices are merged so they become shared.



Copy and merge the two halves to create the entire model.

- 3 Save your work before proceeding to the next section.

At this point, the low resolution version of the helmet model is complete. In the next section you'll preview how the model appears at a higher resolution using the subdiv proxy feature.

Working with a smoothed mesh

Depending on the intended use of your polygonal model, you may want a low resolution version, a high resolution version, or both. In Maya, it's easy to increase the resolution of a model using the Smooth feature (Mesh > Smooth).

In addition, you can preview a higher resolution version of the mesh using either Smooth Mesh Preview or the Proxy > Subdiv Proxy feature. You can then view and edit the low resolution and high resolution smoothed version of the mesh separately or simultaneously.

Smooth Mesh Preview lets you quickly preview how your mesh will appear when smoothed.

The Subdiv Proxy feature links the original and the high resolution smoothed versions via construction history, you can make changes to the low resolution version and immediately see the effect on the high resolution smoothed version.

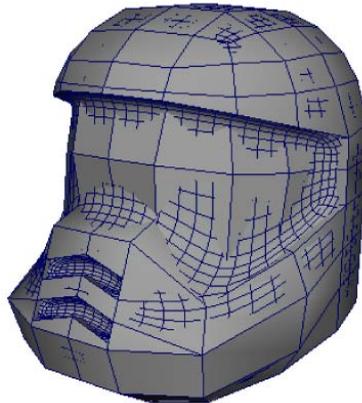
In this lesson, you'll use Subdiv Proxy to work with your smoothed mesh.

To create a high resolution smoothed model using Subdiv Proxy

- 1 Right-click the mesh, change the selection mode to Object, and then select the helmet mesh.
- 2 Select Proxy > Subdiv Proxy > , and set the following:
 - Division Levels: 2
 - Mirror Behavior: None
 - Subdiv Proxy Shader: Keep
- 3 Click the Smooth button.

A higher resolution, smoothed copy of the helmet mesh is created in the same position as the original mesh. It is possible to see the two versions in this position (and you can toggle the display to show either or both) but in this lesson you'll select the high resolution version and move it slightly to the left.

NOTE The two versions of the model are also grouped into the same node hierarchy when the Subdiv Proxy operation is performed.



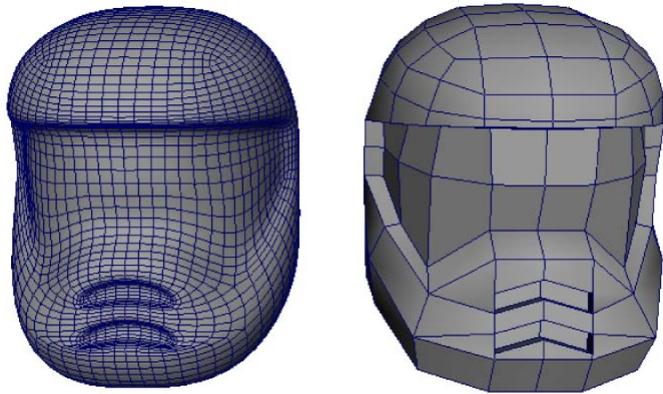
The low resolution and the smoothed high resolution version appear in the same location.

- 4 Select both the low and high resolution versions of the mesh in Object Mode.

- 5 Press the ' key (located to the left of the 1 key) to toggle the display between the low and high resolution versions.
Toggling the display between the two versions is useful when you want to view one or the other.
- 6 Press the ~ key to toggle the display so that both are visible again.

NOTE The Toggle Proxy Display (') and the Both Proxy and Subdiv Display (~) display items are also available from within the Proxy menu.

- 7 Select only the smoothed version of the mesh in Object Mode and move it to the left about 25 units ($X = -25$).
As you edit the low resolution version of the helmet mesh you'll see the resulting updates on the smoothed version beside it.



Move the smoothed version so you can compare the meshes.

Creasing and hardening edges on a mesh

You can crease or harden the edges on your polygon meshes. Hardening or creasing the edges sets how the mesh transitions between faces enhancing the realism of your model.

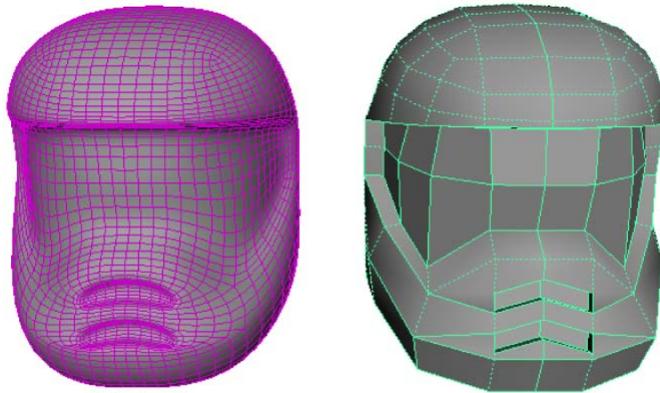
When you harden an edge on a polygon mesh, you change the direction of the normals associated with the shared edge, which in turn affects the shading along those edges. When you crease an edge of a mesh that has an associated subdiv proxy, the edges on the smoothed high resolution version are creased by physically modifying the polygon smoothing surrounding those edges.

In this lesson, you'll try both methods on your helmet mesh.

To display hard and soft edges on a mesh

- 1 Right-click the low resolution mesh, set the selection type to Object Mode, and then select the helmet mesh.
- 2 Select Soft/Hard Edges.

The wireframe mesh on the helmet updates to display both dashed and solid lines. The dashed lines indicate edges that are set to display as soft shaded. The solid lines indicate edges that are set to display as hard shaded. The reason some of the edges are hard and some are soft relates to the particular settings for some polygon tools.

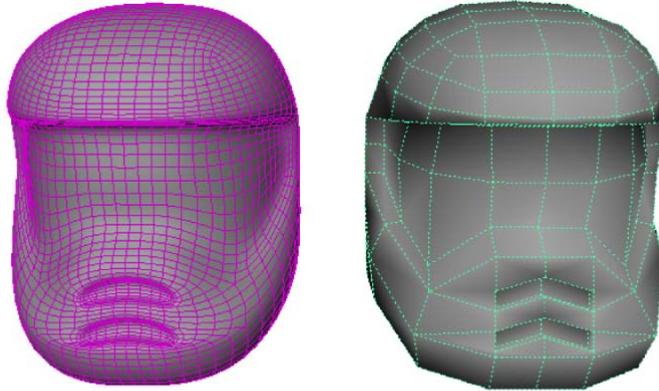


The low resolution version of the helmet displays both solid and dashed wireframe lines to indicate the hard and soft edges.

To soften the edges on the mesh

- With the low resolution version of the helmet still selected, select Normals > Soften Edge.

All of the edges on the helmet mesh are set to be shaded in a softened fashion. The wireframe mesh updates to display as dashed lines indicating that all of the edges are set to be soft shaded. You may not notice any immediate effect on the smoothed high resolution version, but its important that you perform this step before you proceed to harden specific edges on this model.



The low resolution version of the helmet displays only a dashed wireframe when the Soften Edge command is set.

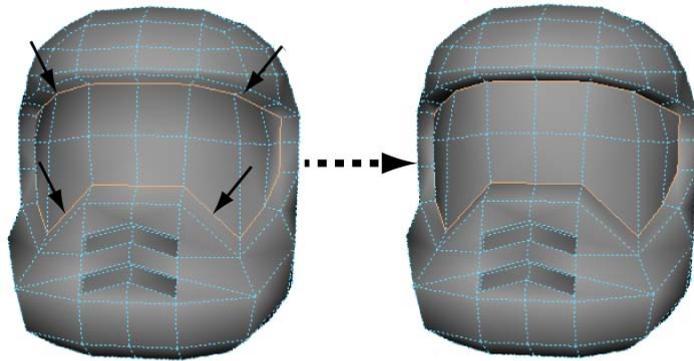
To harden the edges surrounding the face shield

- 1 Right-click the low resolution mesh and set the selection mode to Edge.
- 2 Select the inner edges that surround the perimeter of the face shield (see image below).

TIP You can quickly select the inner edges surrounding the perimeter of the face shield using the Select Edge Loop Tool. You may need to double-click in a couple of areas to complete the selection.

- 3 Select Normals > Harden Edge.

The selected edges are hardened on the low resolution mesh.



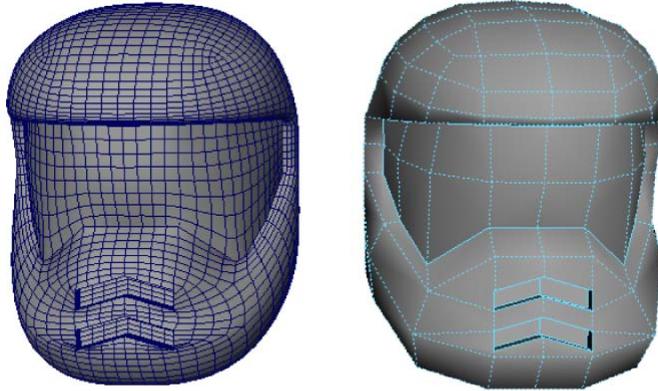
Select the inner edges around the perimeter of the face shield.

The shading on the low resolution version of the helmet changes when Harden Edge is set.

Notice that the same edges on the smoothed high resolution version of the helmet are unchanged. For edges to display sharp on the smoothed version, you must turn on the Keep Hard Edge attribute.

To turn on the Keep Hard Edge attribute

- 1 In the perspective view, select only the smoothed high resolution version of the helmet mesh.
- 2 In the Channel Box, in the Inputs section, click the listed proxy to display its attributes.
- 3 Set the Keep Hard Edge attribute to On.
The sharper transition between the face shield and the rest of the helmet now appears on the smoothed high resolution version.
- 4 Select all of the edges for the grill vents and harden them as well.

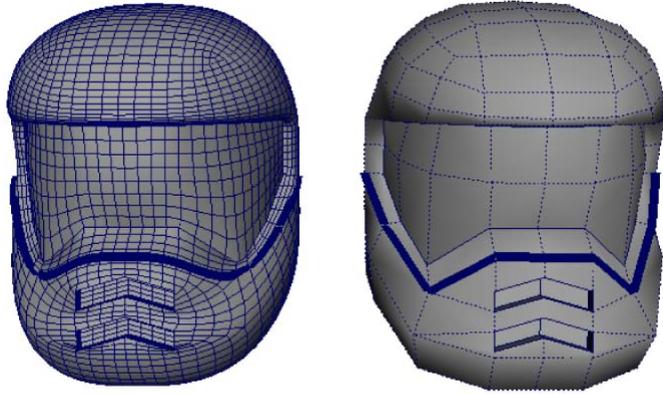


Turning on the Keep Hard Edge attribute displays the hardened edges on the smoothed version of the mesh.

To crease edges on the mesh

- 1 Select the lower outer edges on the face shield (see image).
- 2 Select Edit Mesh > Crease Tool.
- 3 In the scene view, press the middle mouse button then drag the mouse to the right to add a crease to the selected edges.

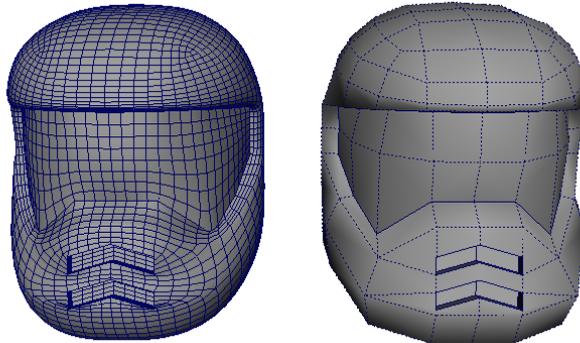
A thick line appears on both the low resolution model and the high resolution version to indicate that a crease has been applied. Notice on the high resolution version that the related edges have become sharper in appearance, but not as sharp as if they were hardened using the Harden Edge feature.



A thick line appears on the edges where the creasing has been set.

4 Save your work.

NOTE You can toggle the creased edge display (thick lines) by selecting Crease Edges.



Your model is now complete. Depending on your requirements you can delete the construction history on both the low resolution and high resolution versions.

Refine any regions on the helmet as required to gain more experience with the tools presented in this lesson.

If you need to work with either version of the model separately you can ungroup the two meshes by selecting the top node of their hierarchy in the

Hypergraph window (Window > Hypergraph: Hierarchy) and selecting Edit > Ungroup.

Beyond the lesson

In this lesson you learned some basic techniques of polygonal modeling:

- You can create complex polygonal models with surprisingly few techniques.
- Starting from a primitive surface such as a cube, you can smooth, scale, move, extrude, split, and rotate components of the primitive to create a low resolution version of the model you want to create.
- You must frequently adjust vertices to fine-tune the shape, and finally smooth the edges between faces where desired in order to produce the final version of your model.
- Another method for previewing a smoothed version of the mesh before it is smoothed is via Smooth Mesh Preview. Press the 2 or 3 key while the mesh is selected to preview it in a smoothed state. Press the 1 key to return it to the unsmoothed state.

Polygonal modeling has many timesaving features not covered in this lesson:

- For example, Boolean operations (union, difference, and intersection) are a common way to create a new object from the interaction of two existing objects.
- If you're planning to use your polygonal surfaces where the poly count is constrained, such as with interactive games, Maya has a number of tools for minimizing the number of polygonal faces of an object such as the Reduce Tool. Fewer faces means simpler geometry. This is critical when fewer polygons means increased interactive performance with games applications. You can view the polygon count on your mesh by selecting the mesh and then choosing Poly Count.
- Maya also has a category of tools called *Deformers*. Deformers let you bend, twist, and scale your objects by enveloping the object in a cage-like manipulator called a *lattice deformer* that you can manipulate.
- The NURBS chapter in this guide describes how to use the Sculpt Geometry Tool to modify a surface by pushing, pulling, and smoothing a surface without first selecting vertices. Though the lesson shows how to shape a NURBS surface, you can apply many of the same techniques when sculpting polygonal surfaces.

- You can also modify a polygon mesh using the Soft Modification Tool which lets you smoothly modify a group of vertices on a mesh.

If you want to learn more about a particular tool or feature that has been presented in this lesson, refer to the Maya Help.

Lesson 2: Sculpting a polygon mesh

Introduction



In this lesson you alter an existing polygonal head model into a goblin by sculpting the mesh with the transformation tools.

In this lesson you'll be introduced to some of the polygon modeling options Maya provides by learning how to:

- Use reflection to perform symmetrical transformations to your model
- Use Soft Select to transform components with falloff
- Modify the falloff area and shape
- Create a custom falloff shape

- Select components using Marquee select and Drag select
- Use Camera based selection to limit the components you select
- Use the Split Edge Tool to separate edge rings

Open the scene for the lesson

In this lesson, you'll work with a scene that we created for your use. The scene contains a human head. In the remainder of the lesson, you'll alter this into a goblin head.

To open the scene

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 74.
- 2 Open the scene file named Poly_head.ma.

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

```
Getting Started/PolygonModeling/Poly_head.ma
```

The scene contains a human head in a visible layer with a hat and glasses on hidden layers.

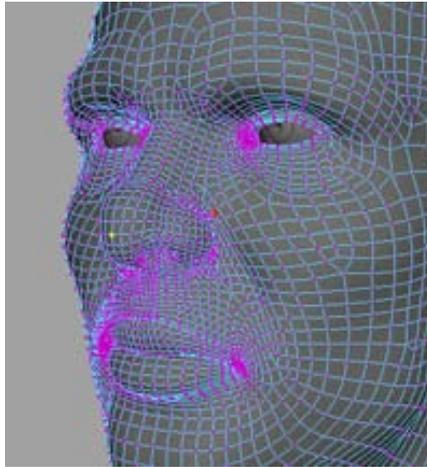
Using Soft Select

You can use Soft Select to select components with falloff on a polygon or NURBS surface. Falloff is an area around the selected components that is affected by a transformation according to a weighting. Transforming components with falloff allows you to perform a smooth transformation over multiple components without having to adjust each component individually.

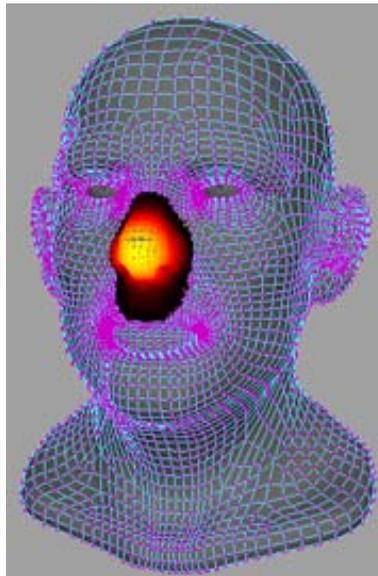
For this lesson you use Soft Select to make a number of adjustments to the head model, starting with the nose.

To stretch the nose

- 1 Select the Move Tool by clicking the icon on the Toolbox.
- 2 Right-click the head model and select Vertex from the marking menu. The vertices appear on the mesh.
- 3 Select a vertex on the tip of the nose.



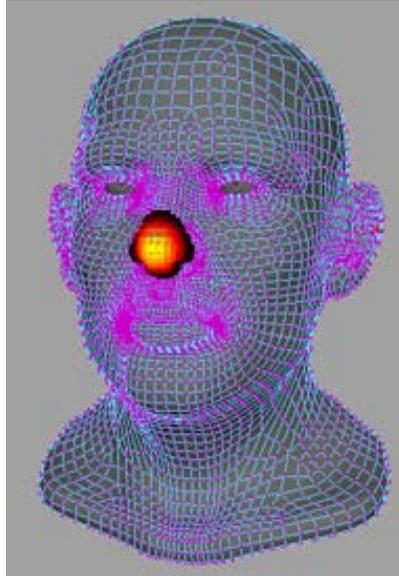
- 4 Press the 'b' key.
A circular colored area appears around the selected vertex.



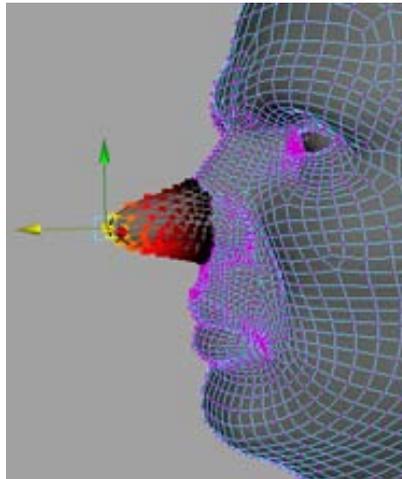
This colored area represents the falloff. Yellow represents areas of low falloff while black represents areas of high falloff. Areas with low falloff are affected more by transformations to selected components than areas of high falloff.

Next you need to adjust the Falloff radius.

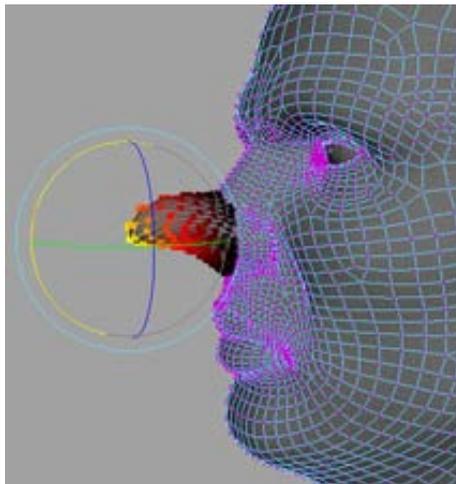
- 5 Hold the b key and drag the left mouse button left or right. The falloff area grows and shrinks as you move the mouse. Adjust the size so that it encompasses as much of the nose as possible without touching the face.



- 6 Drag the z axis manipulator away from the face. As you drag the vertex away, the rest of the nose is smoothly stretched with it.



- 7 Select the Rotate Tool from the toolbox and, using the z axis manipulator, rotate the vertex down approximately 45 degrees.



- 8 Select the Move Tool again and drag the y axis (it appears yellow when selected and green when unselected) manipulator down. This creates a bend in the nose.



- 9 Turn off Soft Select by pressing the b key on your keyboard.

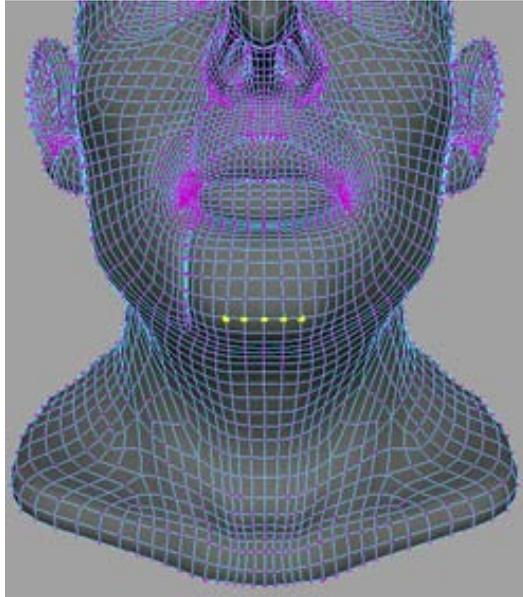
Selecting with Camera based selection

When selecting components on a model you often only want to select the components you can see and not the components behind them. You can use Camera based selection to limit your marquee select to only what the camera can see.

To pull out the chin

- 1 Double-click the Select Tool icon in the toolbox.
The Tool Settings editor appears.
- 2 In the Common Selection Options section, underneath Marquee, turn on Camera based selection.
- 3 Right-click the head and select Vertex from the marking menu.
The vertices appear on the mesh.

- 4 While looking at the front of the head, select the five vertices on the same vertex loop at the front of the chin by holding the left mouse button on one corner of them dragging a marquee selection over them.
Only the five vertices in the front are selected.



- 5 Turn on Soft Select by pressing the 'b' key on your keyboard.
A colored falloff appears on the surface of the face.
- 6 Hold the 'b' key on your keyboard and drag the left mouse button left or right to adjust the Falloff radius so that it encompasses only the chin.
- 7 Select the Move Tool and drag the chin down and away from the face so that it becomes longer and more angular.



Now you have a longer chin without affecting any of the geometry on the neck, which may have occurred accidentally if Camera based selection had been off.

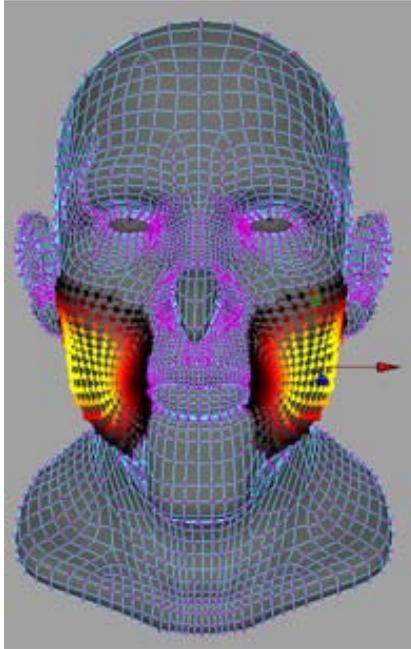
Sculpting with symmetry

To give the head a more goblin-like appearance, you need to make the cheeks and eyebrows more boney. However, to do this, you need to adjust both cheeks equally. You can do this with the transformation tools' Reflection settings.

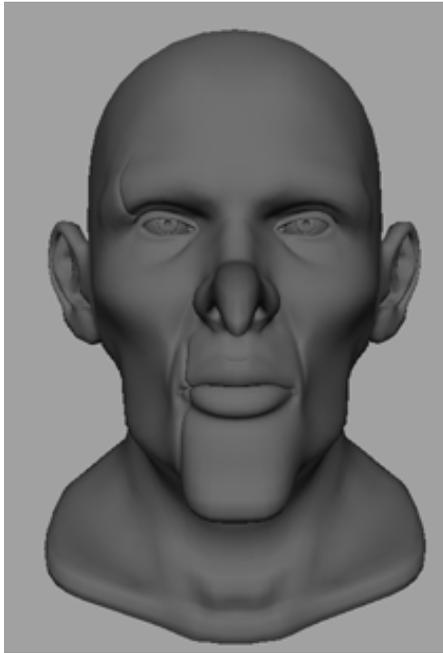
To modify both cheeks simultaneously

- 1 Double-click the Move Tool in the Toolbox.
- 2 In the Reflection section, turn on Reflection and set the Reflection axis to X.
- 3 Right-click the head and select Vertex from the marking menu.
The vertices appear on the mesh.

- 4 With the camera positioned in front of the head, select a set of vertices on the left cheek.
A colored falloff appears on both cheeks.



- 5 Pull the x axis manipulator to the left so that the cheeks are pulled closer together.
Both cheeks are transformed identically and appear more boney.

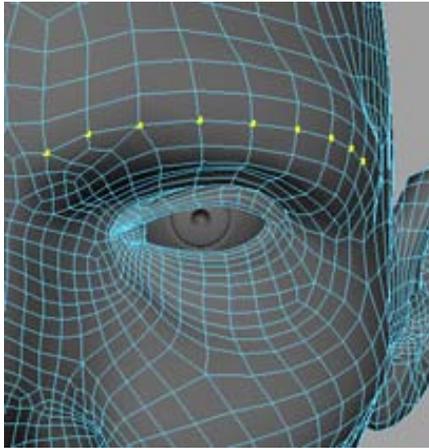


Now you can adjust the eyebrows to make the goblin angrier.

To rotate the eyebrows

- 1 Press the 'b' key on your keyboard to disable Soft Select.
- 2 Click the outermost vertex on the left eyebrow.
- 3 Shift + double-click the innermost vertex on the left eyebrow.

All the vertices on the same loop are selected between and including the two vertices.



NOTE The selection of vertices on the other side of the reflection seam does not appear complete after this step. This is normal and is rectified in step 4.

- 4 Press the 'b' key on your keyboard to enable Soft Select.
- 5 Hold the 'b' key on your keyboard and drag the mouse left and right to adjust the Falloff radius so that it only affects the eyebrow.
- 6 Double-click the Rotate Tool from the Toolbox.
The Tool Settings editor appears.
- 7 Drag the dark blue rotate manipulator in the scene to rotate the eyebrows inward.



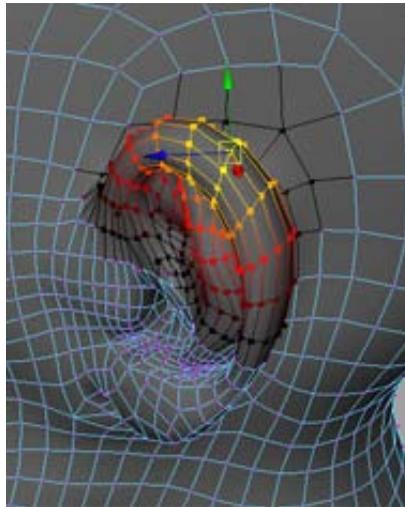
Sculpting with Surface based falloff

The default setting in Maya uses a spherical volume to determine what components are contained in the Falloff radius. However, sometimes you need the Falloff radius to follow the contours of the surface. To do this, you must use Surface based falloff.

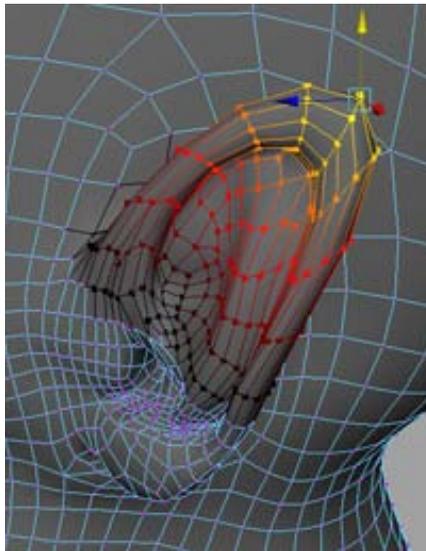
To make the ears pointy

- 1 Double-click the Move Tool in the Toolbox.
The Tool Settings editor appears.
- 2 In the Soft Selection section, set the Falloff mode to Surface.
- 3 Tumble the model so that you have a clear view of the left ear.
- 4 Right-click the head and select Vertex from the marking menu.
The vertices appear on the mesh.
- 5 Select a vertex on the tip of the ear.

- 6 Hold the 'b' key on your keyboard and drag the mouse left or right to adjust the Falloff radius so that it encompasses the top half of the ear.



- 7 Use the manipulators to pull the ear until it's pointed.



Surface based falloff ensures that only the ear is affected by your transformation. If you were using Volume based falloff, parts of the head would also be moved along with the ear.

- 8 Select the Scale Tool and scale the ears along the x axis so they become thinner.
- 9 Select the Move Tool again and pull the pointy end of the ears away from the head.



Selecting with Drag select

In addition to marquee selecting sets of components, you can also select them by dragging over them with the mouse. This is similar to the way that the Paint Selection Tool works.

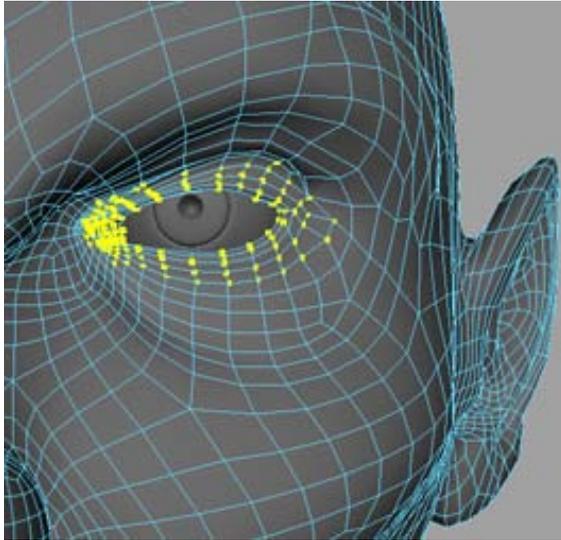
To make the eyes sockets more pronounced

- 1 Double-click the Select Tool in the Toolbox.
The Tool Settings editor appears.
- 2 In the Common Selection Options section, turn on Drag.
The mouse cursor turns into a brush cursor.
- 3 Beneath the Drag option, turn on Camera based selection.
- 4 Right-click the head and select Vertex from the marking menu.

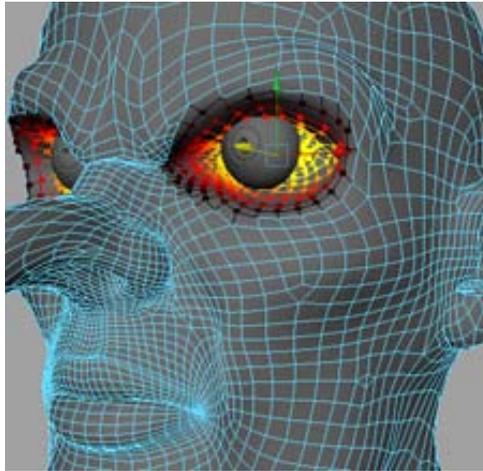
The vertices appear on the mesh.

- 5 On one side of the mesh, drag the mouse around the eye socket, beneath the eyebrow until you've covered a small area around the eye.

You may find it easier to temporarily disable Soft Select to get a better view of what you are selecting. You can quickly toggle Soft Select on and off by pressing the 'b' key on your keyboard.

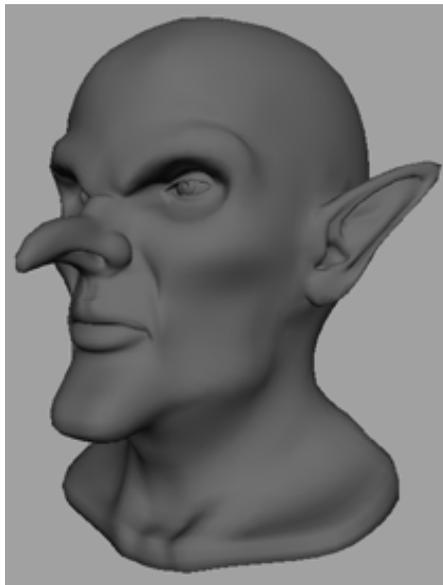


- 6 With Soft Select on, hold the 'b' key and drag the mouse to resize the falloff area so it covers only the eye sockets.
- 7 Select the Move Tool in the Toolbox and pull the z axis manipulator toward the head.



The eye sockets are pulled in so that both the eyebrow line, bridge of the nose, and cheeks are better accented.

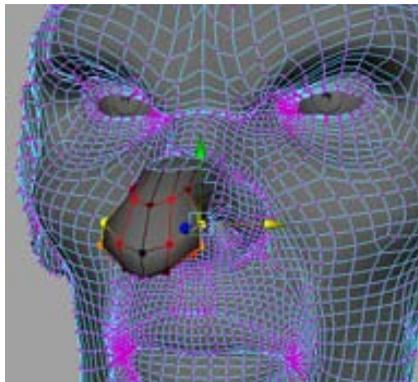
- 8** Press the b key to turn Soft Select off.
- 9** Right-click the head and select Object from the marking menu.
- 10** Select the two eyes and move them back along the z axis so they fit into the eye sockets again.



Adjusting the Seam tolerance

You can use the seam tolerance to fatten the nose at the end.

- 1 Right-click the head and select Vertex from the marking menu.
The vertices appear on the mesh.
- 2 Select a vertex on the left side of the tip of the nose.
- 3 Press the 'b' key to turn Soft Select on.
- 4 Hold the 'b' key and move the mouse left or right to resize the Falloff radius so it only encompasses the tip of the nose.
- 5 Select the Move Tool by double-clicking it in the Toolbox.
- 6 In the Tool Settings Editor, in the Reflection Settings section, click the Reset button.
- 7 Set the Seam Tolerance to 5.
The Seam Tolerance controls the falloff around the seam. By increasing the number, you ensure that the vertices at the tip of the nose spread evenly.
- 8 Move the vertex away from the reflection line along the x axis to widen the end of the nose.



Now you can unhide the hat layer to finish off the goblin.

To unhide the hat

- Open the Channel Box and click the first empty box in the hat layer.



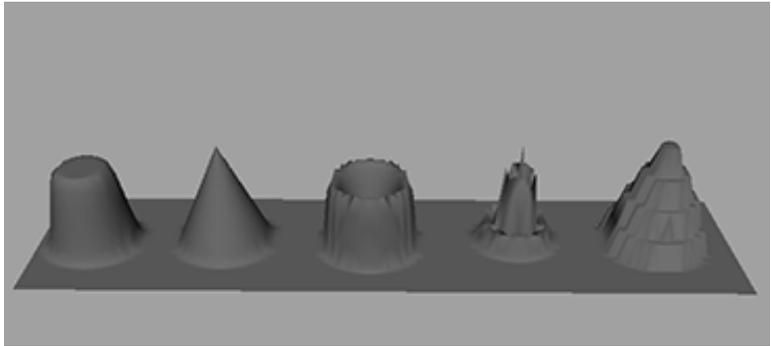
Beyond the lesson

In this lesson, you learned how to:

- Activate Soft Select and adjust the Falloff radius.
- Transform components with Soft Selection.
- Use Camera based selection.
- Select components using reflection.
- Use both surface based and volume based falloff.
- Use Drag select.
- Adjust the Seam tolerance.

There are some additional Soft Selection options which are not described in this lesson:

- You can change the shape of the falloff curve in the Soft Selection tool settings either by adjusting the Falloff curve graph or by selecting any of the Curve presets below it.



You can also save a custom Falloff curve to the Curve presets. For more information, see [Change the shape of the falloff area](#).

NURBS Modeling

4

Introduction

NURBS (Non-Uniform Rational B-splines) use a method of mathematically describing curves and surfaces that are well suited to 3D applications. NURBS are characterized by the smooth organic forms they produce.

NURBS surfaces can be quickly modeled and edited using a variety of techniques. NURBS surfaces are created using one or more NURBS curves that define the profile of the shape that you want for a surface, and then using a specific construction method to create the finished surface.

NURBS curves and surfaces have many applications and are the preferred surface type for industrial and automotive designers where smooth forms with minimal data are required to define a particular form. NURBS curves are ideal for defining a smooth motion path for an animated object. With NURBS, a surface can be modeled and then converted to a poly mesh.

In this chapter, you will learn some basic techniques for working with NURBS curves and surfaces in Maya. This chapter includes these lessons:

- Lesson 1 Revolving a curve to create a surface: [Introduction](#) on page 154
- Lesson 2 Sculpting a NURBS surface: [Introduction](#) on page 160
- Lesson 3 Lofting curves to create a surface: [Introduction](#) on page 175

Preparing for the lessons

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

- 1 Select File > New Scene to create a new scene before starting each lesson.

- 2 Make sure the Construction History icon (below the menu bar) is on:



. (If it is turned off, it has a large X across it.)

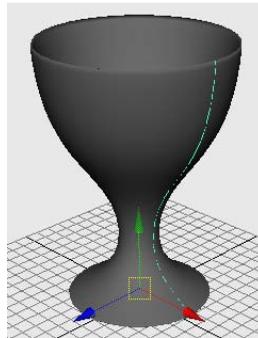
- 3 If you have not already done so, copy the `GettingStarted` folder from its installation location to your `projects` directory. Then, set the `GettingStarted` directory as your Maya project. For more information, see [Copying and setting the Maya project](#) on page 25.
- 4 Select the Surfaces menu set. Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the Surfaces menu set.

NOTE Before you perform the lessons in this book, ensure that the Interactive Creation option for primitives is turned off by selecting `Create > NURBS Primitives > Interactive Creation`. That is, ensure a check mark does not appear beside the menu item.

- 5 Make sure that Soft Selection is turned off by opening the Tool Settings Editor with the Select Tool active and unchecking the box marked Soft Select.

Lesson 1: Revolving a curve to create a surface

Introduction



A simple technique for creating NURBS surfaces is to create a curve for the profile of your desired form and then create a surface using one of the various

NURBS surfacing tools available in Maya. In this lesson, you create an egg holder using NURBS curves and the Revolve Tool.

In this lesson you learn how to:

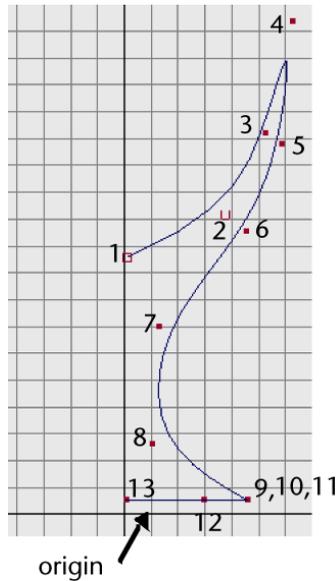
- Create a NURBS curve using the control vertices (CV) creation technique.
- Use the grid for visual reference when modeling.
- Determine the start and end points for a NURBS curve and its direction.
- Create a revolved NURBS surface using the Revolve tool.
- Change the display shading smoothness for a NURBS surface.
- Edit a NURBS surface by editing its initial profile curve when it is linked to the surface by construction history.

Creating a profile curve

Revolve is a common modeling technique similar to shaping a vase on a spinning potter's wheel. You must first create a profile curve in an orthographic view and use the Revolve Tool to create a surface.

To create a profile curve

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 153.
- 2 Select Panels > Layouts > Four Panes so you can see multiple views of the scene.
- 3 Select Create > CV Curve Tool.
This is the most commonly used tool for drawing curves.



- 4 In the front view, click the numbered positions as shown in the figure. Make sure the first and last positions are on the grid's Y-axis (the thickest vertical line of the front view's grid).

Also, click three times in the same spot for positions 9, 10, and 11. This is necessary to create a sharp point or corner in the curve.

To change the position of the most recent point clicked, you can middle-mouse drag it.

- 5 (Windows and Linux) After you click position 13, press Enter to complete the curve's creation.

(Mac OS X) Press Return to complete the curve's creation.

The points you clicked in the scene view become the control vertices (CVs) of the completed curve. CVs are points you manipulate to alter the shape of a curve (or surface). These points often lie away from the curve.

It takes a little practice to be able to predict exactly where a curve will be positioned when you click CV positions with the CV Curve Tool. Once you've learned the technique, you'll be able to create and edit curves with graceful symmetry. By default, you must click at least four points to create a curve with the CV Curve Tool.

For some advanced operations in Maya, you need to know the meaning of the *start of a curve* and the *curve direction*. The start of a curve is the first CV you create when you draw the curve. It is indicated by a small hollow

box at the first CV, visible as you create the curve. Curve direction is shown on the second CV after the start of the curve, indicated by a small u icon as you create the curve.

Creating a revolve surface

With the Revolve surface tool, a curve is rotated about an axis to create a surface. The user can define the axis of revolution.

To create a revolve surface

- 1 With the curve selected, select Surfaces > Revolve. This creates the egg holder surface from the revolved profile curve. Examine the results in the perspective view.

Maya does not delete the profile curve. In a subsequent step, you'll edit the profile curve to alter the shape of the surface.

- 2 Select the surface and rename it `Eggholder` in the Channel Box.
- 3 With the pointer in the perspective view, press 5 (for Shading > Smooth Shade All).

This displays the egg holder as a shaded surface rather than a wireframe in the perspective view.

Press 1, 2, and 3 to switch between the different degrees of display smoothness. (These hotkeys correspond to the menu items under Display > NURBS menu item (Rough/Medium/Fine).

The finer the smoothness, the greater the impact is on interactive performance when you work with complex models.

The smoothness display only affects the scene view. Rendered images display with high quality smoothness regardless of this setting.

Editing a revolve surface

After you create a surface from a revolved curve, you can reshape the original creation curve to reshape the surface. This is possible because Maya's Construction History feature was turned on before you did the revolve operation.

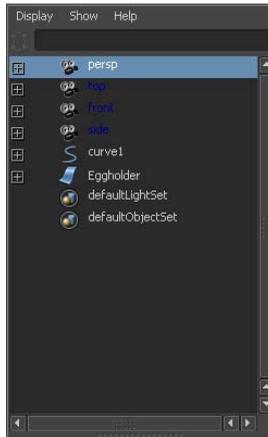
Selecting the original creation curve can be challenging when it is situated amongst other existing curves and surfaces. The Outliner is an editor that is

useful for quickly examining the structure and components of a scene. The Outliner is also useful for selecting objects in situations like this.

To edit a surface with construction history

- 1 Select Window > Outliner.

The Outliner appears and displays a list of the items in the scene.



- 2 In the Outliner, select the curve you revolved (curve1) by clicking on its name in the list.

The curve becomes highlighted in the scene views.

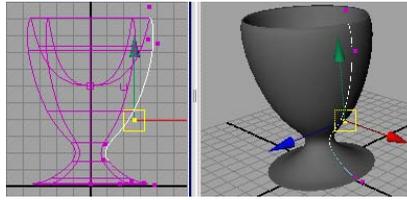
- 3 In the front view, right-click and select Control Vertex from the marking menu.

The CVs for the curve appear.

- 4 Select and move one or more CVs to adjust the curve shape as desired.

If you have trouble selecting one of the CVs in the front view, select it in the perspective view after dollying and tumbling the camera as necessary.

This modifies the shape of the egg holder because it is linked to the shape of the curve by its construction history.



- 5 If desired, save the scene for future review.
- 6 Close the Outliner window.

Beyond the lesson

In this lesson you were introduced to a few basic techniques related to NURBS modeling:

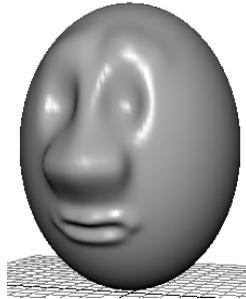
- Revolving a curve is the easiest way to create surfaces with radially symmetrical forms— wheels, vases, glasses, pillars, and so on.
- NURBS surfaces are webs of interconnected curves. The creation curves are used to create and subsequently modify the surfaces if required.
- Proficiency at drawing and editing curves is an important part of NURBS modeling.
- The grid is a useful aid for constructing and modifying curves.

Some additional points you should know are:

- You cannot render curves, so your work with them is always an adjunct to creating and editing surfaces.
- Besides moving a curve's CVs to alter its shape, you can cut, attach, extend, close, and smooth curves.
- After you create a surface from a revolved curve, you can edit the surface at a component level by moving CVs or scaling groups of CVs on that surface to customize the shape of the surface. (You will need to turn off construction history in order to do this.)

Lesson 2: Sculpting a NURBS surface

Introduction



In addition to creating NURBS surfaces using curves, you can edit and sculpt surfaces and primitive objects in Maya using the Sculpt Geometry Tool.

The Sculpt Geometry Tool allows you to interactively push or pull on the surface regions to create areas that are embossed or in relief in relation to the surface.

In this lesson, you learn some of the basic concepts of these tools by sculpting a cartoon face and head from a NURBS primitive sphere using the Sculpt Geometry Tool.

In this lesson you learn:

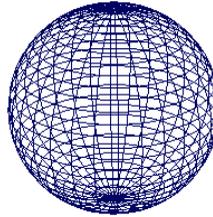
- Basic sculpting operations (push, pull, smooth, relax and erase) with the Sculpt Geometry Tool.
- How the density of isoparametric lines affects the surface detail possible when sculpting.
- How to increase the surface subdivisions on a NURBS surface to aid with surface sculpting.
- How to change the brush radius for the Sculpt Geometry Tool.
- How Opacity and Max Displacement affect the sculpting operations.
- How to import geometry from a pre-existing file into your current scene.

Preparing a surface for sculpting

Because a head is roughly spherical, you can create a primitive sphere as a quick starting point for creating a head and face.

To prepare a sphere for sculpting

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 153.
- 2 Select Create > NURBS Primitives > Sphere > . In the options window, select Edit > Reset Settings, enter the following values, then click the Create button:
 - Radius: 6
 - Number of Sections: 30
 - Number of Spans: 30



The Radius sets the sphere's size in grid units. A value of 6 creates a sphere big enough to use the grid for size comparison.

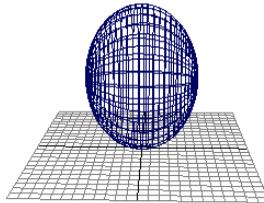
The Number of Sections sets the number of vertical curves, called *isoparms*, for the sphere. Isoparms show the outline of the surface shape. The more isoparms a surface has, the more CVs it has. (By default, CVs are not displayed.) More CVs means better precision as you edit a surface. The Number of Spans sets the number of horizontal isoparms.

The drawback to having too many CVs is that you'll have a harder time making smooth shape changes to broad regions. Lots of CVs also means slower processing time as you work with the surface. It's best to make surfaces with as few CVs as necessary.

We chose 30 Sections and Spans for this lesson because experience has shown that this allows adequate facial subtlety without slowing Maya operation on a workstation of modest processing power.

NOTE In wireframe display mode, if you select Display > NURBS > Fine or Medium, more isoparms appear than there are actual spans and sections. The surface is visually displayed with extra precision, but the extra isoparms have no CVs and cannot be edited.

- 3 Name the sphere Egghead.
- 4 Rotate the sphere 90 degrees on its side (Rotate Z: 90). This positions the sphere's CVs well for modeling a simple head and face. You'll learn why later in this lesson.



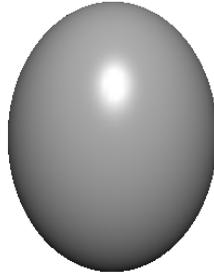
- 5 To give Egghead an oval shape, set the ScaleX for the sphere to 1.3 or so. Optionally, you can move Egghead above the grid so the grid doesn't interfere with your view of Egghead. Also, position the perspective view so the Z-axis of the View Axis points toward you.

Modifying the surface material for easier viewing

Because you will do subtle surface modeling in this lesson, it's helpful to display Egghead with bright highlights so you can clearly see the effects of the changes you make. In the next steps, you'll assign a Blinn material to Egghead to give its surface bright highlights. The steps have no explanations, as the shading subject matter is the focus of a later lesson.

To assign a Blinn surface material

- 1 With Egghead selected, press 5 to display the surface with smooth shading.
- 2 Right-click Egghead and select Assign New Material > Blinn.
The Attribute Editor will immediately be displayed for the Blinn material.
- 3 Drag the Color attribute slider roughly 3/4 of the way to the right, set Eccentricity to 0, and close the window.



Basic sculpting techniques

In the next steps, you'll become familiar with features of the Sculpt Geometry Tool. With this tool, you use your mouse or stylus to push, pull, or smooth a surface's shape without selecting or displaying CVs. After you practice sculpting the surface, you'll erase your practice strokes then begin sculpting Egghead's features.

To practice sculpting using basic sculpting operations

- 1 With Egghead selected, select Edit NURBS > Sculpt Geometry Tool >  . In the Tool Settings window, click Reset Tool, and make sure the Sculpt parameters are displayed.

In the Sculpt Parameters section, there are a number of operations for sculpting a surface. Each affects the region of the surface where you drag (*stroke*) your mouse or stylus relative to the surface normals:



- The *Push* operation depresses the surface in the region of the stroke.
- The *Pull* operation raises the surface in the region of the stroke.
- The *Smooth* operation diminishes bumps or ridges where you drag.

Smoothing works by averaging the position of all features within the brush radius with all other nearby features.

- The *Relax* operation paints over bumps to relax most of the out of place surface features.

Unlike smoothing, Relax works by only averaging the largest surface anomalies so that the overall shape is maintained.

- The *Erase* operation eliminates the effects of the other four operations.

The Erase operation works up to the last time you saved the scene or clicked the Update button to the right of Erase Surface.

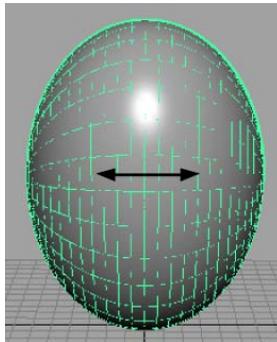
- 2 Experiment with each of the five operations on the surface. Don't be concerned with the results. Just become familiar with the response to your mouse strokes.

By default, Push and Pull strokes deform the surface in a direction normal (perpendicular) to the surface. To create the resulting deformation more accurately, use Push and Pull strokes in a front, side, or top view while examining the results in a separate perspective view. Tumble the perspective view regularly for the best visual feedback.

When you position the mouse pointer on the surface, it changes to a red sculpt icon that shows an abbreviation for the name of the operation you are performing. For instance, Ps stands for Push.

The red icon also displays the radius of the region affected by the stroke. The Radius(U) value changes the radius.

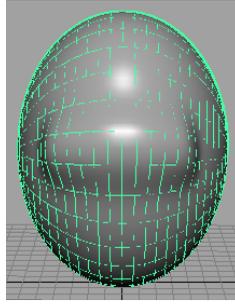
- 3 Select the Erase operation and click the Flood button. This erases all your changes.



- 4 Select the Pull operation and enter a Radius(U) of 0.25, then drag between a pair of horizontal isoparms without crossing either.

This has no effect because the stroke radius didn't make contact with the CVs of either isoparm. Regardless of which operation you use, only CVs are affected by the strokes.

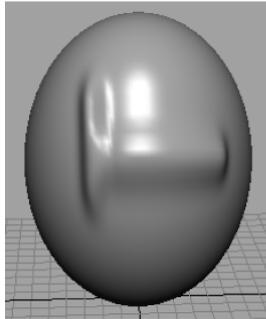
- 5 Change the Radius(U) to 2 and repeat the prior strokes.



The strokes alter the surface because the radius overlaps the CVs. As this example shows, you need to make sure the radius is big enough to influence the desired CVs. It's common to change the radius many times during a sculpting session.

If you prefer to affect a small region without increasing the radius, you can add CVs to the region by inserting more isoparms. You'll do this later in the lesson.

- 6 Flood-erase the changes to the surface as you did previously.
- 7 Select the Pull operation with a Radius(U) of 0.5. Drag along a vertical isoparm. For comparison, drag along a horizontal isoparm.

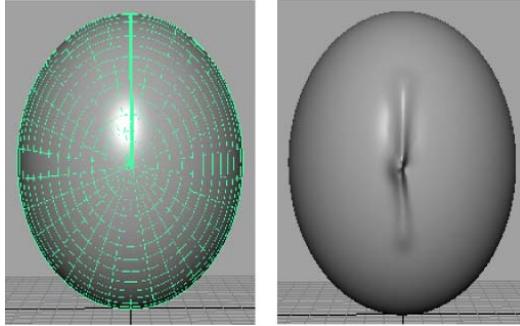


Before taking the snapshot for this figure, we set the Wireframe on Shaded on the Display tab to None(Window > Setting/Preferences > Preferences). This turned off highlighting of the surface isoparms, making surface alterations easier to see. Consider switching between the off and on settings for Show Wireframe as you learn the Sculpt Geometry Tool.

The vertical stroke creates a thinner ridge than the horizontal stroke because the density of vertical isoparms is greater. The number of underlying CVs and their positioning affects the outcome of your strokes.

TIP You can alternatively adjust the radius of the Sculpt Geometry Tool using a Hotkey. Move the tool over the object, press and hold the b Hotkey and drag left or right. The circle on the object with numerical radius value represents the size of your tool. Stroke on the object to try out a new size.

- 8 Flood-erase the changes to the surface as you did previously.
- 9 Rotate the camera view so the X axis of the View Axis points toward you. Draw a vertical Pull stroke again.



As your stroke nears the center of Egghead, a kink occurs. It's challenging to alter a surface symmetrically in a region where many isoparms converge at a single point, called a *pole*. Always consider the position of isoparms as you sculpt a surface. In general, sculpt where isoparms are evenly, regularly distributed.

- 10 Erase all changes again, then reposition the camera view so the Z-axis of the View Axis points toward you.

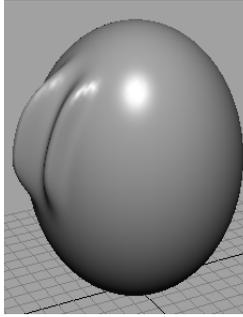
TIP If the Sculpt Geometry Tool is selected but not the surface you want to work on, right-click the surface and select Select from the marking menu.

Additional sculpting techniques

Now that you've learned the basic features, you'll learn other useful techniques before starting to model the facial features.

To practice additional sculpting techniques

- 1 In the Sculpt Parameter section of the Sculpt Geometry Tool, set Max Displacement to 2.



- 2 Draw a vertical Pull stroke.
- 3 Change the Max Displacement to 1.
- 4 Draw another vertical stroke nearby.
The Max Displacement sets the maximum distance the surface's CVs are pushed (or pulled) with a single stroke.
- 5 Erase the changes to the surface.
- 6 Set Max Displacement to 2.
- 7 Draw a vertical Pull stroke.
- 8 Set the Opacity from the default value of 1 to a value of 0.2.
- 9 Draw another vertical stroke nearby.
One ridge is higher than the other. The Opacity value scales the influence of Max Displacement. For example, with an Opacity of 0.2, each stroke has only roughly 0.2 times the effect of the Max Displacement setting.
It's generally best to use a low Opacity value as you Push or Pull. You can increase the deformation gradually with multiple strokes.
If you want to affect only a small region of a surface, use your mouse to click rather than stroke the region. If the Radius(U) and Opacity of the brush is small, you might need to click the nearest intersection point of two isoparms.
- 10 With Opacity set to 1, erase all changes to the surface again.

TIP Consider using an electronic tablet with pen stylus. A pen stylus is more natural for stroking surface changes as it feels like a pen. With a stylus, you can set an option that causes the Radius or Opacity to vary with stylus pressure. For example, you can have heavy strokes create a bigger Radius than light strokes.

To do this, display the Stroke tab of the Sculpt Geometry Tool. For Stylus Pressure, select Opacity, Radius, or Both. Radius is a common choice. If you select Radius, the Radius(U) value on the Sculpt tab sets the maximum radius, while Radius(L) is the minimum. Explore various settings.

If you use a mouse, set the Stylus Pressure to None. The Radius(U) sets a fixed radius. Radius(L) is ignored.

Sculpting a nose

Now you'll begin sculpting Egghead's face, starting with a simple nose.

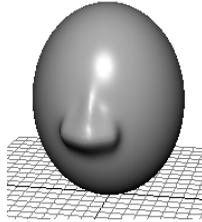
To sculpt a nose for the character

- 1 Position the perspective view so the Z-axis of the View Axis points toward you.
- 2 Reset the Sculpt Geometry tool.
- 3 Select the Pull operation and set the following options:
Radius(U): 1 Opacity: 0.2
- 4 Select the Gaussian brush profile setting so your strokes have a soft, faded edge:



- 5 Above the midpoint of Egghead, stroke vertically downward from the top of the nose to the tip—about the distance between two horizontal isoparms (see the following illustration). Use several strokes to build up the bridge of the nose. Create nostrils by stroking horizontally to the right of the tip of the nose a few times, then to the left of the tip a few times.

To build up a small area, position the stroke icon there and click the mouse rather than drag. Tumble the view to examine your results after each stroke or click.



If the strokes create a bumpy surface, turn on the Smooth operation and click Flood once or twice to smooth all strokes on the surface. Because Opacity is 0.2, the Smooth operation is subtle. Alternatively, you can smooth a selected region by stroking just that area. It's common to smooth a surface regularly while using pull and push strokes.

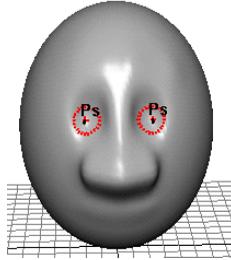
Because there are relatively few isoparms in the nose area, you can only create a simple nose that's broad and rounded. If you want to create sharper features, for instance, depressed nostrils or sharp ridges, you'll need to insert isoparms in the nose region. [Sculpting a mouth](#) on page 171 describes how to insert isoparms.

Sculpting eye sockets

Eye sockets provide an inset and backdrop for eyes.

To sculpt eye sockets for the character

- 1 Select the Push operation.
- 2 Leave other settings the same as for the nose.
- 3 To make a pair of identical eye sockets, display the Stroke settings in the Tool Settings window and turn on Reflection.
- 4 Set the Reflection Axis to X and then position the sculpt icon on Egghead. A pair of identical Push icons appears on Egghead.
- 5 Adjust the positioning as desired for the eye sockets. Click the mouse many times in the desired location.



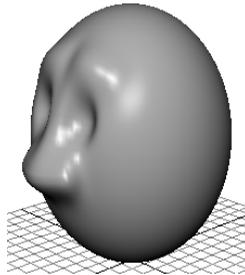
NOTE If the Reflection setting does not push or pull in the same direction on either side of a NURBS surface you can adjust the Reference Vector to alter the direction of these operations. In the Sculpt Parameters section select UV Vector and turn on Enable UV Vector Adjustment. Click and drag the mouse left or right to adjust the direction of the U and V reference vectors.

Sculpting eyebrows

Eyebrows help define how stern or pleasant a face appears.

To sculpt eyebrows for the character

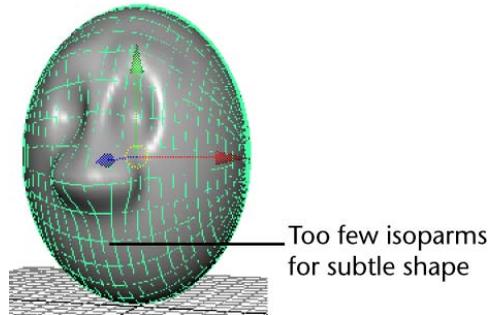
- 1 Select the Pull operation. Use the same option settings as for the eye sockets, including Reflection.



- 2 Position the sculpt icons above the eyes and draw horizontal strokes. If necessary, click positions where you need to build up the eyebrows. Smooth the eyebrows as necessary.

Sculpting a mouth

With the 30 Sections and 30 Spans specified for the original sphere in this lesson, the large space between isoparms in the mouth region makes it impossible to create a subtle shape for the lips.



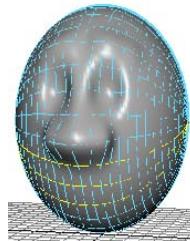
To overcome this problem, you must insert isoparms in the mouth region before sculpting.

To insert additional isoparms

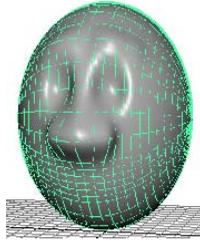
- 1 Right-click Egghead and select Isoparm from the marking menu.
- 2 From the Toolbox, choose the Select Tool.



- 3 Click the horizontal isoparm below the nose and shift-click the next two below it so they become yellow.



- 4 Select Edit NURBS > Insert Isoparms > . In the options window, turn on Between Selections, enter a value of 2 for # Isoparms to Insert, then click Insert.



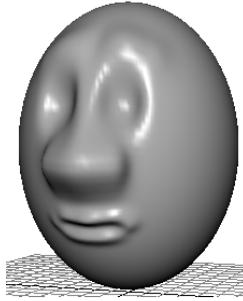
This inserts two extra isoparms between each pair of selected isoparms, for a total of four extra isoparms. This provides enough CVs to create subtlety in the mouth.

You might want to add vertical isoparms at the lips in a similar way. The extra isoparms would also be useful if you were to later enhance the shape of the nose. Insert isoparms only where needed. More isoparms means slower processing speed.

In any case, do not change the number of isoparms by editing the original number of Sections and Spans in the makeNurbSphere history node. This will reshape your sculpted surface undesirably.

To sculpt a mouth for the character

- 1 Select the Sculpt Geometry Tool again.
- 2 Select the Push operation. Enter 0.2 for Radius(U). In the Stroke tab, make sure Reflection is turned on.
- 3 Starting at the center of the area appropriate for the indentation between the lips, stroke outward from the center. It's best to start the stroke with the dual icons on top of one another (in other words, only one icon is displayed).
- 4 Select the Pull operation and set the Radius(U) to 0.3. Using a similar technique as for the prior step, stroke the upper lip. Repeat for the lower lip.



NOTE It might be easier to push or pull the lips with the Ref. Vector set to Z Axis. This moves the region you stroke in the world Z-Axis direction. The default Normal setting moves the region in a direction normal (perpendicular) to the surface. Because the normal direction on a lip might be up, down, or straight out, depending on the part of the lip you stroke, there's more possibility of undesired results when you use the Normal setting.

Sculpting other facial features

You can now optionally add other facial features. Consider these tips:

- For facial features such as the chin, cheeks, and forehead, use a Radius(U) larger than 1 and start with a Maximum Displacement between 0.5 and 1.
- Insert isoparms wherever you want to add extra detail with the Sculpt Geometry Tool.
- To create eyeballs for the sockets, create a sphere and scale it as necessary to fit the socket. You might want to increase the eyeball's Scale X value to make it oblong. To duplicate the eyeball, and make a mirrored copy of the original you'll first need to set the pivot point for the eyeball to be at the origin 0, 0, 0 using the following steps. With the sphere selected, press the Insert (Windows and Linux) or Home (Mac OS X) key so that the sphere's pivot point is displayed. Next move the pivot point to the origin using the X hotkey so that the pivot snaps to the grid at 0, 0, 0. Select Insert (Home) to turn off the pivot point display. Finally, use Edit > Duplicate Special > and set the Scale setting to -1, 1, 1. This makes the new eyeball a mirrored duplicate of the original on the opposite side of Egghead's face. Position the eyeballs in the sockets. Parent the eyeballs to Egghead.

- To create simple, unadorned ears, you can create, scale, and squash a sphere. Create ridges and valleys by pulling CVs or by using the Sculpt Geometry Tool. Duplicate the ear by first setting its pivot point to the origin and with the Scale option set to -1, 1, 1. Parent the ears to Egghead.

To import existing models into your scene:



- 1 If you saved the Eggholder scene in the prior lesson, you can import the egg holder into this Egghead scene, then position Egghead into the egg holder.

To import the egg holder into the Egghead scene, use File > Import and select the name of the scene that contains Eggholder.

Importing a scene imports all objects from that scene.

- 2 Increase (or decrease) the scale of the egg holder (or Egghead) as necessary for a snug fit.

Beyond the lesson

In this lesson you were introduced to a few basic techniques related to sculpting surfaces:

- The Sculpt Geometry Tool is indispensable for quickly shaping a variety of surfaces. As you do your own projects, pay special attention to the position and density of isoparms before you begin sculpting.
- Isoparms converge at a single point (the *pole*). Pole regions of a primitive or surface are hard to sculpt so it's best to avoid using the Sculpt Geometry Tool there. For example, when you created the original sphere in this lesson, you rotated it 90 degrees around its Z-axis. Because of the rotation, ears for the head would need to be modeled via a different technique (for example, you could model a pair of ears and parent them to the head).

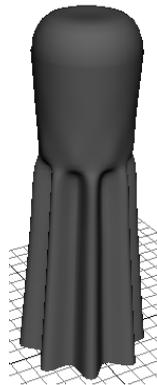
- The density and orientation of isoparms on a surface affects the results with the Sculpt Geometry Tool. As you gain NURBS modeling experience, you'll learn how to use the density and orientation of isoparms to your advantage.
- Primitive objects are useful objects for sculpting in many cases. After you create a primitive, you typically sculpt, scale, trim, or otherwise alter the object into a more complex shape. Though most primitives are surfaces rather than curves, they still derive their shape from curves.



In general, a sphere makes a convenient foundation for creating a simple head, but it's not ideal if you plan to animate an expressive, talking head. Many 3D artists start with a cylinder or a lofted surface. The procedures for doing this are more complex than starting with a sphere and are beyond the scope of this lesson.

Lesson 3: Lofting curves to create a surface

Introduction



Another method for creating NURBS surfaces is to loft a series of curves that define the cross section of your desired surface form. Lofting a surface works like stretching a skin over each of the cross sections to create the final surface.

In this lesson, you learn additional NURBS surfacing techniques by using the Loft tool to create the body of a salt shaker. You will also create the cap for the salt shaker by modifying the shape of a primitive sphere.

In this lesson you learn how to:

- Modify the outline of a circle primitive by editing the position of its CVs.
- Use the magnet snap feature.
- Loft cross section curves to create a NURBS surface using the Loft tool.
- Edit the form of an existing primitive object by moving its CVs.
- Parent one object to another using the Outliner.

Creating profile curves for a surface

You begin the lesson by creating a pair of profile curves. You then create several copies of these curves to form the skeletal contours of the salt shaker's surface.

To create profile curves for the salt shaker's body

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 153.
- 2 Select Create > NURBS Primitives > Circle > . In the options window, select Edit > Reset Settings, enter the following values, then click the Apply button:
 - Radius: 4
 - Number of Sections: 24

The Radius sets the size of the circle in grid units. A value of 4 creates a circle with enough size so that you can use the grid for convenient size comparison.

The Number of Sections sets the number of CVs in the circle. By using 24 CVs, you can create a circle with the subtle contours required in this lesson. Display the CVs of the circle to see the 24 CVs.

A circle is a curve that loops back on itself. The CVs work the same way as for any other curve. The more CVs you create for a curve or surface, the more detail you can give its shape.

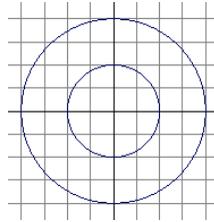
The drawback to using many CVs is that you'll have a harder time making smooth shape changes to broad regions. Also, more CVs means slower processing time as you work with the curve. It's best to make curves with as few CVs as necessary. With experience, you'll learn how many CVs to use in a situation.

3 Create another circle, this time with these options:

- Radius: 2
- Number of Sections: 24

This creates a smaller circle inside the first circle as shown below.

4 Change the scene view so you can see the circles from the top view.

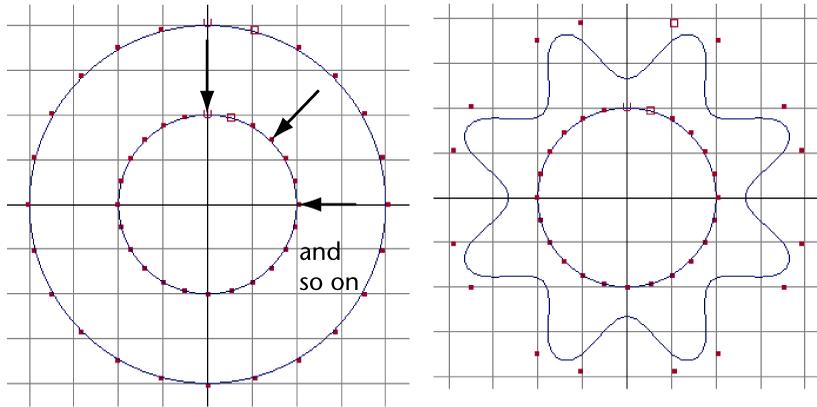


5 Right-click the inner circle and select Control Vertex from the marking menu. Repeat for the outer circle.

6 Turn on Snap to Points (below the menu bar).



7 As shown in the following figure, move CVs from the outer circle to the corresponding inner circle positions. As you drag a CV near its destination, the Snap to Points mode causes the CV to jump to the exact CV position.



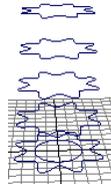
- 8 Turn off Snap to Points. Turn off the display of CVs for both circles by right-clicking on the circles and selecting Object Mode from the marking menu.

Duplicating curves

In the next steps, you'll create several copies of the previously created circles (also called *curves*) to form the skeletal contours of the salt shaker's surface.

To duplicate curves for the lofted surface

- 1 Right-click the outer curve and choose Select from the marking menu.
- 2 Select Edit > Duplicate Special > . In the options window, select Edit > Reset Settings, enter the following values, then click Duplicate Special:



- Translate: 0 3 0
- Scale: 0.93 1 0.93
- Number of Copies: 4

This creates four copies of the original curve, each translated three units above the last, and each scaled smaller than the last.

- 3 Select the inner circle and then move it to 0, 13, 0 by entering these translate values in the Channel Box. Increase its Scale attributes to 1.33, 1.33, 1.33..



- 4 Use Edit > Duplicate Special > with the following options:
 - Translate: 0 2 0
 - Scale: 1.05 1 1.05
 - Number of Copies: 3

This creates three copies of the circle, each translated two units above the last, and each bigger than the last.

Lofting a surface

You've created a total of nine curves. In the next steps, you loft these curves into a surface that matches the contours of their perimeters. Above the midpoint of the surface, the shape will change from a flower-shaped perimeter to a circular perimeter.

To loft a surface for the salt shaker's body

- 1 Starting from the bottom circle, and working towards the top, shift-select all the circles, one at a time.

The order of selection is important when shift-selecting for a loft operation. You must start with the bottom flower-shaped circle and end with the top circular-shaped circle. Alternatively, you can drag a selection box around the circles. The surface to be lofted will be based on the selection order of the curves.

- 2 Select Surfaces > Loft.
This creates the salt shaker's body.
- 3 Name the surface `shakerBody`.
- 4 Press 5 to display the view with smooth shading.



Lofted surface set
to smooth shaded
mode

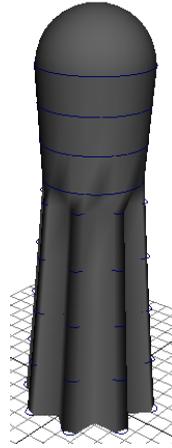
NOTE Many of the surface creation tools also have the option of creating polygon or subdivision surface versions of a surface with the same input curves. You do this by setting the appropriate Output Geometry option for that particular surface tool.

Modifying a primitive object

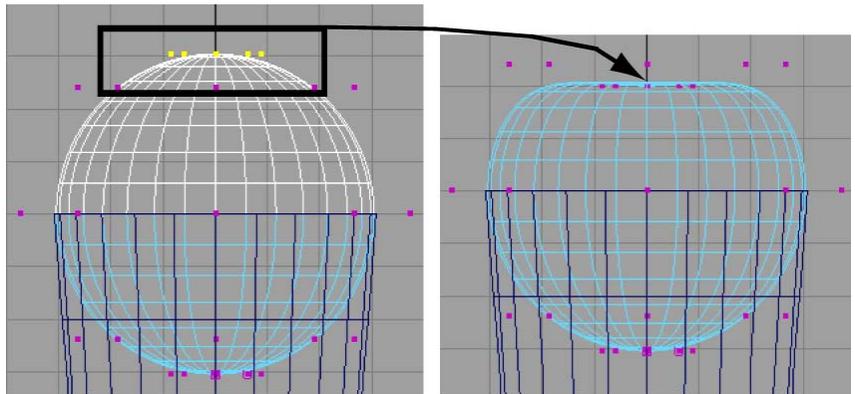
To create a simple cap for the salt shaker, you'll create a sphere, alter its shape, and position it above the body.

To create and modify a sphere for the salt shaker's cap

- 1 Select Create > NURBS Primitives > Sphere > . In the options window, select Edit > Reset Settings, and click the Create button. Maya puts the sphere at the origin, its display currently obstructed by the salt shaker body.
- 2 Name the sphere `Cap`.
- 3 Move the cap to the top of the body.



- 4 Scale the cap size so that its diameter fits snugly at the top of the body.
- 5 Adjust the cap's position as desired.
- 6 In a side view, make sure that the cap is displayed as a wireframe.
- 7 Right-click on the Cap and choose Control Vertex from the marking menu.
- 8 Select the top row of CVs and drag them down until the top of the cap is flattened:



Check that the bottom half of the cap isn't poking through the visible surface of the body. You can scale the bottom CVs inward to avoid this problem.

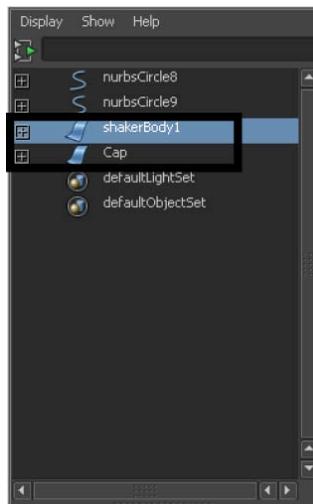
Using the Outliner to parent objects

The Outliner is an editor that is useful for quickly examining the structure and components of a scene. You can use the Outliner to quickly parent objects to create an object hierarchy.



To parent an object using the Outliner

- 1 From the main menu, select Window > Outliner.
The Outliner window is displayed. A list of the scene's components is listed in the Outliner.
- 2 In the Outliner, click on the word `Cap` to select it.
- 3 Parent the cap to the body by middle-mouse dragging `Cap` onto `shakerBody1` in the Outliner.



By parenting the objects, you can move, rotate, or scale the cap and the body as a single entity by selecting only the body.

Notice that the salt shaker displays horizontal curves that wrap around the body. The curves are the original circles you used to loft the surface. Because these curves are part of the surface's construction history, you can alter their shape if you decide you want to alter the shape of the body and the body will update based on the construction history.

If you prefer not to see the curves in the scene view, select the curves and select Hide Selection. The curves won't appear when you render an image of the scene. Maya displays only surfaces, not curves, in rendered images.

Beyond the lesson

In this lesson you were introduced to the basic techniques related to lofting curves to create NURBS surfaces:

- You made a salt shaker with two surfaces: a body made from lofted curves and a cap made from an altered sphere. An advantage of creating a separate surface for the cap and base is that you can easily give each a different color or texture, for example, one chrome and the other marble. Another advantage of creating a separate cap is that you can animate the object separately. For example, you could choose to animate the cap unscrewing from the shakerBody.
- You used a Loft operation rather than a Revolve operation to create the body. The vertical corrugations on the surface would be impossible to create by revolving a curve.
- You can alter the position of the profile curves and the shape of the shaker will update because of the construction history. If you're certain you won't change the body's shape by editing the shape of the lofted curves, you can delete the body's construction history to quicken Maya's processing of your interaction with the surface. (For a surface as simple as the salt shaker, deleting the history won't boost processing much.)
- There are many other useful tools for creating and editing surfaces. For a glimpse of the possibilities, take a look at the Surfaces menu and the Edit NURBS menu.

If you want to learn more about a particular tool or feature that has been presented in this lesson, please refer to the Maya Help.

Subdivision Surfaces

5

Introduction

Subdivision surfaces are a hybrid surface type that possess characteristics of both NURBS and polygonal surfaces as well as other features not offered by the other surface types. Like NURBS, subdivision surfaces are capable of producing smooth organic forms and can be shaped using relatively few control vertices. Like polygonal surfaces, subdivision surfaces let you extrude specific areas and create additional detail in your surfaces. You do this by working at different levels of detail on the subdivision surface and switching between the levels when necessary. With a subdivision surface you can build a smooth organic object from a single primitive and not be concerned about attaching or stitching the various surfaces together as you would for a NURBS surface.

Preparing for the lesson

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

- 1 Review the Polygon Modeling lesson in this guide. Knowledge of polygonal modeling tools is essential to becoming proficient in subdivision surface modeling.
- 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](#) on page 25.
- 3 Create a new scene.
- 4 Select the Surfaces menu set.

- 5 Select Window > Settings/Preferences > Preferences.
- 6 In the Preferences window, select Subdivs from the Categories list to display the subdivision surface preferences.
- 7 In the Subdivision Surfaces Component Display settings, set the Component display to Numbers, then click Save.

Setting the subdivision components to display as numbers allows you to see the subdivision surface level directly on the surface.

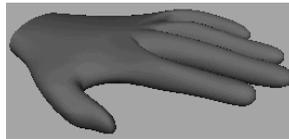
NOTE Before you perform the lessons in this book, ensure that the Interactive Creation option for primitives is turned off by selecting Create > Polygon Primitives > Interactive Creation. That is, ensure a check mark does not appear beside the menu item.

- 8 Make sure that Soft Selection is turned off by opening the Tool Settings Editor with the Select Tool active and unchecking the box marked Soft Select.

Lesson 1: Modeling a subdivision surface

Introduction

In this lesson you learn some of the basic tools for working with subdivision surfaces as you model a human character's hand.



In this lesson, you learn how to:

- Convert a polygon surface to a subdivision surface.
- Work with subdivision surfaces in Polygon Proxy Mode.
- Split subdivision faces to create areas for more detail in a model.
- Extrude the split faces to create fingers on the hand.
- Work with subdivision surfaces in Standard Mode.

- Change the Display Level when working in Standard Mode.
- Add more detail to subdivision surface models using Refine Selected Components.
- Create a crease along a vertex edge.

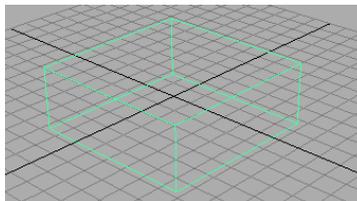
Creating a subdivision surface

You begin the lesson by creating a polygonal cube. You'll convert this cube to a subdivision surface that will be the foundation of the hand. The reason for starting with a polygonal surface and converting it to a subdivision surface is that you can thereafter edit the object's shape with more versatility: you can edit with subdivision surface tools *and* polygonal tools.

To create a polygon cube

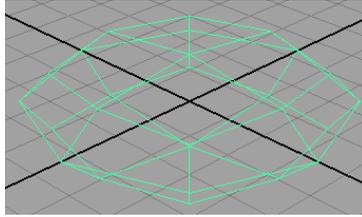
- Select Create > Polygon Primitives > Cube > . In the options window, select Edit > Reset Settings. Then enter the following values and click the Create button.
 - Width: 8
 - Height: 2.5
 - Depth: 8

The resulting polygonal cube is roughly proportional to the palm of a hand.

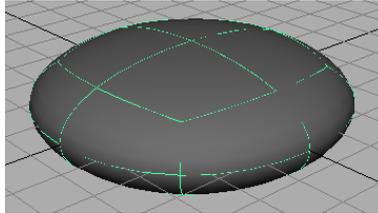


To convert a polygonal cube to a subdivision surface

- 1 To convert the polygonal cube to a subdivision surface, select Modify > Convert > Polygons to Subdiv.



- 2 Press 3 on your keyboard (to select Display > Subdiv Surfaces > Fine).
- 3 Press 5 (to select Shading > Smooth Shade All).

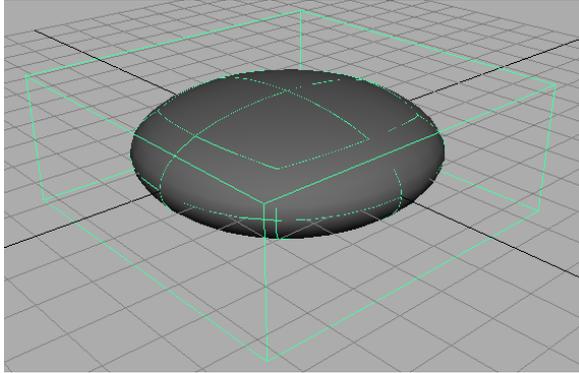


With the smoothness set to Fine, you can see that the conversion to a subdivision surface creates a rounded, smooth shape. By pressing 3, the subdivision surface is displayed more precisely in the scene view. This gives a closer approximation of what the surface will look like when you create a rendered image of the scene.

As an alternative, you could have chosen a Rough smoothness (press 1) so that Maya processes your editing changes in the scene view more quickly. Regardless of the smoothness in the scene view, surfaces always are displayed precisely in rendered images of the scene.

To display a subdivision surface in polygon proxy mode

- Select Subdiv Surfaces > Polygon Proxy Mode.



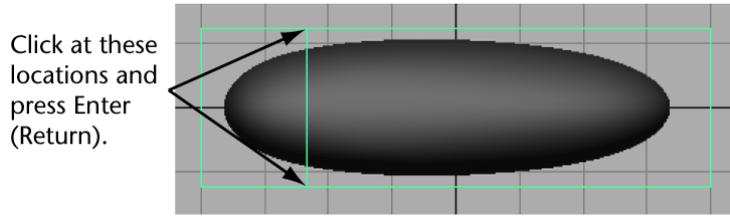
The wireframe cube looks the same as the original polygonal cube. The new cube is called a *polyToSubd1* for the subdivision surface as a result of the conversion. Rename it *LeftHand*. You can use polygonal modeling tools to edit the shape of the polygonal proxy, which indirectly alters the shape of the subdivision surface. Unlike working with an actual polygonal object, your modifications result in perfectly smooth surface changes rather than faceted changes.

Splitting a surface in polygon proxy mode

Next, you use polygonal modeling techniques to split a face into multiple faces to be extruded as fingers.

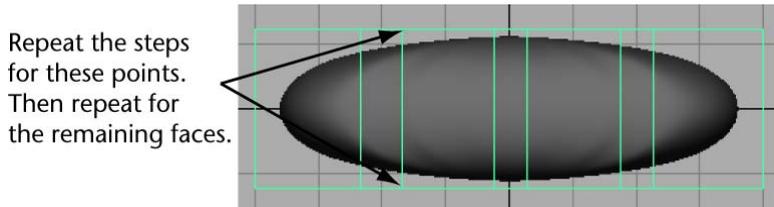
To split the polygon face into multiple faces

- 1 Select the Polygons menu set.
- 2 With *LeftHand* still selected, select Edit Mesh > Split Polygon Tool > . In the tool settings window click the Reset Tool button.
You'll use this tool to split the front face of the proxy into several faces to be extruded into fingers.
- 3 In the front view, click the point on the top edge as shown in the following figure, then click the corresponding point directly below it on the bottom edge. If the line between the points isn't perfectly straight, use the middle mouse to drag the second point to the correct position. Press Enter (Windows and Linux) or Return (Mac OS X).



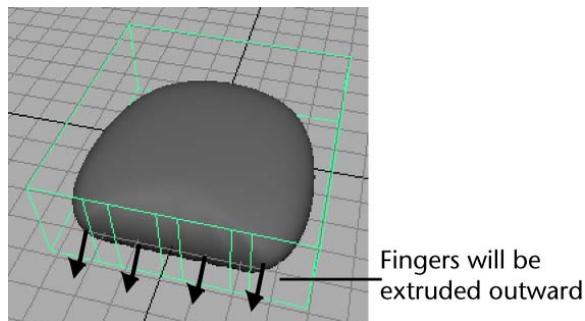
This splits the face into two faces. You'll extrude the left face into a finger later.

- 4 Repeat the preceding two steps as necessary to split the face as follows:



Notice how splitting the face into multiple faces alters the shape of the subdivision surface. The front part of the subdivision surface now resembles the proxy shape more. If you were to split the front face several more times, especially near the outer edges, the subdivision surface would sharpen and resemble the proxy shape even more. More faces means finer control, often at the expense of making the surface harder to work with.

With the added faces, the subdivision surface looks a bit like the palm of a hand. You'll build upon this shape to create a left hand with palm facing downward and fingers extruded outward. You'll extrude the wide faces into fingers, and you'll leave the narrow faces as webbing between the fingers.

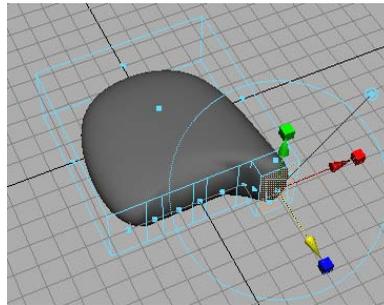


Extruding polygon faces

Next, you'll extrude faces to create fingers for the hand.

To extrude the polygon faces to create fingers

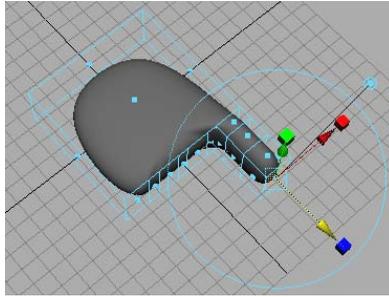
- 1 In a perspective view, right-click LeftHand and select Face from the marking menu.
This lets you select faces.
- 2 Select the right-most face by dragging a selection box around the tiny box at its center. (The subsequent illustration shows which face to select.)
- 3 Select Edit Mesh > Extrude.
- 4 Drag the blue arrow manipulator outward a little to create the first segment of the smallest finger.
The blue arrow turns yellow when selected.



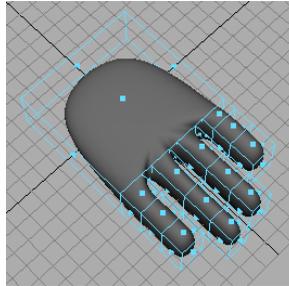
NOTE Do not be alarmed if the small dots at the center of each face do not appear. This means you are in Select Faces by Whole Face mode, which makes no difference for this tutorial. If you want to switch to Select Faces by Center mode, you can do this by opening the Maya preferences (Window > Settings/Preferences > Preferences) and then selecting the Selection section. You can toggle the face selection mode from here.

- 5 Repeat the prior two steps to create the middle segment of the smallest finger.
- 6 Repeat the prior two steps once again to create the top segment and complete the finger.
By creating three segments for a finger, you mimic a real finger's natural structure. The borders between the segments have vertices (not displayed)

currently) that let you reshape those regions, for instance, to create knuckles.



- 7 Similarly, extrude the ring, middle, and index fingers from the appropriate wide faces. Don't extrude the three small faces that lie between the wide spaces. Leave them in position to allow for webbing between the fingers.



NOTE Do not be concerned if the hand you create does not match the lesson's illustrations. Your goal in this lesson is to learn the workflow of subdivision surfaces, not to perfect your modeling technique.

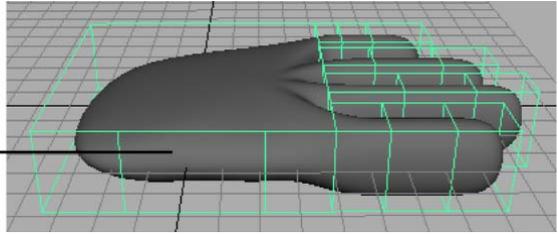
Next, you extrude a thumb using similar techniques as you used for the fingers. The following steps are abbreviated. See the preceding pages for details on the described tools, if necessary.

To split and extrude a face for the thumb

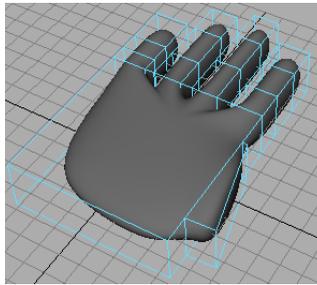
- 1 Select LeftHand.
Make sure it is highlighted in green in the scene view. If it is not green, select LeftHand in the Outliner.
- 2 Use Edit Mesh > Split Polygon Tool to split the side face in the thumb region into three faces.

While splitting, make sure the middle face is larger than the outer faces as shown below.

This face must be larger than the faces on each of its sides.



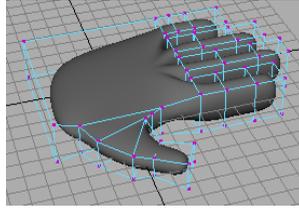
- 3 Extrude the middle face and drag it directly outward.
- 4 Click the surrounding blue circle and use the rotate manipulator to aim it in a direction appropriate for a thumb.



- 5 Extrude three times more to create each segment of the thumb. Use the extrusion manipulator to rotate, move, and scale each extrusion to create the desired shape of a thumb. Use the following figure as a guideline. Again, it's unnecessary to match the illustration or create a realistic thumb for this lesson.

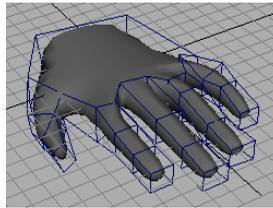
The extrusion manipulator is easy to figure out by clicking and dragging its various elements. If necessary, you can also use the conventional Move, Scale, and Rotate Tools to reshape the faces that control the thumb's shape.

- 6 In the scene view, right-click LeftHand and select Vertex from the marking menu. The purple vertices at the corners of the faces control the adjacent region of the hand.



- 7 Make coarse adjustments to various parts of the hand by repositioning the vertices with the Move, Scale, and Rotate Tools.

The Move Tool works on individual vertices or groups of vertices. Rotate and Scale works on two or more vertices. Undo any changes you don't like using Ctrl-Z (Windows and Linux) or Control-z (Mac OS X).



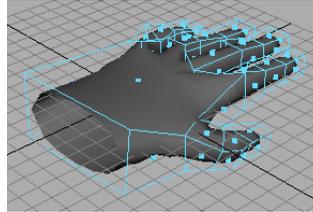
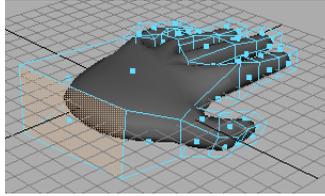
Try to create the approximate thickness, length, and curvature of a cartoon character's hand and fingers. There's no need for perfection. You'll refine the fingers later.

Deleting polygon faces

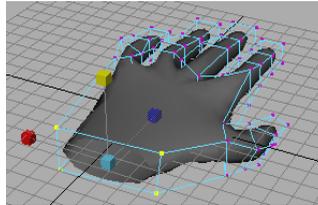
In the next steps, you delete a face in order to create a hole in the wrist where you might choose to attach an arm. Though you won't create and attach an arm in this lesson, it's useful to learn the technique for future use.

To create a hole at the wrist by deleting a face

- 1 In the scene view, right-click LeftHand and select Face from the marking menu.
- 2 Select the face at the wrist and press Delete to create a hole where you could attach an arm.



- 3 Scale the four vertices that surround the hole and scale them inward to narrow the wrist region of the hand.

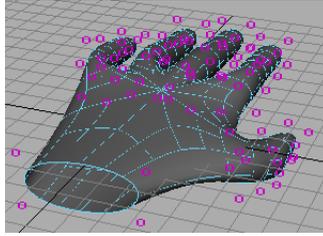


Subdivision surface levels

Next, you refine the shape of the fingers by working in Standard Mode. In Standard Mode, you can make edits to mesh vertices, edges, or faces at different levels of refinement, that are not possible in Polygon Proxy mode. The different levels of refinement are referred to as *subdivision surface levels*.

To refine the fingers using subdivision surface levels

- 1 Select LeftHand.
- 2 Select the Surfaces menu set.
- 3 Select Subdiv Surfaces > Standard Mode. This switches to a mode that lets you modify the surface at various levels of detail to suit your needs.
- 4 In the scene view, right-click LeftHand and select Vertex from the marking menu.



This displays 0s on the surface in the same positions as the vertices in Polygon Proxy mode. In fact, the 0s *are* vertices.

The number 0 refers to the level of detail you are capable of editing. The 0 level is identical to the coarse control possible in Polygon Proxy mode by manipulating the vertices. You can move, rotate, and scale the 0s to alter the shape of the surface just as you did for vertices in Polygon Proxy mode.

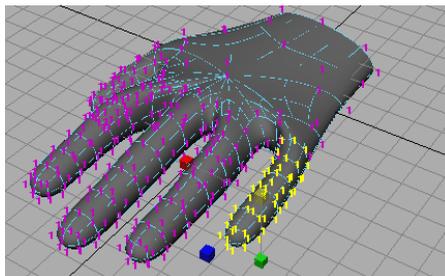
NOTE If numbers do not appear on the hand, open your preferences (Window > Settings/Preferences > Preferences). In the Display > Subdivs section, set Component Display to Numbers.

- 5 Right-click the surface and select Display Level > 1.

Though the 1s are displayed in the same regions as the 0s were previously, there are many more 1s displayed. The 1s are also vertices, and their increased presence means you can refine the shape with more subtlety.

- 6 Refine the shape of the fingers, thumb, and hand as desired by selecting various level 1 vertices and moving, scaling, and rotating them with the transformation tools. (Scaling and rotating has effect only if you select two or more vertices.)

It's essential to tumble the camera from various perspectives to make sure you've selected only the desired vertices.



To return to level 0 to alter the shape with fewer vertices, right-click the surface and select Display Level > 0. Each level has advantages. Level 1 allows finer control because there are more vertices. Level 0 allows control of broader regions, and it's often easier to select the desired vertices since there are fewer of them. It's common to switch back and forth between levels repeatedly in a work session.

The hand has level 2 vertices at various locations, for instance, at the webbing between the fingers. You can also work at that level of detail if desired (select Display Level > 2).

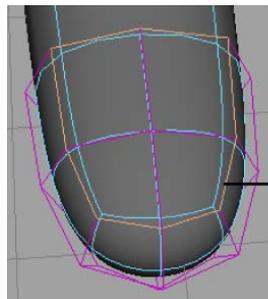
Refining surface components

By default, a subdivision surface displays up to three levels of detail (0, 1, and possibly 2). The 0 level is the least refined. You can increase the level of detail for a selected region to at most 13 levels. Though increasing the number of levels allows more precision, it also slows Maya processing. You'll accomplish most of your modeling needs at levels 0, 1, and 2.

If you display level 2 vertices on the hand, you'll notice that there are no level 2 vertices near the fingertips. To create the indentation of a finger nail, you need more vertices than levels 0 and 1 provide. In the next steps, you'll add level 2 vertices at a fingertip so you can create a fingernail.

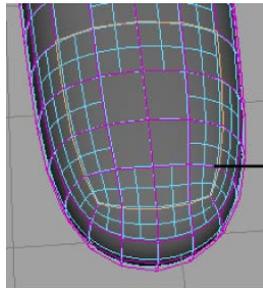
To create more precision in the fingernail region

- 1 Switch to level 1.
- 2 Right-click the surface and select Edge from the marking menu.
- 3 Shift-click to select the edges that surround the region where you'll shape a nail.



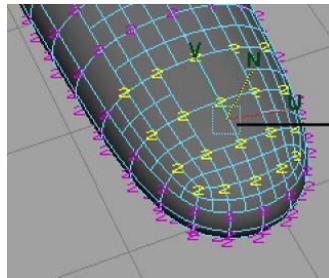
Shift-select the edges that surround the fingernail region

- 4 To increase the amount of detail at the selected edges, select Subdiv Surfaces > Refine Selected Components. Additional surface curves appear in the fingernail area, indicating you can edit with more precision.



Additional curves appear to allow for more precise editing

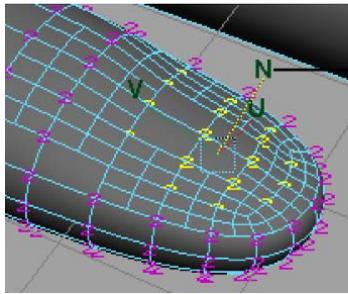
- 5 In the scene view, right-click LeftHand and select Vertex from the marking menu. This displays additional level 2 vertices in the fingertip region.
- 6 Select Modify > Transformation Tools > Move Tool > . In the Tool Settings window, turn on Normal, then close the window. You can thereafter use the Move Tool manipulator to move vertices in a direction normal (perpendicular) to the surface.
- 7 Select the vertices in the nail region.



Selected vertices

NOTE Be careful to select only those vertices in the nail region. You may need to tumble the camera to ensure that all extra vertices are de-selected.

- 8 Drag the Move Tool's N manipulator down slightly to form the depression of a fingernail bed.



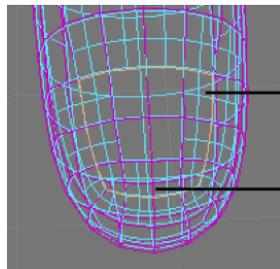
Drag the N manipulator downward to create the depression of a nail bed

Creating a crease in a subdivision surface

One of the unique features of subdivision surfaces is that it's easy to create a crease or ridge on a smooth surface. You'll do this in the next steps.

To crease the edge of the fingernail

- 1 Switch to Edge selection mode and shift-click the edges around the nail to select them. (It might be easier to see the desired edges by first selecting Shading > Wireframe.)

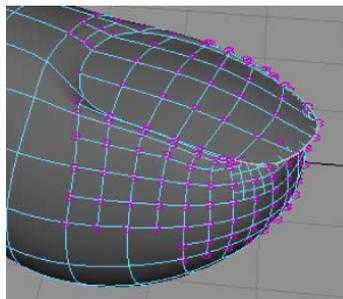


Selected edges

- 2 After you select the edges, select Subdiv Surfaces > Full Crease Edge/Vertex. Once you crease an edge, it displays with a dashed line as a visual reminder that the edge has been creased. If you change your mind, you can remove the crease using Subdiv Surfaces > Uncrease Edge/Vertex.
- 3 To create a ridge at the edge of the nail, switch to Vertex selection mode and select the level 2 vertices on the skin at the perimeter of the previously selected edges. Move them up so that the skin's juncture with the nail forms a slight ridge.

- 4 Now experiment on your own by moving individual vertices to shape the nail and surrounding region as desired. A few suggestions follow:
- In Smooth Shade display mode (Shading > Smooth Shade All), certain vertices might be below the shaded surface and therefore impossible to select and move. To display and select such vertices, switch to wireframe shading (Shading > Wireframe). After selecting the vertices, return to Smooth Shade display mode so you will be able to more easily see the results of moving them.
 - You might get better results using the Move Tool's default World setting rather than the Normal setting made in a prior step. To return to the default setting, select Modify > Transformation Tools > Move Tool > . In the Tool Settings window, click Reset Tool, and then close the window.
 - To extend the front of the nail past the skin with a sharp edge, add a second crease on the edges below the tip of the nail. (With the appropriate edges selected, select Subdiv Surfaces > Full Crease Edge/Vertex.)
You might also want to add an additional level of vertices at the edges. (With the appropriate vertices selected, select Subdiv Surfaces > Refine Selected Components.)
- After you crease the edges and add the extra vertices, you can make the nail's edge overhang the skin below it by dragging the vertices at the nail tip out and away from the finger. Then tuck the row of vertices below the overhanging vertices in the opposite direction.

An example of a completed nail follows:



Completed fingernail

- 5 If you want more practice, repeat the procedure for the rest of the nails.
- 6 When finished, select Modify > Transformation Tools > Move Tool > . In the Tool Settings window, click Reset Tool, and then close the

window. This returns the Move Tool to its default settings. This will avoid confusion in future lessons.

Beyond the lesson

In this lesson, you were introduced to some basic techniques related to subdivision surface modeling:

- Modeling with subdivision surfaces is an easy way to create intricate, smooth objects such as human hands and faces. The ability to create extrusions, subtle surface alterations, and sharp edges means there are innumerable types of objects you can model.
- Many of the techniques used in modeling polygonal surfaces are applicable to subdivision surfaces.

There are many other features related to subdivision surfaces not covered in this lesson. These include the ability to:

- Convert a NURBS surface to subdivision surface.
- Convert a subdivision surface to a polygonal or NURBS surface.
- Create subdivision surface primitive shapes from the Create > Subdiv Primitives menu: Sphere, Cube, Cylinder, Cone, Plane, and Torus. With these shapes, you can quickly start your subdivision surface model without converting from polygons or NURBS surfaces.
- Attach two subdivision surfaces to form a single surface, for instance, to merge a hand and an arm.
- Create a mirrored copy of a subdivision surface.
- You can sculpt subdivision surfaces using the Sculpt Geometry Tool. If you want to learn more about a particular tool or feature that has been presented in this lesson, refer to the Maya Help.

Introduction

Maya lets you apply action to the objects in your 3D scene. In Maya, when an object or attribute changes in relation to time, it is referred to as being animated.

Maya provides a large selection of tools to help you animate the objects in your scene. You may decide to use a combination of several techniques to achieve your desired results.

In this chapter, you learn some common techniques and features that highlight Maya's computer animation technology:

- Lesson 1 Keyframes and the Graph Editor: [Introduction](#) on page 204
- Lesson 2 Set Driven Key: [Introduction](#) on page 219
- Lesson 3 Path animation: [Introduction](#) on page 224
- Lesson 4 Nonlinear animation with Trax: [Introduction](#) on page 241
- Lesson 5 Inverse kinematics: [Introduction](#) on page 276

Preparing for the lessons

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

- 1 Select Window > Settings/Preferences > Preferences. Click the Settings category and set the Time option to Film (24 fps) so your animation plays at the rate of 24 frames per second. Click the Save button.

- 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](#) on page 25.
- 3 Select the Animation menu set. Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the Animation menu set.

Lesson 1: Keyframes and the Graph Editor

Introduction

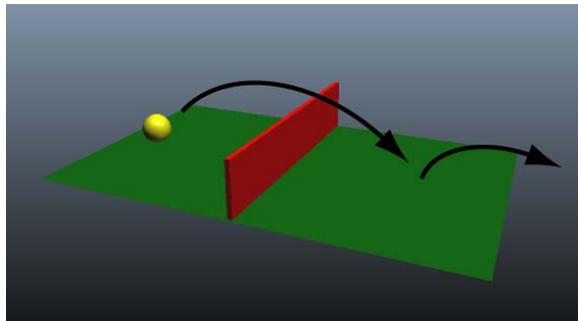
When you set a keyframe (or *key*), you assign a value to an object's attribute (for example, translate, rotate, scale, color) at a specific time.

Most animation systems use the *frame* as the basic unit of measurement because each frame is played back in rapid succession to provide the illusion of motion.

The frame rate (frames per second) that is used to play back an animation is based on the medium that the animation will be played back (for example, film, TV, video game, and so on.)

When you set several keys at different times with different values, Maya generates the attribute values between those times as the scene plays back each frame. The result is the movement or change over time of those objects and attributes.

In this lesson, you will use simple keyframing techniques to make a ball fly over a fence and bounce off the ground.



In this lesson, you learn how to:

- Set keyframes for animatable objects and their attributes.
- Use the Time and Range slider and Playback Controls to control the playback.
- Use keyboard shortcuts to set keyframes.
- Use the Graph Editor to view animation curves.
- Modify the animation of objects using the Graph Editor.
- Set preferences to increase the playback quality.

Watch:

Setting the playback range

In this lesson, you work with a scene we've created for your use. In the following steps, you open the scene and set how long the animation will play.

To open the scene

- 1 Make sure to do the steps in [Preparing for the lessons](#) on page 203.
- 2 Open the scene file named `Keyframing.mb`.

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

`GettingStarted/Anim/Keyframing.mb`

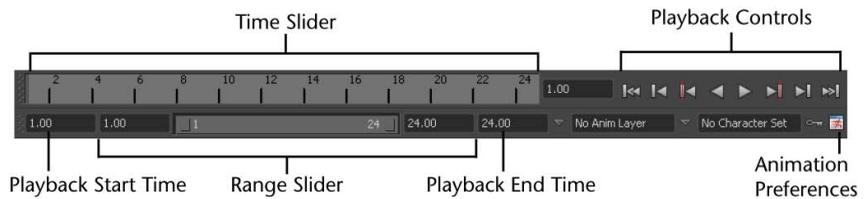
The scene contains a ball object that is currently positioned on the X-axis near the edge of a ground plane.

To animate the ball, you key its position at different times of the playback range.

The playback range is defined by the Time and Range slider. The Time and Range slider controls allow you to playback or scroll through your animation or move to a specific point in time of your animation so you can set keyframes.

To set the playback range for the scene

- 1 Look over the playback controls, as shown in the figure below:



The Time Slider displays the playback range and keys you've set for a selected object. Keys are displayed as red lines. The box at the right of the Time Slider lets you set the current frame (time) of the animation.

The Playback Controls control animation playback. You may recognize the conventional buttons for play and rewind (return to the start time). The stop button appears only when the animation is playing. To find out which operation a button represents, hold the mouse pointer over it.

The Animation Preferences button displays a window for setting animation preference settings such as the playback speed.

The Range Slider controls the range of frames that play when you click the play button.

The above items will have more relevance as you work through this lesson. After you complete this lesson, experiment with these items to learn more what they do.

The playback range is currently set at a range of 1 to 24. At a default playback rate of 24 frames per second, the scene can play for one second. Because you'll animate the ball for a few more seconds than this, you need to lengthen the playback range.

- 2 In the Playback End Time box, enter 72.

A frame rate of 24 frames per second (fps) is the frame rate used for motion picture film. For video, the frame rate can be 30 fps (NTSC) or 25 fps (PAL) depending on the format being used.

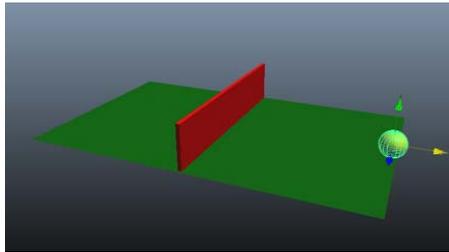
With a playback range of 1 to 72, you'll be able to create three seconds of animation. (72 frames divided by 24 frames per second = 3 seconds.) This is enough time for the short animation you'll create in this lesson.

Setting keyframes

In the following steps, you use keyframes to set the starting and ending positions of the ball's movement.

To set beginning and ending keyframes

- 1 Click the Go To Start button to go to the start of the playback range. This changes the current frame to 1.
- 2 Select the ball, then select **Animate > Set Key**. (Keyboard shortcut: s).
This sets a key at frame 1 for all *transform attributes* of the ball. Transform attributes are the X, Y, Z move attributes. Although you animate only the translate X and Y attributes of the ball in this lesson, keying all transform attributes saves you time having to choose specific attributes to be keyed.
In the Time Slider, notice the red marker at frame 1, known as a tick. This tick appeared when you set the key for frame 1. With the ball selected, ticks in the Time Slider indicate where you've set keys.
- 3 Go to frame 72. A convenient way to do this is to click the desired position in the Time Slider.
- 4 With the Move Tool, drag the ball's X-axis handle to position the ball at the right edge of the ground as shown in the image below.



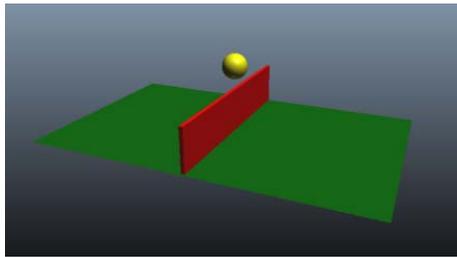
- 5 Set a key at frame 72. (Press s.)
- 6 Go to the start time and play the animation.
From the two keys you've set, Maya creates motion between the positions. By default, the animation plays in a loop from frame 1 to 72. The ball travels through the fence at this stage.
- 7 Press the Stop button on the playback control to stop the animation after you view a few repetitions.
You can drag the mouse back and forth (scrub) in the Time Slider to see the animation play back and forth at the speed you drag the mouse.
If you were to display the scene with **Panels > Layouts > Four Panes**, only the active panel would show the ball moving.

To make the ball fly over the fence rather than pass through it, you need to position the ball above the fence and set a key there.

To set intermediate keyframes

- 1 Go to frame 33 or so—at the moment where the ball sits in the middle of the fence.
- 2 With the Move tool, drag the Y-axis handle of the ball until it sits slightly above the fence.

TIP Throughout this lesson, tumble the perspective view or examine a front view to make sure the positioning is correct.

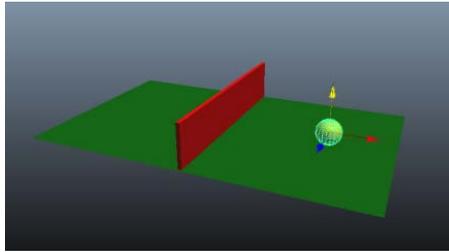


- 3 Set a key. (Press s.)
- 4 Play the animation.
The ball now flies off the ground, over the fence, and back to the ground in a smooth arc between the keyed start, middle, and end positions.
- 5 Press the stop button to end the playback.
In subsequent steps, play the animation after each key you set. It's generally useful to check your work in progress after each key, especially when you are learning.

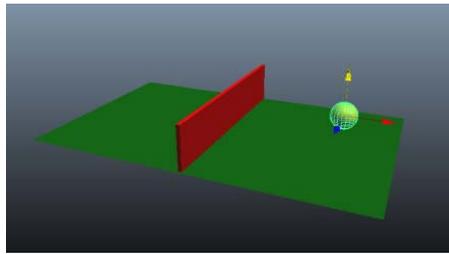
In the next steps, you'll set keys to bounce the ball in the middle of the right half of the ground.

To set keyframes to make the ball bounce

- 1 Go to frame 50. At this moment, the ball sits in a position above the middle section of the right half of the ground.
- 2 Move the ball so it sits on the ground.



- 3 Set a key.
- 4 Go to frame 60.
- 5 Move the ball up again, but not as high as its peak height above the fence.



- 6 Set a key. When you play the animation, the ball travels over the fence and bounces on the other side.

If you have a fast computer, you might notice that the animation plays too fast. By default, Maya plays the animation as fast as it can be processed. Because this scene is simple, the animation might play faster than the default film rate (24 frames per second).

Do not be concerned that the animation plays with a halting or jerky motion. When you render all the frames of your animation for final production, the animation will be smooth. If you want to preview the animation at the smooth production speed (or nearly so), use Window > Playblast.

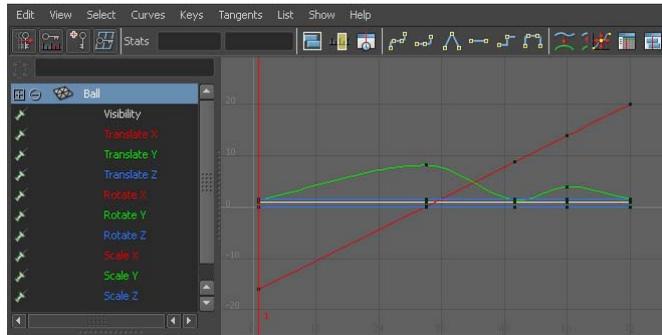
Using the Graph Editor

For a more convincing animation, you need to bounce the ball more sharply off the ground and speed up the ball's horizontal movement. You'll use the *Graph Editor* to make both modifications.

The Graph Editor is an editor that graphically represents the various animated attributes in your scene. The animated attributes are represented by curves called *animation curves*. You edit animation curves in the Graph Editor.

To edit animation curves using the Graph Editor

- 1 With the ball selected, select Window > Animation Editors > Graph Editor.



The Graph Editor displays several *animation curves*, one for each keyed attribute of the ball. The animatable attributes for the ball are listed in the left column. Specifically, it displays the attributes of the selected transform node of the ball.

Each curve graphs how an attribute changes value during the animation. The column of numbers at the left represents attribute values that can be animated, while the row of numbers at the bottom represents time (frame) values. At each point on the curve, you can see the value of an attribute at a particular time. The small black squares on the curves represent points where you've set keys.

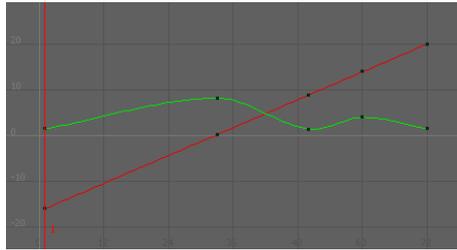
- 2 Shift-Select only the Translate X and Translate Y attributes in the left column of the Graph Editor.

Now only the animation curves for Translate X and Translate Y appear in the Graph Editor.

This simplifies the display. With too many curves present, it's hard to see specific curves. When you use the Graph Editor, you usually focus on one or a few curves.

TIP To change the background or other colors in the Graph Editor, select Window > Settings Preferences > Color Settings, then expand the Animation Editors heading on the General tab. These settings let you change the Grid Axes, Grid Lines, and Background.

- 3 To center the display of the animation curves, select View > Frame Selection (in the Graph Editor window). If you want to see more detail in the graph, use your mouse to dolly and track the graph view.



The green curve represents Translate Y, while the red represents Translate X. The color of each curve matches its attribute name. This color scheme is consistent throughout Maya for X, Y, and Z (red, green, blue).

If you've never used a graph editor before, the relationship between a curve's shape and the animation it represents might be hard to understand. With experience, you'll quickly recognize how curve shape affects animation.

In the above graph, the straight curve of Translate X indicates that the ball moves horizontally across the ground at a constant rate over time.

The wavy curve of Translate Y indicates that the ball increases its height steadily until frame 33, dips more quickly until frame 50, then rises and dips again until frame 72.

When the ball first bounces off the ground at frame 50, it seems to float and slip rather than rebound. The shape of the Translate Y curve illustrates why. Near frame 60, the curve is soft and rounded. The Translate Y values gradually decrease to the low point then gradually increase. The transition from decreasing to increasing values is smooth.

The Graph Editor's usefulness lies in the fact that you can edit the shape of the animation curves to edit the animation of any keyed attribute. To create a sharp bounce, you can edit the curve so that the transition from decreasing to increasing values is abrupt at frame 50. Specifically, you'll create a corner at that key point rather than a rounded curve.

- 4 Select the point on the Translate Y (green) curve at frame 50.

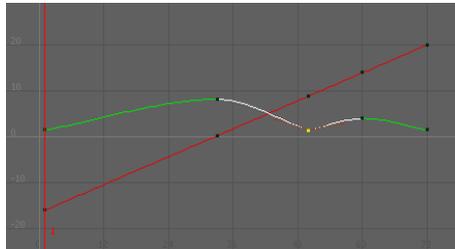
This causes a pair of tangent handles to appear at the point. Each end point of the newly displayed straight line is a tangent handle. The handles let you control the curvature near the key point.

Animation curves have a few control structures for curve editing. You can do any of these operations:

- Use the Move tool and middle mouse button to move a key point. You can alternatively use your keyboard to enter precise values for a selected key's frame and value in the boxes above the graph area.
- Use the Move tool and middle mouse button to drag the tangent handles and change the adjacent curvature. (You can shift-drag with the Move tool to constrain a move operation to a single direction.)
- Use the Scale tool and middle mouse button to scale selected key points closer together or further apart.
- Use the Graph Editor's Tangents menu items to change the adjacent curvature. This is the operation you'll use in the next step.

5 In the Graph Editor, select Tangents > Linear.

This changes the curvature around the key point from rounded to cornered. Specifically, the setting you select specifies how the key point tangent handles lie at this key point. This affects the type of interpolation between key points.



6 Play the animation and you'll see the ball bounce more sharply.

To sharpen the bounce more, you need to edit the positioning of the tangent handles to steepen the curvature approaching the key point.

7 Select one of the tangent handles.

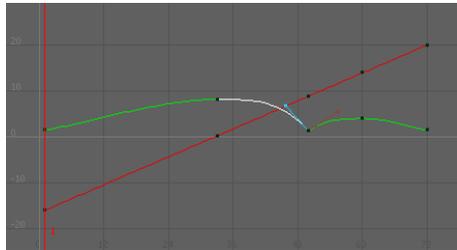
8 Use the Move Tool and the middle mouse button to drag it upwards a little so you can see how the curvature at the key point changes.

By default, when a pair of tangent handles share a key point, they work as a unit. Move one and you move the other in an opposing direction. This is often desirable in an animation curve, because it ensures the

curvature at that point stays symmetrical. Symmetrical curvature often helps prevent unusual animation shifts.

In this case, however, you want to steepen the curvature's approach toward the curve point in the same direction on both sides. You therefore need to break the symmetrical interdependency between the two tangents.

- 9 Undo your previous move of the tangent handle.
- 10 Select the key point at frame 50 (not a tangent handle).
- 11 In the Graph Editor, select Keys > Break Tangents.
This lets you move each tangent handle independently.
- 12 Select the right tangent handle and use the Move Tool to move it up a little. Do the same for the left tangent handle. Be careful not to select the key point.



This sharpens the bounce to better simulate the effect of gravity and elasticity.

Changing the timing of an attribute

The ball seems to move too slowly in its journey. In the following steps, you'll use the Graph Editor to hasten its movement. Specifically, you'll cause the ball's animation to finish its journey in two seconds (48 frames) rather than three (72 frames).

TIP Widen the Graph Editor window as necessary to see the graph clearly. Also, dolly the view or select View > Frame Selection (in the Graph Editor window).

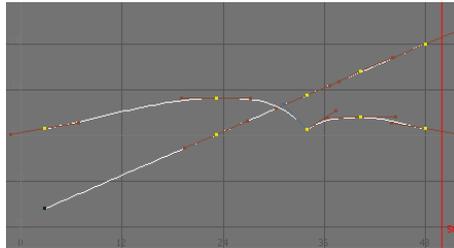
To speed up the animation of the ball's movement

- 1 Drag a selection box around all key points of both curves.
- 2 Select the Scale tool.

- 3 Click and hold the middle mouse button anywhere in the graph at frame 1 (slightly to the right of 0).

Notice the question mark icon. The location of this icon indicates the point from which scaling occurs. (Because you need to scale the animation inward toward frame 1, you need to start your drag operation at frame 1.)

- 4 Without releasing the mouse button, drag to the left until the right-most key points on both curves are positioned roughly at frame 48. (When selected, the right-most key points are yellow.)



This scales the curves symmetrically towards frame 1. The animation of the ball now plays from frame 1 to 48 rather than 1 to 72. The ball traverses the scene in less time, which, of course, means it moves faster.

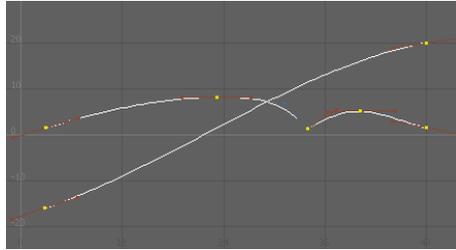
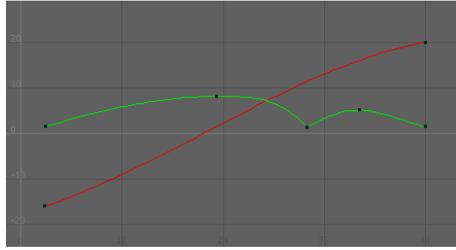
If you had started your drag operation from the middle of the frame range, the scale operation would have pulled the curves toward the middle of the frame range. You can snap the keyframes to whole values to keep the keyframes on a consistent timing.

Fine tuning an animation

Now you can improvise adjustments to the animation on your own. For example, you might choose to speed the ball's horizontal movement up to the first bounce but not thereafter. You might also decide to raise the peak height of the ball's motion so that it doesn't seem to hover over the fence unrealistically.

Whatever you choose to do, you'll likely need to edit both the Translate X and Translate Y curves, not just one of them. Play the animation after each adjustment.

The following two figures show examples of curves after modification. The two graphs are identical, except the curve points are selected in the second figure. The second graph shows the position of the tangent handles.



If you want your curves to match the shape of the curves in the above figure, here's what you would need to do:

To adjust the animation curves to match the above images

- 1 On the Translate X curve, delete the three key points between the two key points at the end. To delete keys, select the key points and press Delete. (The preceding figures show the curve after the points were deleted.) To delete the points, drag a selection box around them and press Delete.

With fewer key points on the curve, it's easier to maintain the curve's smoothness for larger distances as you edit its shape. Small kinks in an animation curve can ruin an otherwise perfect animation, so it's useful to remove key points that you aren't using.

- 2 On the Translate X curve, select the left-most key point and move its right tangent handle down slightly. (Remember to use the middle mouse button when moving a handle or point.) Select the right-most key point and move its left tangent handle up slightly.

Notice that the slope of the Translate X curve increases slightly in the early part of the animation, then tapers off toward the end. This causes the ball to accelerate slightly at the beginning of the animation, then decelerate after its first bounce.

- 3 On the Translate Y curve, remove the key point at frame 22. (The preceding figures show the curve after the point was removed.) This point wasn't essential to the curve shape.
- 4 On the Translate Y curve, select the left-most key point and move its right tangent handle up slightly.
- 5 At the key point where the bounce occurs on the Translate Y curve, move the left tangent handle up a bit. This raises the high part of the curve left of that key point, which makes the ball rise higher over the fence.
- 6 If you want to change the shape of Translate Y curve but lack adequate control with the existing key points, you can add a key point at the position of your choice. (No key points were added in the prior figures.)

To add a key, click the Add Keys Tool icon , select the curve, and middle-click the desired position on or off the curve.

Deleting extra keyframes and static channels

As you set keys, you often create many unintended keys. For example, when you used the Set Key operation on the ball, Maya created keys on all transform node attributes of the ball, for instance Rotate Z, not just the intended Translate X and Translate Y attributes. The curves representing such attributes have unchanging values. The attributes are known as *static channels*.

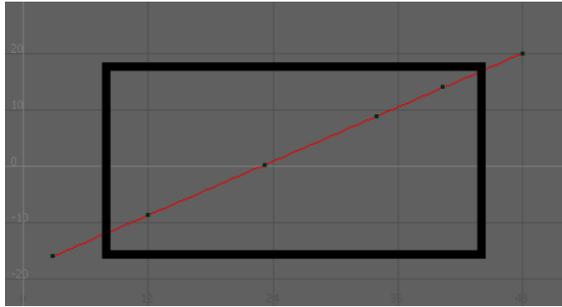
You can identify static channels by examining animation curves in the Graph Editor. If a curve is horizontally flat its whole length, the value of the attribute it represents isn't changing. The attribute is a static channel. Static channels slow Maya processing, so it's beneficial to remove them in complex scenes.

To delete static channels

- From the main menu, select Edit > Delete by Type > Static Channels. See Edit > Delete by Type > Channels, Edit > Delete by Type > Static Channels, Edit > Delete by Type > Non-particle Expressions for more information.

This deletes all unnecessary keys for all objects in the scene.

In addition to static channels, you'll often create excess keys—keys that aren't being used to control the shape of a curve. Whenever you see three key points that lie in a straight line, the one in the middle is unnecessary. For example, in this lesson's original Translate X curve, there were four redundant keys:



When you remove redundant key points, the shape of the curve doesn't change and you speed up Maya processing.

Using Playblast to playback an animation

When you play a scene, you see a rough approximation of the animation. The playback speed and quality is imprecise as compared to rendering each frame and playing the frames in sequence with a playback utility specifically designed for this purpose. The reason for this is that Maya processes the animation of each frame before display, while a playback utility simply plays previously processed and rendered images.

The more detail that Maya needs to process, the slower the playback. For example, playback will likely be slow for any scene that has several complex objects displayed with Smooth Shade All and Hardware Texturing.

To preview the animation with more accurate timing

- 1 Select Window > Playblast.
For the next several seconds, Maya processes the entire animation frame by frame. When the process is complete, a playback window appears.
- 2 Click the play button in this window to watch the animation.
- 3 Close the Playblast window when you're done.

Beyond the lesson

In this lesson, you learned how to:

- Set keys for attributes, then use the Graph Editor to refine the animation and remove unnecessary keys. This is a typical workflow when keyframing.

You can animate most any attribute in Maya, not just the Translate, Rotate, and Scale attributes of a surface. For example, you can animate the intensity of a light, the transparency of a surface, the rotation of a camera view, or the position of CVs.

- Work with the Graph Editor to adjust the animation attributes animation curves directly.

The Graph Editor is the best tool for editing keyed animation by reshaping animation curves. You can use it to change key positions, add or remove keys, and alter the fluidity and symmetry of animation.

Each key point on an animation curve has a Tangent setting that specifies the curvature leading into that point. The default Tangent setting (Spline) creates rounded curvature—useful for animating attributes that change smoothly over time. The Linear setting creates angular curvature, useful for a bounce effect or any other abrupt change in attribute value. Other Tangent settings are available for different animation effects. For example, a Stepped setting lets you make an instantaneous leap in value, useful when you want to flash a light off and on.

- Use the Time and Range slider to play back your animation. Although the Graph Editor is a popular animation tool, you can also cut, copy, paste, and delete keys directly in the Time Slider to edit animation. Select a key position in the Time Slider, then right-click to select the desired operation from a pop-up menu.

You may find the following techniques useful when keyframing your animations:

- There are alternative object display modes that quicken the display of animation in the scene view. When you select Shading > Bounding Box (from the panel menu in the scene view), Maya displays simple box-shaped geometry in place of the actual objects in your scene. The simpler shapes enable Maya to respond to any camera and object movement faster. The drawback is that you cannot edit the shape of the objects in this display mode.
- It is generally a recommended practice to animate a parent (group) node rather than objects themselves. By animating a parent node, you can avoid problems that occur when the animation of one object in a hierarchy conflicts with the animation in another part of the hierarchy.

Lesson 2: Set Driven Key

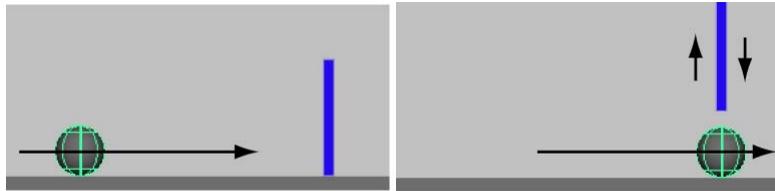
Introduction

With keyframe animation, you key an attribute value to a time in the Time Slider. You repeat this process with different values at different times to animate the object.

When you must animate multiple objects or attributes that interrelate, setting keyframes can quickly become a complex task. *Set Driven Key* is a technique for driving one object's or attribute's animation from another attribute.

With *driven* keys, you relate an attribute value to the value of another attribute. You repeat this with different values to create a dependent link between a pair of attributes. A change in the *driver* attribute alters the value of the *driven* attribute. In this way the animation of the driven attribute doesn't need to be manually set; it occurs automatically once the relationship between attributes has been established. This makes setting up some types of animations much more efficient. For example, you can use driven keys to make a door open when a character walks in front of it.

In this lesson, you will learn how to use Set Driven Key in order to animate a door rising upwards when a ball approaches it.



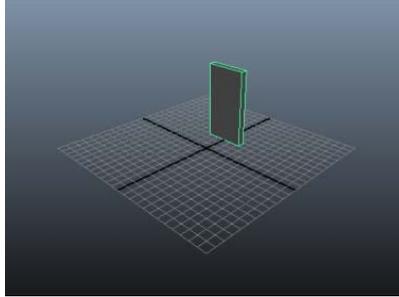
In this lesson you learn how to:

- Link the object attribute behavior between two objects which will link the movement of one keyframed object to another with no keyframes assigned.
- Use the Graph Editor to adjust the animation for the driven object.

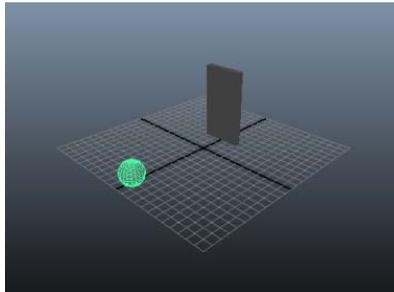
Lesson setup

In the following steps, you set a playback range that's long enough to see the animation clearly. You also create the ball and the door, and position them in the scene view in preparation for the animation.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 203.
- 2 Create a new scene.
- 3 Go to the start frame.
- 4 Create a polygonal cube and name it `Door`.
- 5 With the pointer in the perspective view, press 5 (for Shading > Smooth Shade All).



- 6 Scale and position it roughly as shown. Have the bottom edge of the door lie roughly along the X-axis.
- 7 Create a small polygonal sphere and name it `Ball`. Scale and position it roughly as shown in the prior figure in the introduction. Have it sit roughly at 0, 1, 10 (X, Y, Z).

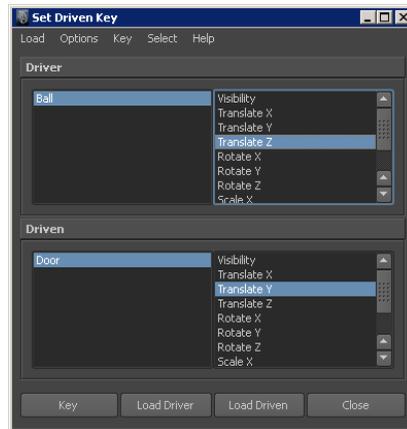


Using Set Driven Key to link attributes

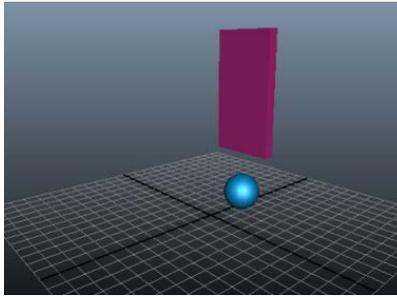
The following steps link the door's movement to the ball's movement using Set Driven Key.

To link the door's movement to the ball's movement

- 1 Select the door, then select Animate > Set Driven Key > Set.
The Set Driven Key window appears with the door in the Driven list.
- 2 Click translateY in the Driven list.
This is the attribute to be driven by the ball's movement.
- 3 Select the ball in the scene view.
- 4 In the Set Driven Key window, click Load Driver.
The ball appears in the Driver list.
- 5 In the Set Driven Key window, click translateZ in the Driver list.
This is the attribute that will drive the door's movement.
You can set a driven key only after you select an attribute in the Driver list and in the Driven list.

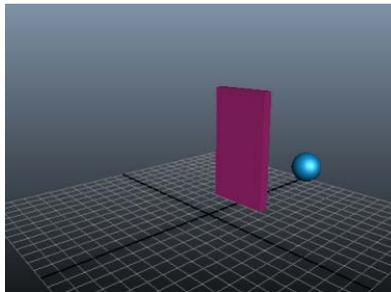


- 6 In the Set Driven Key window, click Key.
This sets a driven key relationship that links the current Translate Z value of the ball to the current Translate Y value of the door. Whenever the ball's Translate Z is at this position, the door's Translate Y will be at its current position.
The Time Slider is not involved in a driven key relationship and displays no red markers for keys.
- 7 Move the ball to the door's position and then move the door above the ball.



NOTE The objects in these images have materials applied to give them color. Try [Lesson 2: Shading surfaces](#) on page 464 in the *Rendering* chapter of the **Getting Started** guide to learn how to apply materials.

- 8 In the Set Driven Key window, click Key.
This sets another driven key that links the current Translate Z value of the ball to the Translate Y value of the door.
- 9 Move the ball to the right of the door and then lower the door to its previous position as shown in the following figure.



- 10 In the Set Driven Key window, click Key.
This sets another driven key that links the attributes.
You've set three driven keys that link the attribute values at different positions. Maya interpolates the linked values to generate the values between the keys.
If you drag the ball toward the door from either side, the door rises smoothly.
Playing back the animation from the Time Slider doesn't move the ball or the door. Driven keys link one attribute to another. They don't link attributes to times in the Time Slider. However, if you were to animate

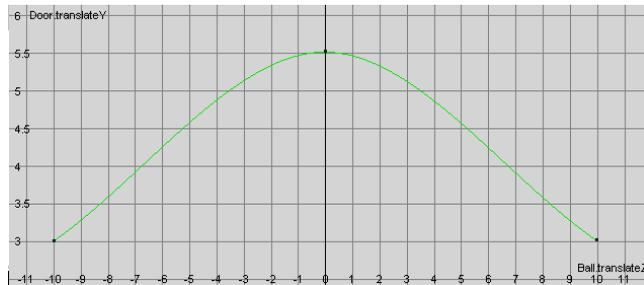
the movement of the ball, the door would move based on the driven key relationship that has been set.

Viewing the results in the Graph Editor

In the next steps you'll examine the animation curve resulting from the driven keys you've set. If you are unfamiliar with animation curves and the Graph Editor, see [Lesson 1: Keyframes and the Graph Editor](#) on page 204.

To view the set driven key results in the Graph Editor

- 1 Select the door.
To display the animation curve for a driven key in the Graph Editor, you must select the object containing the *driven* attribute, not the object containing the driving attribute.
- 2 Select Window > Animation Editors > Graph Editor.
- 3 From the Graph Editor, select View > Frame All. The Graph Editor shows the driven keys that link the door's TranslateY value to the ball's Translate Z value. An example follows:



The column of numbers on the left lists the driven attribute values. The row at the bottom lists driver attribute values. The animation curve shows the relationship between the values—the door's Translate Y value rises as the ball's Translate Z value approaches 0.

You can use the Graph Editor to edit the shape of the curve to make the door rise faster, slower, and higher as the ball approaches it.

Beyond the lesson

In this lesson you learned how to:

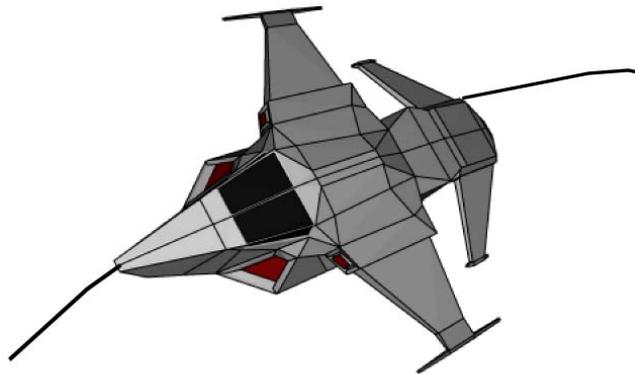
- Set driven keys to link one attribute's values to another.
A driven attribute is not directly affected when you play the animation. It is linked to a driving attribute's value, not animation time. However, you can use regular animation keys, expressions, motion paths, or other techniques to animate the driving attributes. This indirectly animates the driven attribute.

You may find the following notes helpful when using Set Driven Key:

- You can set driven keys to control a driven attribute with multiple driving attributes. For instance, you can have a muscle bulge when an elbow rotates, and have the bulge increase when the wrist rotates.

Lesson 3: Path animation

Introduction



Model created by: Ted Charlton

Path animation allows you to animate an object along a path specified by a curve. In this way, the curve controls the motion of the object. With keyframe animation, Maya calculates the motion of the object for the animation based on the positions that were set at the keyframes. With path animation the motion for the object is defined by its location along the path curve.

Path animation is useful for animating objects such as trains, boats, airplanes, and moving cameras whose motion follows a set path. To animate an object to do this type of motion smoothly with keyframe animation would require you to laboriously create and edit many keys for the motion. By having the object move along a curve, you can easily adjust the object's path by editing the curve.

In this lesson, you make an aircraft follow a motion path so it appears to fly and bank while it changes trajectory. We provide a scene for your use in the lesson.

In this lesson, you learn how to:

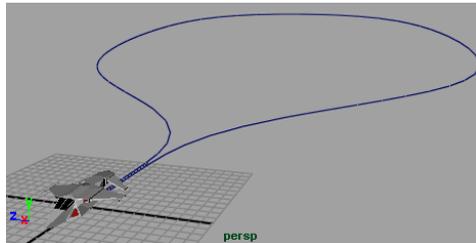
- Set an object to animate along a motion path using a NURBS curve as the path.
- Modify the timing and rotation of an object along a motion path.
- Blend between keyframe and motion path animation.

Open the scene for the lesson

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

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 203.
- 2 Open the scene file named `PathAnim.mb`.

This file can be found in the `GettingStarted` directory that you set as your Maya project: `GettingStarted/Anim/PathAnim.mb`



The scene contains two objects named `Aircraft` and `PathCurve`. When the aircraft travels on the path curve it will return to where it began.

- 3 In the Time Slider, ensure that the Playback End Time is set to 240.



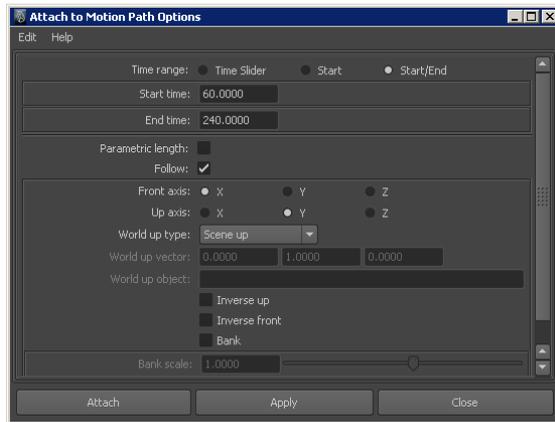
The path animation occurs between frames 60 and 240 (180 frames in total). Between frames 1 and 60, you keyframe the aircraft's motion so it rises from the ground plane. You then blend between the two animation types.

Animating an object along a motion path

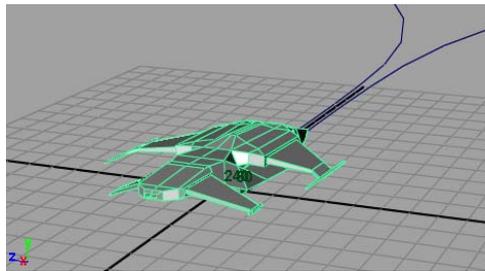
To animate the aircraft along the path curve, first select the aircraft and the path curve and then set appropriate options for the motion path animation. In the options window, you need to set the required time range for the path animation. You also need to ensure that the aircraft is oriented facing towards the direction of travel. The Follow, Front, and Up axis option settings determine the aircraft's orientation along the path.

To attach the aircraft to the motion path

- 1 Go to frame 1 on the Time Slider.
- 2 Select the aircraft and then Shift-select the path curve.
- 3 In the main menu, select **Animate > Motion Paths > Attach to Motion Path >** .
The Attach to Motion Path Options window appears.
- 4 In the options window, ensure the options are set as follows, then click **Attach**:
 - Time Range: Start/End
 - Start Time: 60
 - End Time: 240
 - Follow: On
 - Front Axis: X
 - Up Axis: Y
 - World Up Type: Scene Up
 - Leave all other settings off



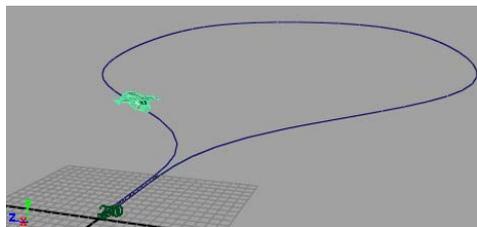
The aircraft is repositioned to the start of the curve, and oriented towards the direction of travel.



The Start and End Times set the time duration the aircraft travels along the path curve (240 - 60 = 180 frames). The start and end times are displayed at the ends of the curve.

- 5 Click play on the Time Slider playback controls to play back the animation.

The aircraft travels along the path. Observe that it begins its motion at frame 60.



Changing the timing of an object along a motion path

When you first assign an object to a motion path, by default, the object travels along the path at a constant speed. In this lesson, you want the aircraft to initially travel along the path slowly, accelerate, and then finally move more slowly when it nears the end of the motion path.

You can change the rate of travel for the aircraft by keyframing attributes that affect where the aircraft is positioned on the curve at a given time.

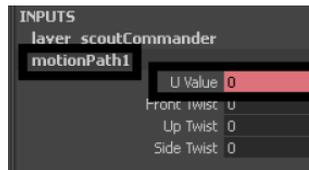
To view the attributes for the motion path

- 1 Go to frame 1 on the Time Slider.
- 2 Select the aircraft.
- 3 Open the Channel Box by clicking the Show/Hide Channel Box icon on the Status Line.

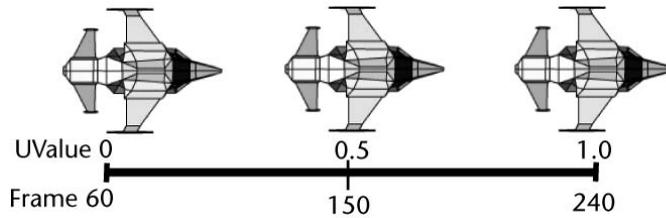


- 4 In the Channel Box, click on the word `motionPath1` that appears in the Aircraft's Inputs list.

The attributes for `motionPath1` appear at the bottom of the Inputs list. (Scroll the Channel Box window if the `MotionPath1` attributes are not fully visible).



The `UValue` attribute controls where the aircraft gets positioned along the curve. The `UValue` refers to the curves *parameterization*. Parameterization is a method used by Maya to divide a curve into increments of known amounts so that a location along the curve can be determined.



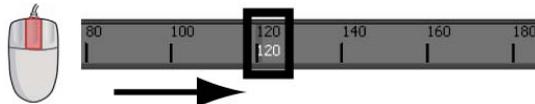
By default, the parameterization of a path curve is set between zero and one. At frame 60, the aircraft is at the beginning of the curve where the parameterization value is zero. When the aircraft is at the end of the curve (frame 240) the parameterization value for the curve is one. When the aircraft is halfway through the path animation (frame 150), it is located along the curve at a UValue of 0.5.

By setting the frame number and UValue and then setting a key, you change the timing for the aircraft's motion along the path.

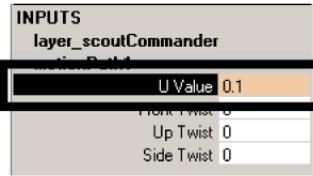
To change the timing for the aircraft along the path

- 1 Ensure the aircraft is selected so its attributes display in the Channel Box.
- 2 In the Time Slider, drag the current time indicator to frame 120 using the middle mouse button.

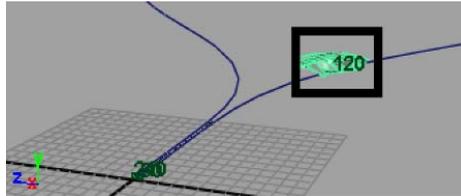
Middle-dragging the current time indicator, instead of using the left mouse button, changes the current time without repositioning the aircraft along the path.



- 3 In the Channel Box, set the UValue for MotionPath1 to 0.1.
The aircraft is repositioned along the curve near the first bend. When you set the UValue to 0.1, the aircraft is positioned at a point that is a distance of 10 percent from the beginning of the curve (that is, the aircraft has completed 10 percent of the distance along the path).
- 4 Select the UValue channel, then right-click the name and select Key Selected from the drop-down list.



A position marker appears on the path curve indicating that a key frame has been set. The position marker is useful for determining where the aircraft is at a given time. Position markers do not appear when you render the animation. Using Key Selected ensures that a keyframe is set only for the item selected and nothing else.



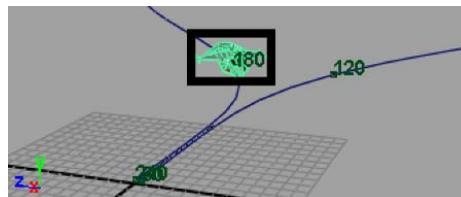
5 In the Time Slider, middle-drag the current time indicator to frame 180.

6 In the Channel Box, set the UValue for MotionPath1 to 0.9.

The aircraft is repositioned along the curve near the last bend. When you set the UValue to 0.9, the aircraft is positioned at a point that is a distance of 90 percent from the beginning of the curve. (that is, the aircraft has completed 90 percent of the distance along the path)

7 Select the UValue channel, then right-click and select Key Selected from the drop-down list.

A position marker appears on the path curve.

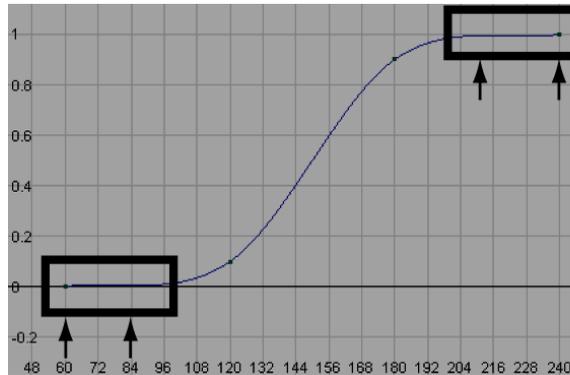


8 Click play on the playback control to play back the animation.

The motion of the aircraft is not smooth as it travels along the path. It initially moves forwards and backwards a bit at the beginning and end of the path. This indicates that some adjustment of the animation is required. You adjust the animation using the Graph Editor.

To view the Graph Editor

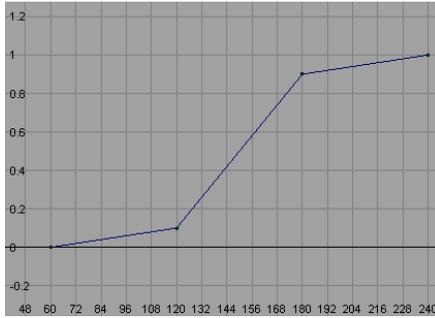
- 1 From the main menu, select Window > Animation Editors > Graph Editor.
The Graph Editor appears.
- 2 In the Graph Editor, select View > Frame All.
The Graph Editor updates to display the animation curve for the motion path animation.



The shape of the animation curve provides some clues as to why the motion of the aircraft is not smooth. Between frames 60 and 84 the curve rises and then drops slightly. Because the animation curve represents translation along the curve, a drop in the curve indicates that the object travels backwards. A similar occurrence happens at the end of the curve as well. To correct this, you need to modify the tangents for the keys in the Graph Editor.

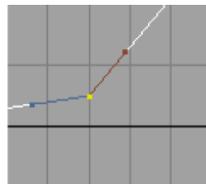
To edit the tangents for the keys in the Graph Editor

- 1 In the Graph Editor, press the shift key, then double click on the animation curve to select the four keys (frames 60, 120, 180 & 240)
- 2 In the Graph Editor menu, select Tangents > Linear.
The animation curve updates so that the keys connect via straight lines. This eliminates the dips in the curve that caused the aircraft to travel backwards.

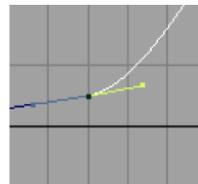


Next, you modify the tangents for keys 120 and 180 so the aircraft accelerates and decelerates near those points on the path curve.

- 1 In the Graph Editor, shift-select only the two keys for frames 120 and 180.
- 2 In the Graph Editor menu, select Keys > Break Tangents.
Breaking the tangent for the keys allows you to modify the shape of the curve as it enters or leaves a key.
- 3 Select only the key for frame 120.
A pair of tangent handles appear on either side of the key. The handles let you edit the curvature of the animation curve near the key point.
- 4 Select the tangent handle to the right of the key.
The handle highlights in a yellow color.
- 5 From the Toolbox, select the move tool.
- 6 Using the middle mouse button, drag the handle so the curve has a gentle curvature as it leaves the key as shown below.
Modifying the tangent in this manner changes how the motion transitions as the curve changes direction as it passes through the key. The aircraft will accelerate smoothly instead of immediately travelling at a different rate of speed.

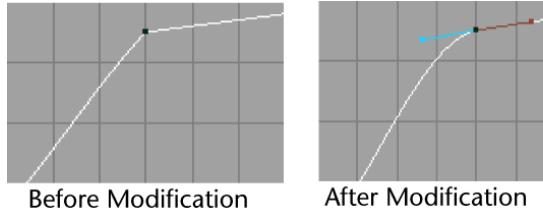


Before Modification

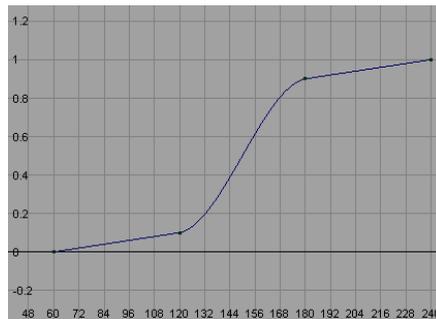


After Modification

- 7 Select the key for frame 180.
- 8 Select the handle to the left of the key.
The handle highlights in blue.
- 9 Using the middle mouse button, drag the handle so the curve has a gentle curvature as it enters the key as shown below.
Modifying this tangent for this key will cause the aircraft to decelerate smoothly near the end of its travel.



The modified animation curve appears as shown below.



- 10 Close the Graph Editor.
- 11 Click play on the Time Slider's playback control to play back the animation.
The backwards and forwards motion that previously occurred is corrected. The aircraft travels slowly towards the first bend in the path curve, then accelerates and travels at a constant speed around the curve until it nears the end of the path where it decelerates and then slowly moves towards the end of the curve.

Rotating an object along a motion path

As the aircraft travels along the motion path, its orientation remains the same throughout the animation. A real aircraft rolls to one side (banks) as it flies in an arc. To make the aircraft roll to one side as it travels along the path you set keyframes for the Twist attributes in the Channel Box.

To animate the roll of the aircraft along the path

- 1 Ensure the aircraft is selected.
- 2 Set the Time Slider to frame 120.
The aircraft is repositioned along the curve near the first bend.
- 3 In the Inputs section of the Channel Box, click on motionPath1 to view its attributes.
- 4 Ensure the Front Twist value for motionPath1 is set to 0.
- 5 Select the FrontTwist channel, then right-click and select Key Selected.



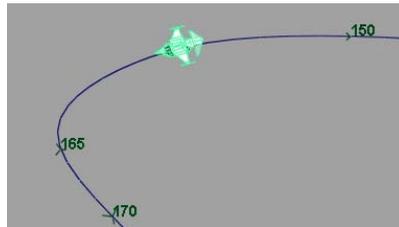
- 6 To set the remaining keyframes for the rotation of the aircraft, use the table below as a guide, keeping in mind that you set the keyframes in the following order:
 - Set the frame in the Time Slider
 - Set the values for Front Twist in the Channel Box.
 - Select the Front Twist channel by clicking its name, then right-click and select Key Selected.

Frame	Select	Set channel attribute	Choose
130	Aircraft	Front Twist: -15	key selected
135	Aircraft	Front Twist: -45	key selected
150	Aircraft	Front Twist: -70	key selected
165	Aircraft	Front Twist: -45	key selected

Frame	Select	Set channel attribute	Choose
170	Aircraft	Front Twist: -15	key selected
180	Aircraft	Front Twist: 0	key selected

- 7 Click play on the Time Slider playback controls to play back the animation.

The aircraft rolls to one side (banks) as it travels along the motion path.



Blending keyframe and motion path animation

Blending between motion path and keyframe animation types is possible in Maya. Blending between the two animation types allows you to take advantage of the characteristics each animation type can provide without having to expend a lot of effort in determining where the change between the two types must occur.

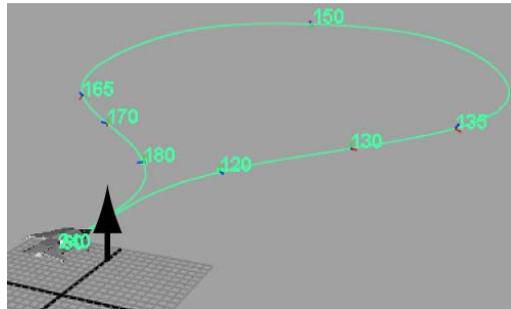
In the steps that follow, you keyframe the aircraft to rise vertically from the floor surface and then travel along the motion path by blending between the two animation types. To accomplish this, you do the following:

- Move the motion path up above the ground plane.
- Keyframe the aircraft so it rises vertically above the ground plane.
- Blend between the keyframe and motion path animations.

To move the motion path

- 1 In the scene view, select only the path curve.
- 2 In the Channel Box, set Translate Y to 5 for PathCurve.

The path curve is repositioned above the ground plane in the scene view. The aircraft moves with the path curve because it is attached to the motion path.



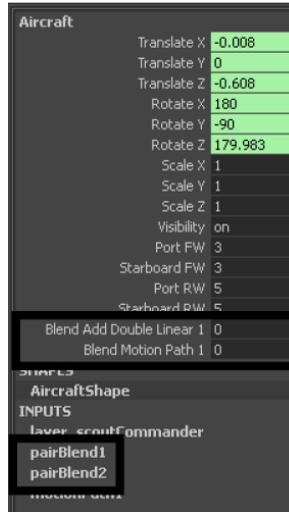
To keyframe the motion for the aircraft

- 1 Select only the aircraft.
- 2 Set the Time Slider to frame 1.
- 3 In the Channel Box, set Translate Y to 0.
The aircraft is repositioned so it rests on the ground plane.
- 4 Select the Translate X, Y, and Z channels as well as the Rotate X, Y, and Z channels by click-dragging on their names so they become highlighted.

Aircraft	
Translate X	-0.008
Translate Y	0
Translate Z	-0.608
Rotate X	180
Rotate Y	0.454
Rotate Z	179.983

- 5 Right-click on any of the highlighted names, and choose Key Selected from the drop down list that appears.
Keys are set for the selected channels. Keyframing these channels establishes a blending connection that will be discussed in subsequent steps.
- 6 In the Channel Box, set Rotate Y to -90.
The aircraft is rotated 90 degrees from its motion path position.
- 7 Select the Rotate Y channel, then right-click and select Key Selected from the drop-down list.
A keyframe is set for the aircraft.

Additional channels appear in the Channel Box. When a keyframe is set for an object that already has motion path animation, the software automatically creates a connection to blend between the two animation types. These new channels are used to control the blend between the keyframe and motion path animation types. They control the blend for the Translation and Rotation attributes.



8 To set the remaining keyframes for the keyframe motion of the aircraft, use the table below as a guide, keeping in mind that you set the keyframes in the following order:

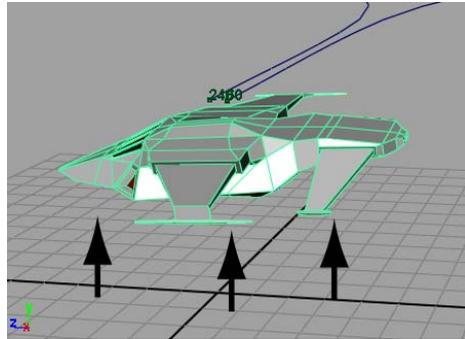
- Set the frame in the Time Slider.
- Set the Translate and Rotate values in the Channel Box.
- Select only the channels for those values. Right-click on the selected name and choose Key Selected.

Frame	Select	Set channel attribute	Choose
20	Aircraft	Translate Y: 1	key selected
30	Aircraft	Translate Y: 2	key selected
30	Aircraft	Rotate Z: 170	key selected
40	Aircraft	Translate Y: 3	key selected

Frame	Select	Set channel attribute	Choose
40	Aircraft	Rotate Z: 170	key selected

- Click play on the Time Slider playback controls to play back the animation.

The aircraft rises above the ground plane and then stops at frame 40.



The animation stops because the motion path animation was shut off when the blend attribute was automatically created when you set the first keyframe. The blend attribute values are set to zero by default. When the blend values are set to zero, the keyframe animation has full influence on the object, and the path animation is shut off. When the blend values are set to one, the motion path has full influence on the object and the keyframing is shut off. When the blend values are between zero and one, the aircraft's motion is blended between the keyframe and motion path animation.

In the next steps, you keyframe the blending attributes so the blending begins at frame 30 and ends at frame 70. This blends the two animation types together over a period of 40 frames.

To blend between the two animation types

- Set the Time Slider to frame 30.
- In the Channel Box, ensure the blend values are set as follows:
 - BlendAddDoubleLinear: 0
 - BlendMotionPath: 0

- 3 Select both the BlendAddDoubleLinear1 and BlendMotionPath1 channels, then right-click on either name and select Key Selected from the drop down list.

A key is set for the blend starting point for the aircraft. At frame 30 the two animation types will begin to blend together.

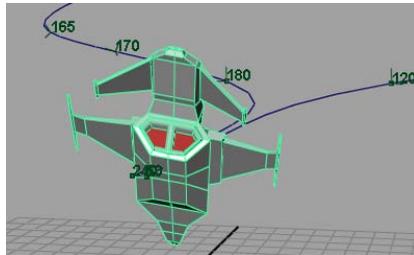
- 4 Set the Time Slider to frame 70.
- 5 In the Channel Box, set the following:
 - BlendAddDoubleLinear1: 1
 - BlendMotionPath1: 1

- 6 Right-click on one of the names, and choose Key Selected from the drop down list that appears.

A key is set for the blend ending point for the aircraft. At frame 70 the motion path animation will have full influence.

- 7 Click play on the Time Slider playback controls to play back the animation.

The two animation types are combined but the motion does not appear correct. The aircraft rotation flips upward momentarily when it begins to travel along the motion path.



The interpolation type for the rotations during the blending is not set correctly.

Setting the pairBlend Rot Interpolation setting to Quaternions changes how the rotations are interpolated during the blend. In this example, Quaternions provides a better interpolation type when blending between the keyframe and motion path animation. For more information see quaternion rotation in the Maya Help.

To edit the rotation interpolation type for blending

- 1 In the Channel Box, click on the pairBlend1 channel.
- 2 Scroll the Channel Box so you can view the Attributes for pairBlend1.
- 3 Set the Rot Interpolation setting to Quaternions by clicking on the Euler angles name with the left mouse button.
- 4 Click on the pairBlend2 channel.
- 5 Click on the Euler angles name in the Rot Interpolation attribute and choose Quaternions.
- 6 Click play on the Time Slider playback controls to play back the animation.

The animation plays back smoothly. Observe how the blending also takes into account the rotation of the aircraft so it meets up with the motion path smoothly.

Using Playblast to playback an animation

Playblast allows you to preview an animation by calculating the separate frames and playing them back in a multimedia player window based on the Playblast options. Playblast provides a more accurately timed playback than using the Time Slider.

To playback the animation using Playblast

- 1 Select the path curve, and then select Display > Hide > Hide Selection to hide the curve while you Playblast.
- 2 Select Window > Playblast.
Playblast calculates the frames for the animation and then plays them back in the player window.

You have completed this lesson.

Beyond the lesson

In this lesson you learned how to:

- Create a motion path animation using a NURBS curve as the motion path.

When you draw your own curves remember that a curve has a defined start and an end point based on how you create it. If your object is initially positioned at the wrong end of the curve, you can reverse the direction of the curve by setting the Menu Set to Modeling and then selecting Edit Curves > Reverse Curve Direction.

If your objects are not oriented correctly on the path initially, you can change the options in the Attach to Motion Path Options window. See Animate > Motion Paths > Attach to Motion Path in the Maya Help.

- Adjust the timing and rotation of an object when it travels along the motion path.

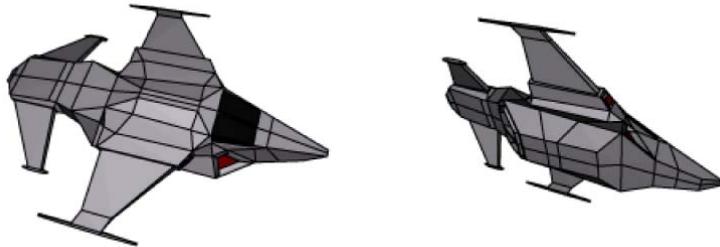
In this lesson you were shown how to keyframe the roll (bank) of the aircraft. The Attach to Motion Path Options window provides an option called Bank. The Bank option rotates an object as it travels around curved portions of a motion path. It is useful for simple motion path animation.

- Blend between keyframe and motion path animation types on the same object.

For more information on how this works, see pairBlend in the Maya Help.

Lesson 4: Nonlinear animation with Trax

Introduction



Nonlinear animation allows you to create and edit an animation sequence by arranging smaller animation sequences (known as *clips*) on a timeline. A clip is a time-independent sequence of animation that you create from an existing animation (keyframe animation, expressions, constraints, and motion capture data - *except* motion path animation).

Nonlinear animation tools in Maya® Trax™ give you the ability to experiment with your animation sequences and character animations without having to redo all of your work when you want to modify or construct a different animation.

Nonlinear animation is a non-destructive process. You can modify a particular clip's attributes without affecting the source animation: the source material remains untouched. This makes nonlinear animation useful when you want to experiment, iterate, or change a particular animation sequence.

The primary tool for working with nonlinear animation is the Trax Editor. You can also store your clips for future use by creating a library of clips.

In this lesson you learn how to:

- Key animation sequences for use as nonlinear clips in Trax.
- Create clips from your animation sequences for use in the Trax Editor.
- Use the Trax editor to position, modify, and arrange clips in order to create and modify animation sequences in the scene.
- Scale clips to modify the timing of an animation sequence.
- Trim clips to remove unwanted motion.
- Cycle clips to create motion that repeats.
- Use the Graph Editor to modify the animation of a clip.
- Use the Trax Editor to work with motion capture data.
- Reuse clips using the Outliner and the Visor.

Open the first scene for the lesson

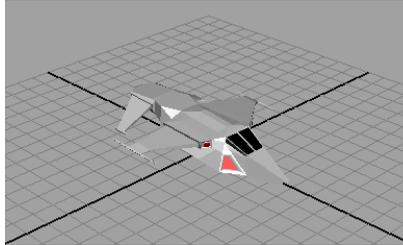
This lesson is presented in two parts. You work with two separate scene files we've created for your use. In the first part of this lesson, you learn some of the fundamental techniques for working with clips in the Trax Editor by animating an aircraft taking off and landing.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 203.
- 2 Open the scene file named `Trax_Lesson1.mb`.

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

GettingStarted/Anim/Trax_Lesson1.mb

The scene contains a model of an aircraft. For the first part of this lesson, you create and work with animation clips to control the motion of the aircraft.



- 3 In the Time Slider, set the Playback range as follows:
 - Playback Start Time: 1
 - Playback End Time: 240

To work more efficiently with the scene and the Trax Editor, you need to set a panel layout.

To set a panel layout for use with the Trax Editor

- From the perspective view's menu, select Panels > Saved Layouts > Persp/Trax/Outliner.

The panel layout updates to display the Perspective view, Outliner, and Trax Editor simultaneously. This allows you to easily work with these windows without having to open and close them repeatedly.

Next, set the Outliner so it displays clips. The Outliner allows you to manage the various clips that get created with Trax.

To set the Outliner to display clips

- 1 In the Outliner menu, make sure that Display > DAG Objects Only is unchecked.
- 2 In the Outliner menu, select Show > Objects > Clips.
The Outliner displays an empty list until you create your first clip.

Creating clips with Trax

A clip is a time-independent sequence of animation data for one or more animated attributes that you create from existing animation (keyframe animation, expressions, constraints, motion capture data, and so on). A clip is time independent, allowing you to position, scale, and cycle a clip independently of other animatable attributes or other clips. This means you can arrange the clips into a particular sequence, and then play it back to see the results. You can adjust the timing, length, and rearrange or edit the same clips in a different order to modify the motion sequence.

Most commonly-used clips are reusable sequences of animation that are easy to sequence or blend with other clips. For example, suppose you create three clips: a karate punch, tumble, and kick. You can run the clips in any order, and you can repeat clips or blend the motion between them. You can scale or cycle clips to alter the time frame over which the animation occurs. You can either return the clip to its original state or reload a duplicate of the original source clip and begin again. In this way, the Trax Editor allows you to experiment with animation sequences on a character.

In the steps that follow, you key some animation for the aircraft and then create clips from the keyed animation for use in the Trax Editor.

To create keyframe animation for the aircraft

- 1 From the perspective view's panel menu, select Panels > Orthographic > front.

The panel updates to display the aircraft's side from the front view.

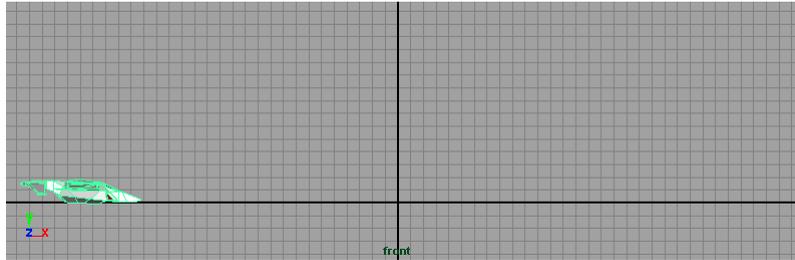
- 2 In the front view, click on the aircraft to select it.

- 3 In the Channel Box, set the following:

- Translate X: -25

The aircraft is repositioned at -25 along the X axis. The aircraft disappears from the front view because it has been repositioned outside of the viewing area.

- 4 Dolly and track the front view until the aircraft appears on the lower left hand side of the front view as shown below.



- 5 Press **s** to set a keyframe.

In the Channel Box, the channels display in a different color. The color change indicates that the channels now have animation applied to them.

NOTE If you normally use the Auto Keyframe function, you must shut it off for this lesson. For more information see Auto Key in the Maya Help.

- 6 In the Time Slider, set the current time indicator to frame 240.

- 7 In the Channel Box, set the following:

- Translate X: 25

The aircraft is repositioned at 25 along the X axis (the right hand side of the current front view).

- 8 In the Channel Box, select the Translate X channel by clicking on the name.

The Translate X name becomes highlighted.

- 9 Right-click on the selected name and choose Key Selected from the drop-down list that appears.

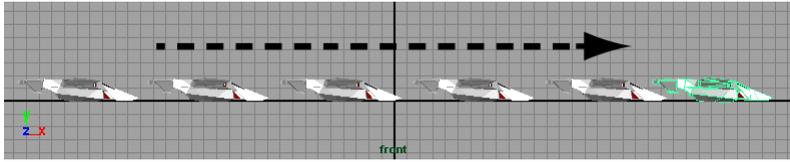
Key Selected sets a keyframe for only the selected attribute.

Confirm that the animation sequence works as expected in the steps that follow.

To playback the animation using the playback controls

- 1 Click play on the Time Slider's playback control to play back the animation.

The aircraft travels from left to right in the front view.



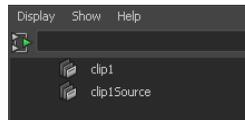
- 2 Click stop on the Time Slider's playback control.

Now that you have keyframed an animation sequence and confirmed that it works, you can create an animation clip.

To create an animation clip of the keyframe animation

- 1 In the Trax Editor menu, with the aircraft still selected, select Create > Animation Clip > .
- 2 In the Create Clip Options window, select Edit > Reset Settings, and then click on the Create Clip button.

The Outliner updates and lists two items: clip1 and clip1Source. If the clips do not display refer to [To set the Outliner to display clips](#) on page 243.



When you created the clip, two clips were generated: a *regular clip* and a *source clip*. The source clip contains the animation curves for the character (based on the keyed attributes that the clip was created from). The source clip is stored with the file and can only be accessed from the Visor or Outliner. Maya saves the source clip outside of the Trax Editor to protect the original animation curves from being accidentally modified and ensures you have an unmodified clip to revert back to.

A regular clip (referred to simply as a clip) contains the same information as the source clip, and is placed in a track in the Trax timeline. It is this regular clip that you work with and modify in Trax.

A clip does not yet appear in the track area of Trax Editor because the character set associated with the clip (the aircraft) has not been loaded to display.

- 3 In the Trax menu, with the aircraft selected, click the Load Selected Characters icon.



The clip appears as a blue rectangular box in a *track* in the Trax Editor. A track is an area in the Trax Editor to place and work with one or more non-overlapping clips in relation to the animation timeline. Tracks allow you to work with your clips in a nonlinear manner by allowing you to reposition and scale clips in time or juxtapose them with other clips. You can work with multiple tracks in the Trax Editor.

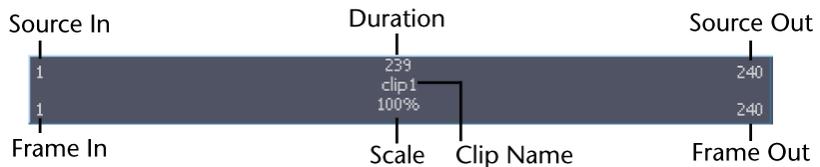


- 4 In the Trax menu, select View > Frame All.

The Trax Editor updates to show the beginning and end point for all of the items displayed in the Trax Editor. (In this case the new clip.) The clip represents the Translate X animation for the aircraft going from -25 to 25, for a time range of 1 to 240 frames.

Anatomy of a clip

Every clip contains interaction controls that relate to information about the clip:



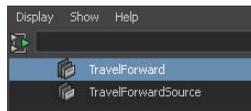
- *Frame In* and *Frame Out* indicates the current start and end frame numbers for the clip in relation to its position on the timeline.
- *Source In* and *Source Out* indicates the range of frames from the original source clip.
- *Duration* indicates the length (duration) of the original source clip in frames.
- *Clip Name*...the name of the clip. A default clip name is assigned to the clip, unless you specify it from the Create Clip Options window when you first create the clip.

- *Scale* indicates the scale percentage for the clip. The scale percentage is the multiplier on the length of the original clip. For example, if you scale a clip fifty percent longer than its original length, the scale value would read 150%.

Each area on the clip has a hot spot which you can either double-click on for numeric values, and in some cases, click-drag to modify the related attribute. For example, you can double-click on the hot spot for the clip name to edit the name. For more information about clips and tracks, see Track view area in the Maya Help.

To edit the name of the clip in the Trax Editor

- 1 In the Trax Editor, double-click on the clip1 name on the clip.
The clip highlights in yellow to indicate it is selected and the clip1 name becomes highlighted.
- 2 Type `Travel_Forward` and then press Enter.
The clip is renamed to `Travel_Forward`. The clip and the source clip's name are updated in the Outliner to reflect the change.



If you play this clip in Trax, the animation is played back from the clip data, not the keyframe data that you set previously in the scene. When you created the clip, the source of the animation for the aircraft was changed to come from the clip instead of the keyframes.

When you create a clip, Maya creates a *character set* to contain the clip. A character set is the parent of the hierarchy of clips and tracks for a particular animated object. A clip's animation resides under the character set node and additional clips are placed under that node unless you specify otherwise.

For more information see Nonlinear animation components in Trax in the Maya Help.

Now that you have created your first animation clip and renamed it using the Trax Editor, you need to create a few more clips by setting additional keys and creating clips of the sequences. You need to create separate clips of the aircraft taking off and landing.

To create an animation clip of the aircraft taking off

- 1 In the scene view, select the aircraft.
- 2 In the Channel Box, set Translate Y to 0.
- 3 Select the Translate Y name, then right-click and select Key Selected to set a keyframe for this position.
Key Selected sets a keyframe for only the selected attribute.
- 4 In the Time Slider, drag the current time indicator to frame 60.
- 5 In the Channel Box, set Translate Y to 7.0.
- 6 Select the Translate Y name, then right-click and select Key Selected to set a keyframe for this position.
- 7 In the Trax Editor menu, with the aircraft selected, select Create > Animation Clip (or click the Create Clip icon on the Toolbar).

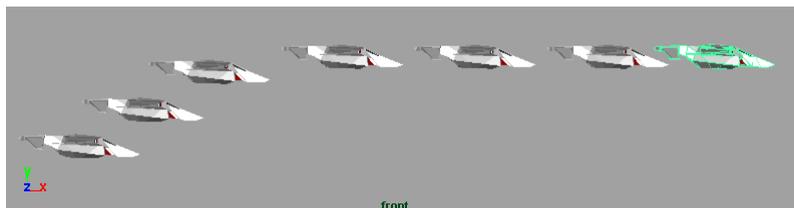


The Trax Editor updates to create an additional track and places the new clip on the new track. The Outliner updates to list the new clip and source clip.

- 8 In the Trax Editor, double-click on the new clip and rename it to Aircraft_Rise.



- 9 Click play on the Time Slider's playback control to play back the animation.

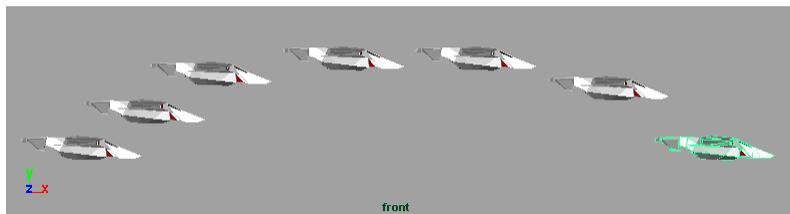


The aircraft rises as it travels forward.

10 Click stop on the Time Slider's playback control.

To create an animation clip of the aircraft landing

- 1 In the Time Slider, set the current time indicator to frame 180.
- 2 In the Channel Box, with the aircraft still selected, set Translate Y to 7.
- 3 Select the Translate Y name, then right-click on the name and select Key Selected to set a keyframe for this position.
- 4 In the Time Slider, set the current time indicator to frame 240.
- 5 In the Channel Box, set the Translate Y value to 0.
- 6 Select the Translate Y name, then right-click on the name and select Key Selected.
- 7 Click play on the Time Slider playback controls to play back the animation.

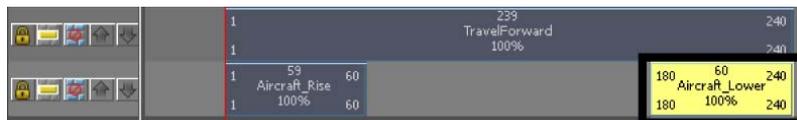


The aircraft travels forward and then lands as it nears the end of its travel.

- 8 Click stop on the Time Slider's playback control.
- 9 In the Trax Editor menu, with the aircraft selected, select Create > Animation Clip.

The Trax Editor and Outliner update to show the new clip.

- 10 In the Trax Editor, double-click on the new clip and rename it to Aircraft_Lower.



Changing the position of clips with Trax

It's easy to alter the start time of clips to synchronize the action between clips. For example, the aircraft rises as it travels forward because the Travel_Forward and Aircraft_Rise clips overlap in time. If you position the clips on their respective tracks so that one clip's action occurs in the Trax timeline before or after the other, the motion occurs separately.

In this section, you learn how to reposition the clips so that the aircraft takes off before it travels forward in X, and then have the aircraft land at the end of the sequence.

To change the position of a clip in a track

- 1 In the Trax Editor, position the mouse cursor over top of the Travel_Forward clip.

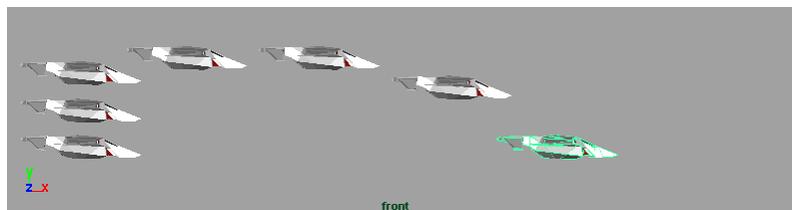
When the mouse cursor is over top of a clip in the Trax Editor, the cursor

icon changes to a Move cursor , indicating that you can drag the clip to either side or up and down between tracks.

- 2 Drag the Travel_Forward clip horizontally to the right in its track until the Frame In reads 60 as shown below.



- 3 Click play on the Time Slider playback controls to play back the animation.

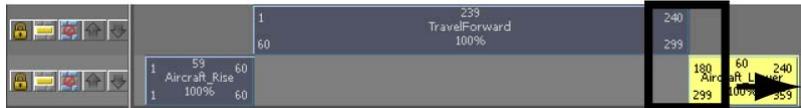


The aircraft takes off vertically, then travels to the right, and lands while it is still travelling to the right. Because the two clips (Travel_Forward and Aircraft_Rise) are positioned one after the other, the actions occur in a series versus simultaneously.

- 4 Click stop on the Time Slider's playback control.

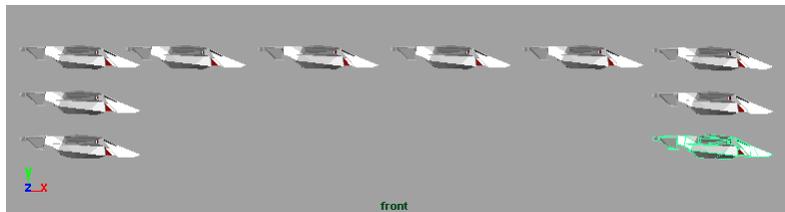
- 5 Click-drag the Aircraft_Lower clip horizontally to the right until the beginning of the clip matches the end of the Travel_Forward clip on the track above it. (Frame In = 299)

If the view of the clips appears limited, select View > Frame All in the Trax Editor to resize the view to encompass the clips.



You also need to modify the Playback End Time because the animation timeline is longer as a result of repositioning the clips.

- 6 In the Range Slider, set the Playback End Time to 360.
- 7 Click play on the playback controls to play back the animation.



The aircraft takes off vertically, then travels to the right, and then lands vertically once the Travel_Forward clip's motion is complete. Because the three clips are positioned end to end in relation to the timeline, the animation occurs as a series of separate events.

- 8 Click stop on the playback controls.

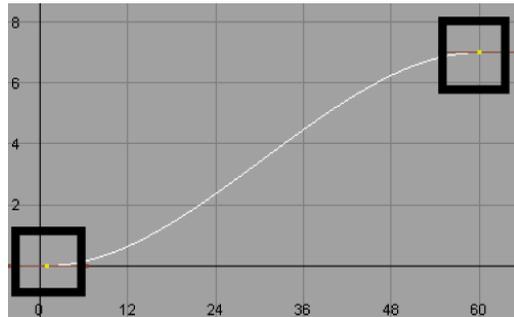
Editing the animation of clips

You can modify the animation of a clip using the Graph Editor in the same way that you would modify any other type of animation. Because you are working with the clip, and not the original animation, you won't change the original source clip animation.

In this section, you use the Graph Editor to modify the animation for each clip so the aircraft's motion accelerates and decelerates at the beginning and end of each clip.

To use the Graph Editor to modify a clip in Trax

- 1 Click on the Aircraft_Rise clip to select it.
The selected clip highlights in yellow.
- 2 From the Trax menu, select View > Graph Anim Curves.
The Graph Editor appears.
- 3 In the Graph Editor, shift-select the two keys on the animation curve as shown below.
- 4 In the Graph Editor, select Tangents > Flat.



The shape of the animation curve updates to a gentle S-shape. Setting the tangents for the animation curve in this manner modifies the clip's animation to cause the aircraft to accelerate when it first rises, and decelerate when it nears the end of its vertical travel.

- 5 Repeat steps 1 through 4 to set the animation curve tangents in a similar manner for the two remaining clips: Travel_Forward and Aircraft_Lower. The shape of the animation curve for Aircraft_Lower will be reversed from the other clip curves because the motion is occurring in the opposite direction.
- 6 Close the Graph Editor window.
- 7 Set the Time Slider's current time indicator to frame 1.
- 8 Click play on the Time Slider's playback control to play back the animation.

The motion accelerates and decelerates at the beginning and end of each clip. As a result, the motion appears more natural. It's important to note that the modified tangents for each clip are independent from the other animation clips and do not affect any other motion.

- 9 Click stop on the Time Slider's playback control.

Reusing clips within Trax

You can save and reuse clips for future use. You can key specific types of motion or attribute effects, save them as clips, and reuse the clips for other characters.

When you reuse a clip created for one character and apply it to another, the clip attributes must be mapped to correspond between the two characters. For more information, see Mapping animation between characters in the Maya Help.

In this lesson, you work with clips created for this aircraft animation. The clips contain animation to make the aircraft shift to the left and right and bank left and right.

To load a clip from disk

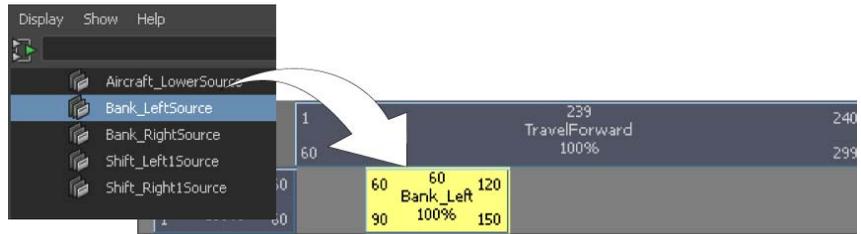
- 1 From the Trax menu, select File > Import Animation Clip.
A file browser appears. When importing a clip, the file browser defaults to the clip directory of your current project.
- 2 Find the clip file named `Bank_Left.mb` in the `GettingStarted/Anim/clips` directory.
- 3 Select the file named `Bank_Left.mb` and click Open.
The clip is imported and appears in the Outliner named as `Bank_LeftSource` to indicate it is a source clip.
- 4 Repeat steps 1 and 2 to import the clips `Bank_Right`, `Shift_Left` and `Shift_Right`.

When you have finished, four new clips appear in the Outliner as source clips. In the steps that follow, you place and position the clips on tracks in the Trax Editor.

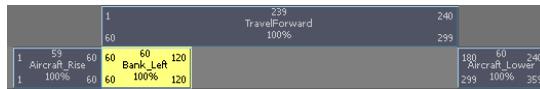
NOTE When you import clips they also appear in, and can be accessed from, the *Visor* (Window > General Editors > Visor). The Visor is a graphical interface that allows you to import items that reside on disk or in your current file (files, textures, clips poses, and so on). Clips that are assigned to a character in Trax will appear listed under the Character Clips tab in the Visor. Clips that have not yet been placed on a track will appear listed in the Visor under the Unused Clips tab.

To place clips into a track from the Outliner

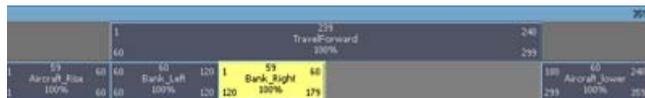
- 1 In the Outliner, using the middle mouse button drag the clip named Bank_LeftSource into the same track as the Rise and Lower clips. Ensure you drag into an open area on the track that the clip will fit into.



- 2 Position the Bank_Left clip on the track so its Frame In number matches the Frame Out number on the Aircraft_Rise clip, as shown below.

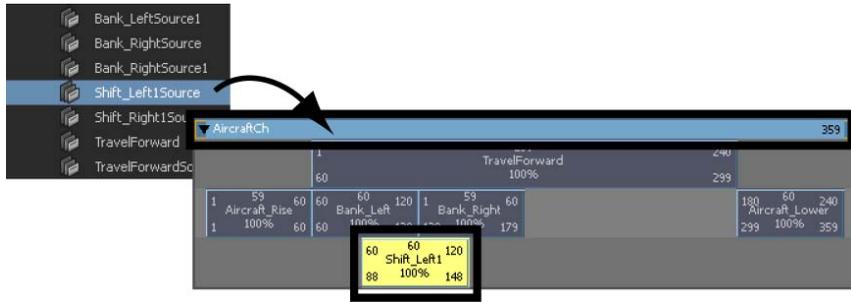


- 3 Using the middle mouse button drag the clip named Bank_RightSource from the Outliner into the same track as the Rise and Lower clips. Ensure you drag into an open area on the track that the clip will fit into.
- 4 Position the Bank_Right clip on the track so its Frame In number matches the Frame Out number on the Bank_Left clip, as shown below.



You can also drag clips from the Outliner or Visor into the Trax Editor and create a new track for them to be placed in.

- 1 Drag the clip named Shift_Left1Source from the Outliner using the middle mouse button, but this time, drag to the narrow blue area above the tracks before releasing your middle mouse button.



A new track appears in the Trax Editor and the clip is placed on the new track.

The narrow blue box is the *summary track* for the character set (in this case the Aircraft) The summary track represents all of the clips under a character, subcharacter, or group. When you import a clip onto the summary area by dragging it from the Outliner or Visor, the Trax Editor creates a new track for the clip.

- 2 Click-drag the Shift_Left clip so it lies directly below the Bank_Left clip on the track above it.



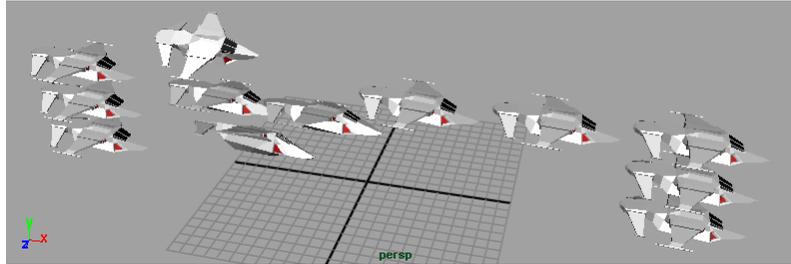
Because the Shift_Left clip is positioned directly below the Bank_Left clip, the actions will occur simultaneously when the animation is played back.

- 3 Using the middle mouse button drag the clip named Shift_Right1Source from the Outliner into the same track as the Shift_Left clip. Ensure you drag into an open area on the track.
- 4 Position the Shift_Right clip on the track so its Frame In number matches the Frame Out number on the Shift_Left clip, as shown below.



- 5 In the front view menu, select Panels > Perspective > persp. The view updates to display the aircraft from a perspective view.

- 6 Dolly and track the view so you can fully see the area the aircraft will travel within.
- 7 Click play on the Time Slider's playback control to play back the animation.



After the aircraft rises, it travels forwards as before but now it shifts sideways to the left while one wing tips downwards, and then shifts sideways to the right, while the opposite wing tips downwards.

- 8 Click stop on the Time Slider's playback control.

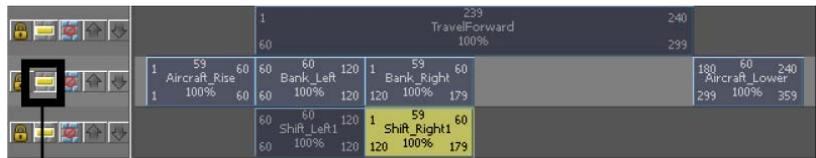
Soloing and muting tracks

Soloing a track allows you to see a track play back in isolation from other tracks. When a track is soloed, the other tracks are disabled and do not play back. Soloing is useful when you want to review the effect a specific track will have on a character.

To solo a track

- 1 In the Trax control area, click the Solo button for the track that contains the Aircraft_Rise and Bank_Left clips.

The other tracks dim to indicate that they will be disabled when the animation is played back.



Solo button

- 2 Click play on the Time Slider's playback control to play back the animation.
The aircraft rises, tilts left and right, and then lowers, without moving forward. Because the other tracks have been shut off temporarily, the aircraft's motion is controlled by only one track.
- 3 Click stop on the Time Slider's playback control.
- 4 Click the Solo button again to enable all the tracks.
The dimmed tracks become active again.

Muting a track disables only the selected track so it does not play back. Muting is useful when you want to review the animation on one or more tracks together without specific tracks being activated. You can mute one or more tracks.

To mute a track

- 1 In the Trax control area, click the Mute button for the track that contains the Shift_Left and Shift_Right clips.
The muted track dims to indicate it is disabled.



Mute button

- 2 Click play on the Time Slider's playback control to play back the animation.
The aircraft rises and tilts left and right as it travels forward, and then lowers. Because the Shift_Left and Shift_Right track is temporarily muted, only the two remaining tracks affect the animation playback.
- 3 Click stop on the Time Slider's playback control.
- 4 Unmute the track by clicking its mute button before proceeding to the next section.

Scaling clips within Trax

You can scale single or multiple clips within the Trax Editor. Scaling a clip either compresses or expands the animation of a clip so it occurs over a shorter or longer period of time. The resulting animation occurs faster or slower as a result.

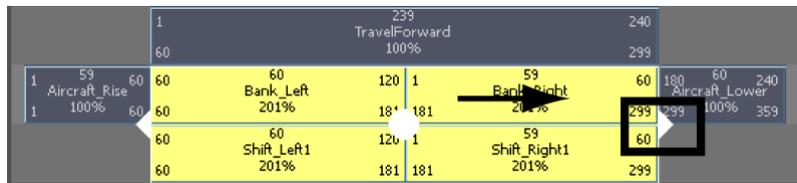
In the next steps, you select the four clips you imported to Trax, and scale them together so they occur over the same time frame as the Travel_Forward clip.

To scale multiple clips in the Trax Editor

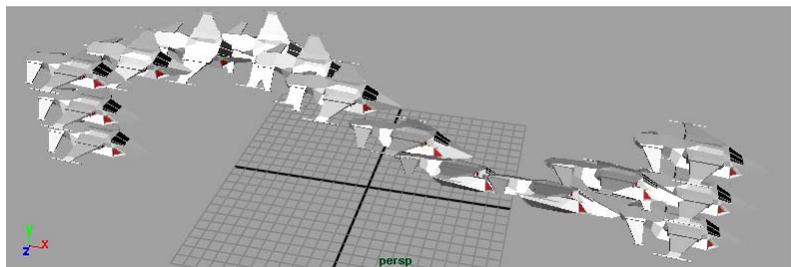
- 1 In the Trax Editor, Shift-click on the clips: Bank_Left, Bank_Right, Shift_Left1, and Shift_Right1.

The *clip manipulator* appears. The clip manipulator appears as a white box that encompasses the four clips with handles (triangles) on either end and a circle handle in the center. The clip manipulator allows you to move or scale multiple clips.

- 2 Using the clip manipulator, drag the right-hand scale handle to the right until the Frame Out numbers on the Bank_Right and Shift_Right clips match the Travel_Forward clip (Frame 299).



- 3 Click play on the Time Slider playback controls to play back the animation.



When the aircraft travels, the sideways and banking motions occur over the entire length of the Travel_Forward clip.

- 4 Click the stop button on the playback controls to stop the playback.

This completes the first half of the Trax lesson. In the second half of this lesson you work with motion capture data in Trax.

Open the second scene for the lesson

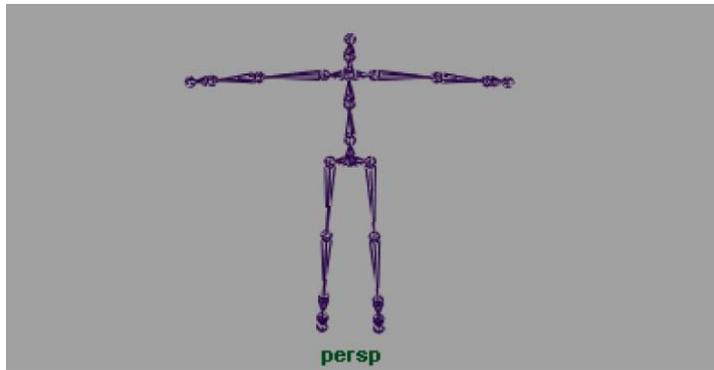
In this second half of this lesson, you work with a second scene file that contains *motion capture data* applied to a skeleton.

Motion capture data is produced by capturing the movement of a real human or animal and digitizing it so it can be applied to a 3D animation character. Motion capture data contains keyframe animation. Motion capture data is useful because it can capture the subtle nuances of motion and gesture that make a character's actions appear life-like. Motion capture data can substantially reduce the work related to posing and keyframing your characters.

- Open the scene file named `Trax_Lesson2.mb`.

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

`GettingStarted/Anim/Trax_Lesson2.mb`.



Motion Capture Data courtesy of:
House of Moves Motion Capture Studios

The skeleton appears in the Maya scene in the standard da Vinci pose. In this pose, the character stands straight with its arms extended out to either side. This pose is useful when you need to skin a character, retarget animation from one skeleton to another, or set a rest pose for the skeleton.

In this half of the lesson you learn how to use the Trax Editor to work with motion capture data by extending and redirecting the motion for a character.

NOTE If you are beginning the lesson from this point, you must first set your panel layouts and display settings for the Trax Editor and Outliner. Refer to [To set a panel layout for use with the Trax Editor](#) on page 243 and [To set the Outliner to display clips](#) on page 243.

If your skeleton doesn't appear as shown, you will need to increase the size of the skeleton's joints.

To resize the joints on the skeleton

- 1 From the main menu, select Display > Animation > Joint Size.
The Joint Display Scale window appears.
- 2 Set the Joint Display Scale to 10.
- 3 Close the Joint Display Scale window.

To preview the motion capture animation

- 1 Click play on the Time Slider playback controls to play back the animation.
The skeleton is repositioned and then walks from left to right, stops, and then sits.
- 2 Press stop on the Time Slider's playback control.

Creating clips from motion capture data

When creating a clip from motion capture, it is a good practice to begin by creating a source clip from the entire motion sequence that excludes the Da Vinci pose. You can create the clip and save it to the Visor or Outliner directly without loading it onto a track. You can later revert back to the source clip and load it into a track if necessary.

To create a source clip of the motion capture data

- 1 In the Time Slider, set the Playback Start Time to 1.

- 2 Press the Go to Start of playback range button to return to frame 1 in the Time Slider.

The skeleton is repositioned at frame 1 where the walking motion begins.

- 3 In the perspective view, select the skeleton by click-dragging a bounding box around the skeleton.

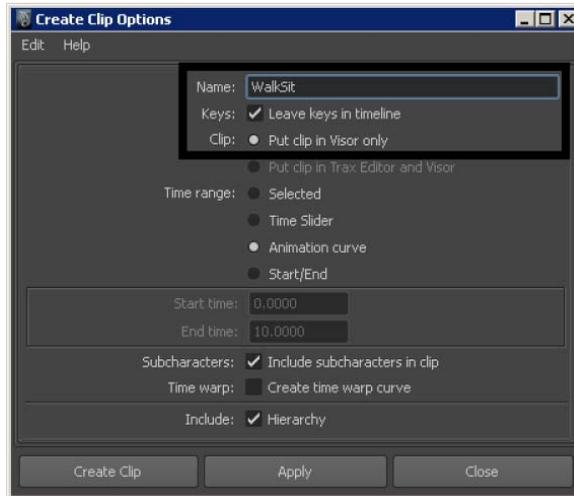
The skeleton is highlighted. When the skeleton is selected, many red ticks appear in the Time Slider. The red ticks represent keyframes from the motion capture data.

- 4 In the Trax Editor menu, select Create > Animation Clip >  .

The Create Clip Options window appears.

- 5 In the Create Clip Options window, click Edit > Reset Settings to set the tool to its default settings, and then set the following:

- Type `walkSit` in the Name field.
- Keys - Leave Keys in Timeline: On
- Put Clip in Visor Only: On
- Click Create Clip.



The source clip appears in the Outliner named as WalkSitSource. (If you use the Visor, the clip appears under the Character Clips tab.)

Extending the length of motion capture data

You often need more animation of a motion sequence than is provided in the motion capture data. For example, you may need to have a walk cycle that is substantially longer than what was provided in the original file. You can use the Trax Editor to edit the motion capture data and create clips that can be *cycled* to make longer animation sequences. When you cycle a clip, the clip is extended in length with duplicate animation from the original clip over a defined interval of time.

If you want to cycle a clip, you must first determine areas within the animation sequence where the clip repeats seamlessly. For example, if you want to create a walk cycle from the motion capture data, you need to determine the time (frame) where the right foot is planted on the ground, and then move forward in the Time Slider, frame by frame, until you find the next frame where the right foot is planted on the ground in exactly the same (or very similar) manner, and then create a clip. The resulting clip will contain two steps of the walking motion. The clip can then be cycled to produce a motion that repeats. How seamlessly it cycles depends on the original motion capture data.

In the steps that follow you create a walk cycle from the motion capture data by:

- Creating a clip from the walking portion of the motion capture sequence that can be repeated.
- Cycle the clip to create a longer walk sequence.

To create a clip that can be cycled

- 1 In the Time Slider, drag the current time indicator to frame 6.
The skeleton updates in the scene to a walking pose where the right foot is planted on the ground.
- 2 Drag the current time indicator forward in the Time Slider until you reach frame 45.
At frame 45, the skeleton is posed in a similar manner as it appeared at frame 6. If you use this range of frames (6 to 45), you can create a clip that can be cycled.
- 3 In the persp view ensure the skeleton is selected.
- 4 In the Trax Editor menu, select Create > Animation Clip > .
The Create Clip Options window appears.

- 5 In the Create Clip Options window, click Edit > Reset Settings to set the tool to its default settings, and then set the following:
 - Type `walkCycle` in the Name field.
 - Put Clip in Trax Editor and Visor: On
 - Time Range: Start/End
 - Start Time: 6
 - End Time: 45
 - Click Create Clip

Two clips appear in the Outliner named WalkCycle and WalkCycleSource.

- 6 In the Trax Editor, with the skeleton selected, click the Load Selected Characters icon.

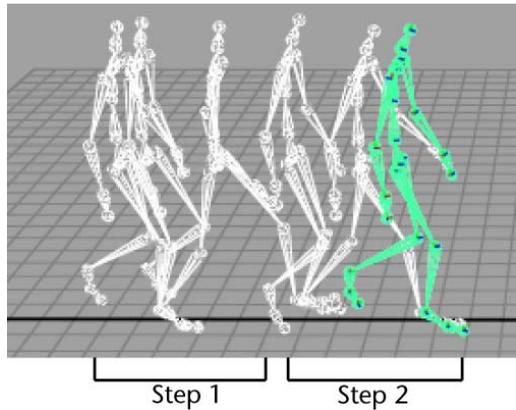


The clip named WalkCycle appears in a new track. Loading the Character displays any tracks associated with the character; in this case the WalkCycle clip.

TIP Click the Frame All  button in the Trax Editor if you don't see the clip.



- 7 In the Time Slider, set the current time indicator to frame 1.
- 8 Click play on the Time Slider playback controls to play back the clip. The skeleton takes two steps and stops.



- 9 Press stop on the playback controls to stop the playback and reset the current time indicator to frame 1.
- 10 In the Trax Editor, drag the clip to the left in its track until the Frame In number reads 1.



This sets the clip to begin at frame 1 in the Time Slider.

In the steps that follow, you cycle the clip to extend the animation so the skeleton repeats the two step motion and travels farther in the scene.

To cycle the motion for the walk clip

- 1 In the Trax Editor, Shift-drag the lower right corner of the Walk clip to the right until a C2.0 number appears in the new region of the clip. When you shift-drag over the Frame Out area, the mouse cursor changes to a circular pointing arrow to indicate you are cycling the clip.



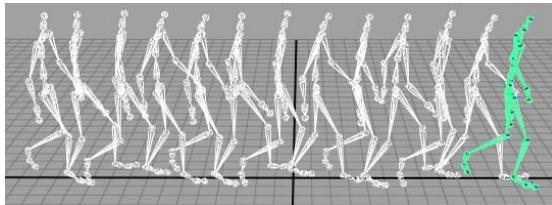
- 2 Double-click on the cycled clip in the C2 region.

The region is highlighted to indicate you can edit the values in the region. The number will likely not be exactly 2.0. This means that the cycle will not repeat exactly twice.

- 3 Type 2.0 to modify the clip so it cycles exactly twice.
- 4 With the clip selected in the Trax Editor, open the Attribute Editor. Open the Channel Offsets menu and ensure that the WalkCycle clip's root.scale and root.translate offset attributes are set to Relative and root.rotate offset attributes are set to Absolute.

Setting the offset attributes for the clip to Relative ensures that each cycle begins at the position where the last sequence left off. This ensures that the skeleton doesn't jump back to its start point after each cycle.

- 5 Click play on the Time Slider playback controls to play back the clip. The skeleton takes six steps and then stops.



When the clip is cycled twice, the two-step walking motion occurs three times in total.

- 6 Press stop on the playback controls to stop the playback and reset the current time indicator to frame 1.

Redirecting the motion within a clip

Motion capture data can be used in a variety of ways. You can redirect the motion of a character using the same motion capture data. The *Redirect tool* allows you to alter the original animation for a character by modifying its path or orientation at any time during its animation. The Redirect tool does this by offsetting the translation and rotation attributes for the selected character.

For example, you can change the start or end point of a character's animation by redirecting the character's starting position or by redirecting its motion during the animation. You can redirect a character in an existing animation

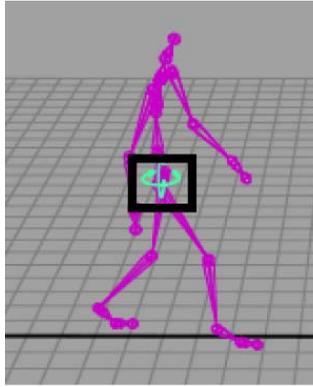
sequence in the scene or a character associated with a clip within the Trax Editor.

In the steps that follow, you redirect the skeleton to turn to the right while walking. The workflow for the redirection is as follows:

- Scrub forward in the animation to determine a location in the clip where you want the character to turn. A realistic choice is where the ball of the skeleton's right foot is planted on the ground plane because this accurately simulates how a real person might turn.
- Select the entire skeleton. In this example, selecting the entire skeleton selects it at the root joint (the top node of the skeleton's hierarchy) Selecting at the root joint ensures that the entire skeleton gets redirected when the redirection control is applied.
- Create and position a rotation redirection control using the Redirect tool.
- Keyframe an initial starting rotation for the redirection control.
- Rotate the redirection control so the skeleton faces the new direction of travel.
- Keyframe the second rotation for the redirection control.

To create a redirection control for the skeleton

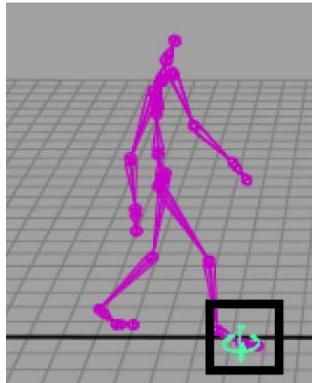
- 1 In the Time Slider, drag the current time indicator to frame 36.
This frame is where the ball of the skeleton's right foot is planted on the ground plane. Use this frame as the location where the skeleton will turn to the right.
- 2 Select the skeleton in the scene view by click-dragging a bounding box around it.
- 3 In the main menu, select Character > Redirect > .
- 4 In the Character Redirection Options Window set the Redirection Type to Rotation only and then click the Redirect button.
A Rotation redirection control appears between the hips of the skeleton, where the root joint for the skeleton's hierarchy is.



You want the skeleton to pivot (rotate) while standing on the ball of its right foot. To do this, you need to move the redirection control to the ball of the right foot of the skeleton.

To move the redirection control to the ball of the foot

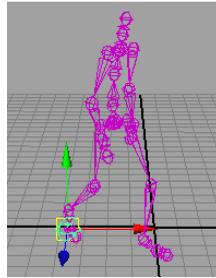
- 1 From the Toolbox, select the Move tool.
The Move tool manipulator appears on the redirect manipulator.
- 2 Press the `v` key and click-drag the redirection control until it is located on the ball of the right foot as shown below.
Pressing the `v` key while using the Move Tool momentarily enables the Snap to Points tool.



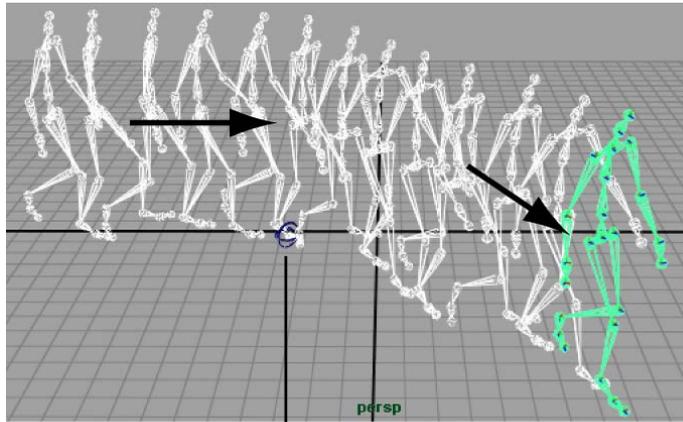
The redirection control is now positioned to redirect the skeleton's walking motion. To complete the redirection process, you need to set keyframes while setting the redirection control so the redirection of the skeleton can occur.

To keyframe the redirection control

- 1 With the redirection control selected, press **s** to set a keyframe.
A red tick appears in the Time Slider at frame 36 indicating the keyframe. Setting a keyframe establishes the starting rotation for the redirection rotation of the skeleton.
- 2 In the Time Slider, drag the current time indicator to frame 57.
The skeleton updates to a position where its left foot is planted on the ground plane. This is the location where you want the skeleton to be rotated to its new direction of travel.
- 3 In the Channel Box, with the redirection control still selected, set the Rotate Y attribute to -45.
- 4 Rotate the redirection control approximately -45 degrees about the Y axis until the skeleton is facing in the direction shown below. (You can use the Channel Box values as a guide.)



- 5 With the redirection control still selected, press **s** to set a keyframe.
Setting a second keyframe establishes when the redirection rotation for the skeleton ends.
- 6 Set the current time indicator to frame 1.
- 7 Click anywhere in the perspective view to deselect any items in the scene.
- 8 Click play on the Time Slider playback controls to play back the clip.
The skeleton takes two steps and then turns to its right after it reaches the redirection control, and continues walking in the new direction of travel and stops when it completes its walk cycle.



Redirect occurs

- 9 Press stop on the playback controls to stop the playback and reset the current time indicator to frame 1.

NOTE When redirecting motion within Trax, you may want to consider creating a clip that contains the redirection animation to make it possible to reposition the redirect animation as a clip.

Now that you've redirected the character to a different direction of travel during the walk cycle, you can reuse some of the motion from the original source clip you created earlier in the lesson and saved to the Outliner/Visor.

You insert the source clip from the clip Library in Trax. The clip Library keeps track of the source clips and poses for the current character set. When the source clip is inserted from the Library, a copy of the source clip is created in a new track.

Once you import the source clip and make a copy of it, you trim the duplicate clip to make the character stop walking and then sit in a different location than the original motion capture.

To insert the source clip

- 1 In the Trax Editor, select Library > Insert Clip. Choose WalkSitSource from the drop-down list.

A duplicate copy of the WalkSitSource clip is created, and inserted onto a new track.

- 2 In the Trax Editor, select View > Frame All.

The track view area updates to display all clips in their entirety.

- 3 In the Trax control area, click the Mute button for the track that contains the WalkCycle clip.

The muted track dims to indicate it is inactive. When the walk cycle track is muted, the character updates to appear in the Da Vinci pose at the origin based on the original source clip.

- 4 Drag the current time indicator to frame 98.

The skeleton updates in the view and the redirection that was applied also occurs. Remember that the redirection occurs for the character, and not the clip.

- 5 Click on the WalkSit clip in the track to select it.

- 6 In the Trax tool bar, click the Trim Clip Before Current Time icon to trim the section of the clip that occurs before frame 98.



The WalkSit clip is trimmed and the portion of the clip before frame 98 is discarded.

- 7 Click the Mute button again for the track containing the WalkCycle clip to reactivate the track.

The dimmed track becomes active again. The skeleton however has updated to an incorrect pose near its previous position. This occurs because the two active clips overlap, and the character is receiving conflicting positional information as a result. (That is, double translation and rotation information.)

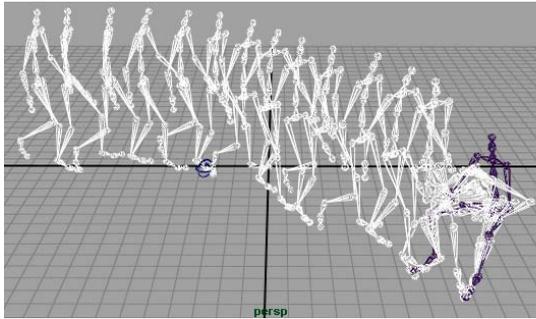
- 8 Drag the WalkSit clip to the right in its track until its Frame In number matches the Frame Out for the WalkCycle clip.

The pose for the skeleton is updated to the correct position.

- 9 Reset the current time indicator to frame 1.

- 10 Click play on the Time Slider playback controls to play back the animation.

The skeleton takes two steps and then turns to its right after it reaches the redirection control, and continues walking in the new direction of travel, stops, and then sits.

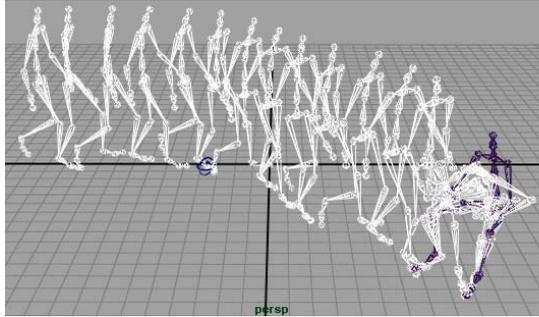


- 11 Press stop on the playback controls to stop the playback and reset the current time indicator to frame 1.

A small jump cut occurs during the transition between the two clips. The jump cut occurs because the poses for the skeleton do not match from the end of the first clip and the beginning of the second. In your own work, try to anticipate how motion will blend between clips when working in Trax. You can correct the jump cut in this particular example by extending the WalkCycle clip.

To extend the WalkCycle clip to correct the jump cut

- 1 In the Trax Editor, double-click on the WalkCycle clip in the C2 region. The number becomes highlighted indicating you can edit the number.
- 2 Type 2.2 to edit the clip so it cycles 2.2 times. The new Frame Out number for the WalkCycle clip becomes 126. It now overlaps in time with the WalkSit clip.
- 3 Drag the WalkSit clip to the right until its Frame In number becomes 126.
- 4 Click play on the Time Slider's playback control to play back the animation. The jump cut is less noticeable than before.



- 5 Press stop on the Time Slider's playback control.
You have completed this lesson. Before leaving this lesson, reset the joint size back to its default setting.

To reset the joint size

- 1 From the main menu, select Display > Animation.
The Joint Display Scale window appears.
- 2 Set the Joint Display Scale to 1.
- 3 Close the Joint Display Scale window.

Beyond the lesson

In this lesson, you learned some fundamental techniques for using the nonlinear animation tools in Trax. You learned how to:

- Key animation sequences for use as nonlinear clips in Trax.
Clips can contain minimal or many animated attributes. You can create clips from existing animation or key animation specifically for use in clips. You should consider what attributes you need represented in a clip when you key the animation before you decide to use nonlinear animation in your work. In this way, a clip can be used to control a specific attribute.
- Create clips from your animation sequences for use in the Trax Editor.
When you create a clip, a character set is created if one does not already exist. You can load and work with multiple character sets in the Trax Editor. To select different character sets when working in Trax you can use the Select Current Character drop-down menu. Clips can also be arranged,

scaled, and moved as groups. For more information see Move clips and Track view area in the Maya Help.

- Use the Trax editor to position, re-arrange, cycle, and scale clips in order to create and modify animation sequences in the scene.
If a character animates or scales in a manner you had not expected when working in Trax, you may want to check to see if the clips overlap or are arranged in a manner that doesn't provide the expected results. In addition, a clip containing motion that is intended to be appended to another clip's motion must have its Offset attribute set to Relative, and Absolute Rotations set to On (in most instances). There are many other tasks you can perform with clips in the Trax Editor that are beyond the scope of this lesson. For more information see Nonlinear animation components in Trax in the Maya Help.
- Use the Trax Editor to work with motion capture data.
The Trax Editor is useful when working with motion capture data. Since motion capture data usually contains a lot of keyframe information, the Trax Editor can be useful for creating, modify, and extending the motion data provided. Cycling and redirecting motion are two methods that can be used to get more from your motion capture data. For more information see the section on Motion Capture Animation in the Maya Help.
- Redirect the motion for a character using the Redirect tool.
You can use redirection controls to redirect any type of motion in Maya (Keyframe, motion capture, expressions, dynamics, and so on). For more information and additional examples on redirection see Redirect the motion of an animated character in the Maya Help.
- Trim clips in the Trax Editor.
You can trim a clip either before or after the current time indicator discarding the remainder of the clip in the process. You can also split a clip into two parts. The original source clip is not affected. For more information see Split clips in the Maya Help.

Blending clips

You can blend the animation between clips. Blends allow you to create smooth transitions and mixes between different motions. A blend can be applied between any two clips that overlap entirely, partially, or not at all. The best results are obtained when blending between similar motions. In some situations, a blend can be used to correct jump cuts. For more information see Create and edit blends in the Maya Help.

Key into clip

You can set keyframes to alter the animation within a clip. This process is called Keying into a clip. When you key into a clip, keyframes are placed at the current time on all the animation curves in the selected clip. For more information see Key into a clip in the Maya Help.

Time Warps

Time Warps let you change the timing of a clip without modifying the clip's animation curves. You modify the Time Warp by editing the animation curve that controls the warp. Time Warps can also be used to reverse the animation in a clip. For more information see Create and edit time warps in the Maya Help.

Visor and Outliner

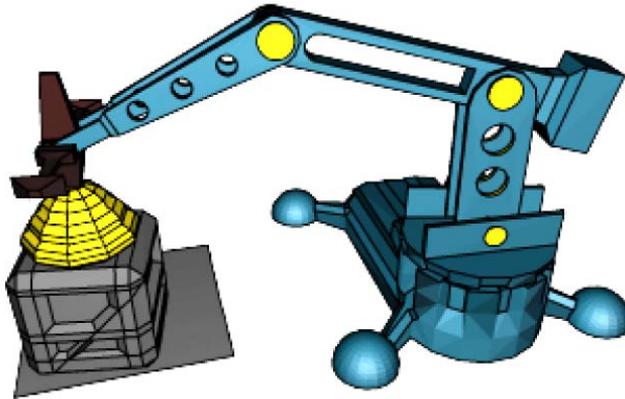
Both the Visor and Outliner can be used to access clips. The Visor provides a graphical interface to access other resources (clips, poses, shaders, textures, brushes and so on) in your scene. The Outliner displays clips in a textual manner, and is a bit more compact.

Audio clips

You can display and play multiple audio files in Trax allowing you to synchronize your motion clips to specific audio events (sound effects, musical notes, drum beats and so on) in the audio file. Once you import an audio file into Trax, you can then move and rename the audio clip. For more information see Work with audio in Trax in the Maya Help.

Lesson 5: Inverse kinematics

Introduction



Model created by Rob Ormond

Inverse Kinematics (IK) allows you to efficiently pose your models and characters for animation. You pose the skeleton by positioning IK handles, usually located at the end of an IK joint chain (for example at the hand). The rotations for the other joints are calculated automatically by an *IK solver*. An IK solver calculates the rotations of each joint back up the hierarchy of the joint chain based on the position of the IK handle. See [Creating a skeleton hierarchy](#) on page 281 for more information on skeletons and joints.

By contrast, Forward Kinematics (FK) requires you to rotate each joint in turn until you have the desired pose for a keyframe. This can be a tedious process when posing a complex skeleton. When using Inverse Kinematics, it's faster to pose a character because the joint rotations further up the hierarchy are calculated automatically based on the placement of the IK Handle. Using both FK and IK techniques on the same skeleton is called FK/IK blending.

IK is obviously useful for posing animated characters that have a skeleton (two-legged, four-legged and so on) and has others applications, such as posing skeletons within mechanical-like models.

The goal of this lesson is to animate a mechanical arm so that it extends into a position over a cargo box, picks up and moves the cargo box to another location, and then sets it down.

As you work through the lesson, you'll learn about many of the basic features related to skeleton hierarchies, IK, and parent constraints. Some initial setup of the model and IK is required so that it can be posed and keyframed to provide a convincing animation.

In this lesson you learn how to:

- Create and use hierarchies to define logical relationships between animatable objects.
- Combine the components of a 3D mechanical arm model into a hierarchical grouping.
- Construct a skeleton for use with IK.
- Combine the skeleton with the mechanical arm model by parenting the components into a hierarchy.
- Create an IK system to allow the skeleton hierarchy to be posed.
- Create a control object to manipulate the IK system.
- Use constraints to control the position and orientation of components within the IK system.
- Limit the range of movement on the IK system.
- Pose and set keyframes for the IK system.
- Animate an object by constraining its motion to two or more objects.

Open the scene for the lesson

In this lesson, you work with a scene we created for your use. In the first steps of this lesson, you open the scene and view the various elements that have been created for your use.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 203.

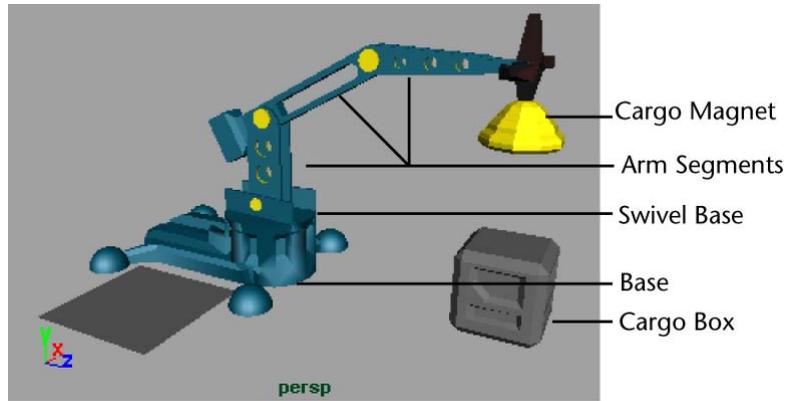
- 2 Open the scene file named `IK_MechArm.mb`.

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

```
GettingStarted/Anim/IK_MechArm.mb
```

The scene contains a model of a mechanical arm and a cargo box to be moved by the arm. The components of the mechanical arm include:

- A cargo magnet cup on the end
- Three arm segments
- A base for the arm to swivel about
- A four-legged base for the assembly

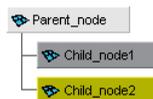


To work with this model using IK you must create a hierarchy for it.

Understanding hierarchies

Inverse Kinematics relies on hierarchical relationships between the components of a model and the IK system to pose and animate the components. Before using Inverse Kinematics it is important to understand hierarchies in Maya.

A hierarchy is composed of a series of nodes that are combined for some purposeful relationship. Individual nodes may represent the surfaces of a model, a network of textures that describe a shading material, or series of joints in a skeleton. Hierarchies are structured in a top-down manner, with one node at the top (the *parent node* or *root node*) and other nodes (*child nodes* or *leaf nodes*) attached and interconnected beneath the top node.



Selecting the parent node at the top of the hierarchy also selects the items contained in the hierarchy below. Selecting a child node lower in the hierarchy selects any child nodes that are lower in the hierarchy.

Hierarchies allow you to create complex structures with relationships between components. For example, when you animate a hierarchical model, you simply need to select and move the parent node of the hierarchy and the rest of the model (child nodes) also moves.

Regardless of whether you decide to animate the mechanical arm using FK or IK, or a combination of both methods on the same model, the best approach involves making the mechanical arm model into a hierarchical structure.

Creating a hierarchy for the mechanical arm ensures that:

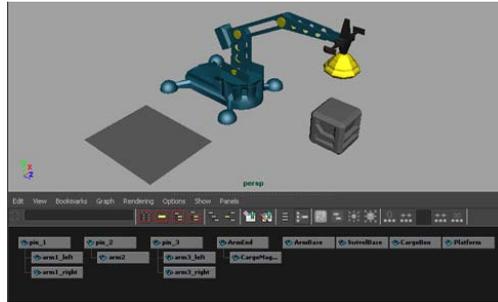
- Components of the hierarchy can be rotated as if they were one unit. A common pivot point is created at the parent node level. This allows the separate components to rotate as one unit based on their membership in the hierarchy. For example, if the mechanical arm's base (parent node) is rotated, anything lower down the hierarchy, for example, the mechanical arm which is constructed of child nodes, will also rotate because of its membership and position in the hierarchy.
- When a particular segment of the model is posed and keyframed, the components that are lower in the hierarchy are also posed and keyframed.

Viewing hierarchies using the Hypergraph

In the following steps, you use the *Hypergraph* to view the hierarchy for the mechanical arm. The Hypergraph is a scene management editor that presents a graphical view of the scene hierarchy. It allows you to more easily create, understand, and manage the hierarchy of objects and nodes in your scene.

To view hierarchies using the Hypergraph

- 1 From the perspective view's panel menu, select Panels > Layouts > Two Panes Stacked.
The workspace displays two panels stacked on top of each other with an identical perspective view of the scene.
- 2 In the lower panel, select Panels > Hypergraph Panel > Hypergraph.
The lower panel displays the Hypergraph.

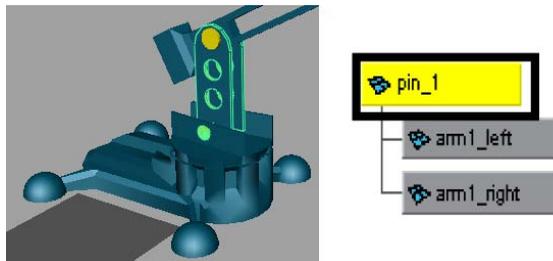


- 3 In the Hypergraph menu, select Graph > Scene Hierarchy.
If the menu item appears dimmed, it indicates that the Hypergraph is already displaying the scene hierarchy
The Hypergraph displays with a black background by default. In order to see the relationships between the items, the background has been modified for the images in this lesson.
- 4 In the Hypergraph menu, select View > Frame All to see all the components of your scene.

The boxes represent the *nodes* for the various items in the scene. Some items in the scene have already been parented into a hierarchy. This is indicated by the lines connecting some of the nodes. Selecting the nodes in the Hypergraph also selects the item in the scene view.

To select an item using the Hypergraph

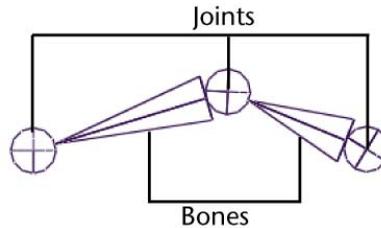
- In the Hypergraph, click each of the separate items in the hierarchy at the parent level to select them. At the same time, observe which objects become highlighted on the model in the perspective view.



Hierarchies combine the various objects into discrete components for the mechanical arm, that is, the base, lower arm, middle arm, upper arm, and so

on. If you select and rotate any of the parent nodes in the hierarchies, none of the other arm components follow because they are not combined into one unified hierarchy.

Creating a skeleton hierarchy



To link the various components of the mechanical arm into a hierarchy that can be posed, you need to create a *skeleton*. A skeleton is a hierarchy of *joints* that are connected together with *bones*.

In the scene, a joint represents a special type of node that gets created in a skeleton hierarchy. A joint acts as the parent node for any other joints that occur in the hierarchy below it. Each joint has a rotational pivot point associated with it. A bone is the visual representation used in the scene view to connect the joints and help visualize the joint chain.

A skeleton is similar to a skeleton in the real world in that it acts as the underlying structure for the surfaces to be attached. While you can view the skeleton with its bones and joints in the scene view, it does not appear in your rendered images. Its purpose is to assist you in setting up and posing your models and characters and to visualize the motion you want to achieve.



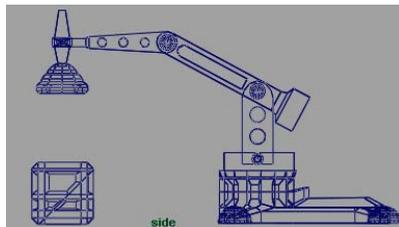


Skeletons are integral to the animation of any character or hierarchical model. Examples of characters that use skeletons are two and four legged animals. Skeletons are also useful for animating other organic components such as the tail of an animal, a tentacle of an octopus, or a snake.

In the following steps, you create a simple skeleton consisting of four joints that you combine with the mechanical arm model into a skeleton hierarchy, allowing you to pose and animate the model.

To create a skeleton for the mechanical arm

- 1 In the perspective view menu, select Panels > Orthographic > side.
The view updates to display the side view.
You need to view the components of the mechanical arm fully in the side view so you can draw the joints for the skeleton.
- 2 Resize the side view to a larger size by dragging the border between the side and Hypergraph panel views downwards.
- 3 In the side view menu, select Shading > Wireframe (Hotkey 4).

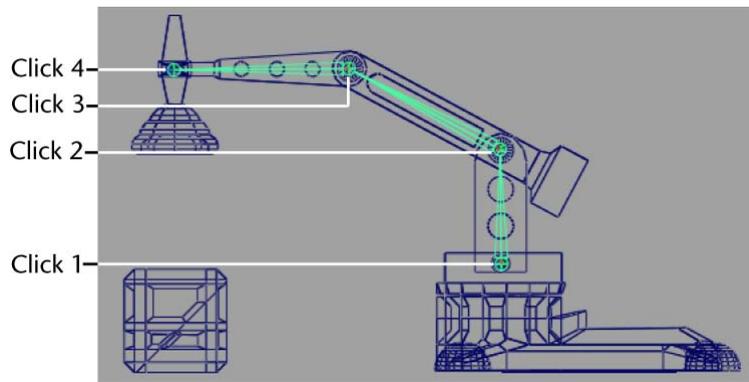


- 4 From the main menu, select Skeleton > Joint Tool >  .
The Tool Settings window for the Joint Tool appears. The Joint Tool is used to create the joints and bones for a skeleton.
- 5 In the Joint Tool settings window, do the following:
 - Click Reset Tool to set the tool to its default settings.

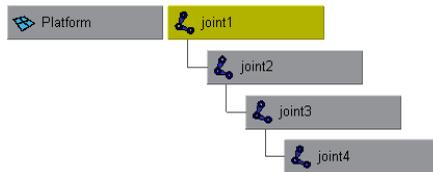
- 6 In the side view, starting from the base of the mechanical arm, do the following:
 - Click in the center of each pivot pin on the mechanical arm to place four joints as shown in the image below.

Try to click as close to the center of each pivot pin as you can, as the rotation of the joints on the model will be based on the location of the joints on the skeleton. As you place the joints, a bone appears, connecting each joint. (Because the model lies along the YZ plane, the joints are created close to the center of each corresponding joint on the mechanical arm model.)

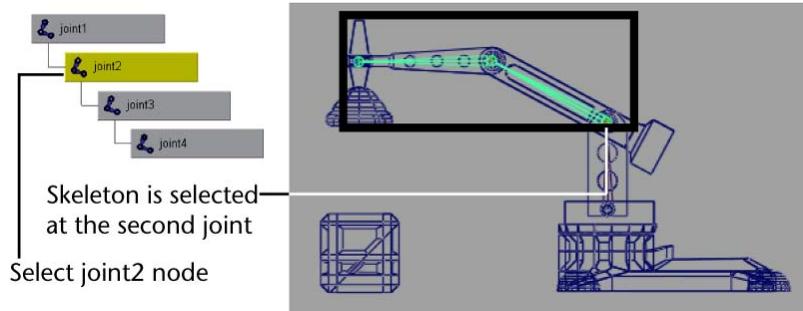
- After you place the fourth joint, press Enter to indicate that the last joint has been placed.



- 7 In the Hypergraph, dolly the view to see the skeleton node hierarchy you just created. (It appears at the right side of the view.)



- 8 In the Hypergraph, select the joint1 node.
The entire skeleton becomes selected. Any rotations on this node also affect anything lower down the hierarchy.
- 9 Select joint2 (You can also press the down arrow on your keyboard to select further down the hierarchy).



The skeleton is selected from joint2 downwards in the hierarchy. Any rotations that you make on this node affect only joint2 and nodes below joint2.

Once a skeletal hierarchy is created, the various surface components can be added to the hierarchy.

Parenting a model into a skeleton hierarchy

Parenting is the term used when placing a node beneath another in a hierarchy so it becomes the child of the node above it (its parent) in the hierarchy.

In the following steps, you add the surfaces of the mechanical arm model into the skeleton hierarchy. This process links the models' surfaces to the skeleton hierarchy making it possible to pose and animate the model like a real mechanical arm. Using the Hypergraph to select the nodes allows you to better visualize the process and the hierarchical relationships that result.

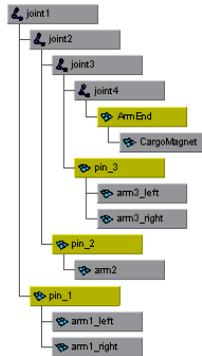
NOTE The following parenting procedure works well for the mechanical arm (no organic bending or deformation occurs) and provides one technique for adding modeled objects to a skeletal hierarchy.

A more common technique for combining a character model into a skeleton hierarchy is referred to as *skinning* or *binding*. The Bind Skin tool combines the surface components and joints together into a hierarchy and also takes into account any deformations that may occur when a character bends or flexes (knees, waist, neck, elbows and so on). Maya provides a few methods for skinning. For more information see *Skeletons and skinning* in the Maya Help. For a tutorial on skinning see the *Smooth Skinning* lesson in the Character Setup chapter in this guide.

To parent the mechanical arm into the skeleton

- 1 In the Hypergraph menu, select View > Frame All.
The Hypergraph displays all the nodes for the scene.
- 2 In the Hypergraph, select the node named ArmEnd and then Shift-select the node named joint4.
The order of selection is important. You want to select the child node first, and then Shift-select the item that will become the parent node second.
- 3 In the main menu, select Edit > Parent (Hotkey p).
In the Hypergraph, joint4 is now the parent node for the ArmEnd node in the skeleton hierarchy.
- 4 Repeat the parenting process for the following items:
 - pin3 is parented under joint3
 - pin2 is parented under joint2
 - pin1 is parented under joint1

Your new hierarchy should appear as shown in the diagram below.

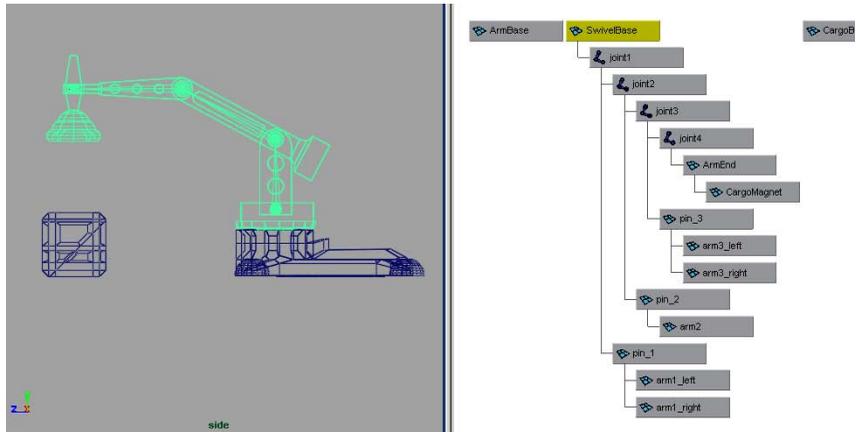


In the next steps, SwivelBase becomes the parent node for the new hierarchy so that the entire arm assembly swivels about the center of the base component.

- 5 In the Hypergraph, select the node named joint1 and then Shift-select the node named SwivelBase.
- 6 In the main menu, select Edit > Parent (Hotkey p).

Change the panel display to a layout that allows you to view the hierarchy in the Hypergraph.

- 7 In the side view menu, select Panels > Layouts > Two panes side by side. The panels display in a side by side configuration.
- 8 Dolly the Hypergraph view as required to view the new hierarchy.



- 9 In the Hypergraph, select the node named Swivel_base.
In the side view, observe that both the mechanical arm model and the skeleton are selected. Swivel_base is the parent or root node for the hierarchy. If you rotate SwivelBase, everything else below that node (the entire arm with skeleton) rotates.
- 10 Select the node named joint3.
Observe that the last two components of the mechanical arm are selected. If you rotate joint3, only the components below that node in the hierarchy will be affected.
- 11 Select the node named pin2.
Observe that only one isolated segment of the model is selected. No other components or joints are selected when pin2 node is selected. If you rotate the model at pin2, only the component at pin2 will rotate in isolation from the rest of the hierarchy because it is located lower in the hierarchy.

It is now possible to select and rotate the skeleton joints to pose the mechanical arm as a unit, and then set keyframes to animate it. Posing a character using joint rotations is referred to as Forward Kinematics (FK). FK allows for precise

control when setting up a pose, but it can be a time consuming process to select and rotate each joint into the desired position.

When an animation is more focused on the goal of placing the end joint in a hierarchy in a particular position, (for example, a hand or a foot), it is more efficient to use Inverse Kinematics to pose the skeleton. For more information on forward and inverse kinematics, see *Posing skeletons in the Maya Help*.

NOTE It's not a simple matter to state that one technique or another is the only approach to achieving a desired result when animating in Maya. You must decide whether a particular technique works for a particular scenario based on the options it provides. A combination of animation techniques can be creatively used in a variety of ways.

Applying IK to a skeleton hierarchy

IK allows you to control the joint rotations in a skeleton using an *IK handle*. You pose the skeleton by positioning the IK handle (usually located at the end extremity of the skeleton) and having the rotations for the other joints calculated automatically by the *IK solver*. In the steps that follow, you apply IK to the skeletal hierarchy.

To create an IK handle for the mechanical arm

- 1 From the main menu, select **Skeleton > IK Handle Tool > **.

The IK Handle Tool Settings window for the Joint Tool appears. When the mouse pointer is in the orthographic view its appearance changes to the tool's cross hair cursor.

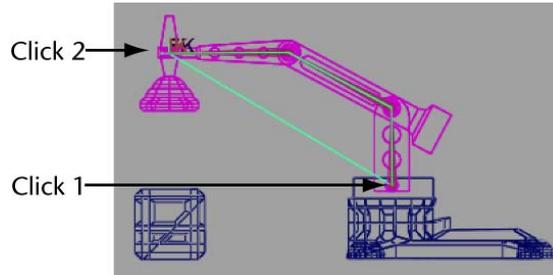


- 2 In the IK Handle Tool Settings window, do the following:
 - Click **Reset Tool** to set the tool to its default settings.
 - Set the **IK Handle Settings: Current Solver** - ikSCsolver.

The ikSCsolver setting selects the IK Single Chain Solver. This solver ensures that the joints in the skeleton will lie along a single plane. This solver is well suited for the mechanical arm, because the arm doesn't need to twist, unlike a real arm.

3 Working in the side view, starting from the base of the mechanical arm, do the following:

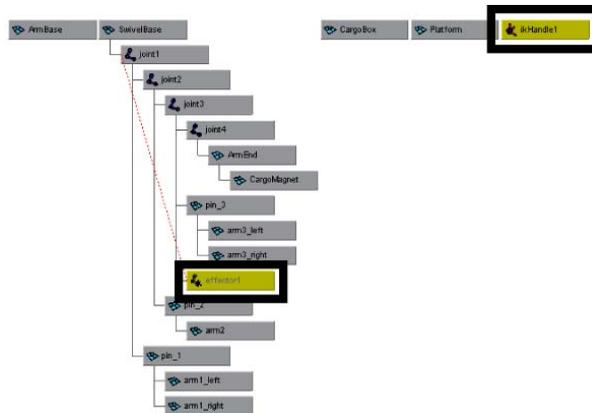
- Click on the skeleton joint at the base of the mechanical arm.
- Click on the skeleton joint at the tip of the mechanical arm. (The IK Tool is designed to attach to the joints closest to your selection point.)



In the side view, the IK handle is drawn as a line from the start and end joints of the IK chain.

4 In the Hypergraph menu, select View > Frame All.

In the Hypergraph, two new nodes appear in the hierarchy that represent the IK chain: an IK Handle node and an end effector node.



5 In the Hypergraph, select the node named ikHandle1.

The IK Handle (in combination with the end effector) is used by the IK solver to calculate the rotation of the joints in the IK chain. By default, the IK Handle is located at the last joint of the IK chain in the scene view.

- 6 Select the Move Tool from the Toolbox, and drag the Move Tool's Z axis manipulator (blue arrow) to reposition the mechanical arm.
When you drag the manipulator, the skeleton and mechanical arm model are repositioned.
- 7 Reset the mechanical arm to its previous position by selecting Edit > Undo, Redo, Repeat until the mechanical arm is positioned into its previous orientation.

NOTE When using the Undo feature, do not undo so many times you inadvertently undo the IK handle you've just created.

The next steps describe how to manipulate the IK Handle using a control object.

Creating a control object for an IK system

Using the IK Handle to pose an IK chain is generally not a good practice. The IK Handle can be challenging to select in the scene view, especially when it's grouped into a hierarchy that includes a skeleton and other surface components. A better practice is to create a *control object*.

You create a control object to select and manipulate an IK system instead of directly using the IK Handle. You can create a control object with a curve or a locator, which are often used because they don't appear in the rendered image.

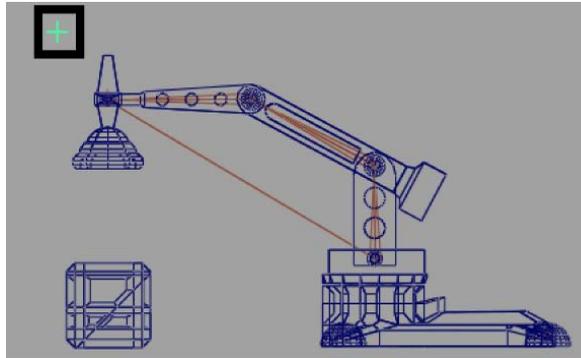
The control object controls the movement of the IK Handle using a *constraint*. You can constrain the position, orientation, or scale of an object to other objects using constraints.

To create a control object:

- Create and position the control object in the scene view.
- Name the node for the control object in the Hypergraph.
- Freeze transformations for the control object.
- Label the control object in the scene view.
- Constrain the IK Handle to the control object.

To create a control object using a locator

- 1 From the main menu, select Create > Locator.
A three-dimensional cross is created at the origin in the scene view.
- 2 Select the Move Tool from the Toolbox.
- 3 In the side view, reposition the locator slightly above and in front of the end of the mechanical arm as shown in the image below.



To easily identify the locator as a control object in the Hypergraph, rename the locator.

To rename the control object using the Hypergraph

- 1 In the Hypergraph, right-click the node named locator1. Choose Rename from the pop-up menu that appears.
The locator1 name is highlighted on the node.
- 2 Type the name ArmControl and press Enter to save the name for the locator.



Before you constrain the IK handle to the control object, you need to freeze the transformations for the control object. *Freeze Transformations* zeros the transformations for an object without changing the position of the object. This is another good practice; If it becomes necessary to reset the arm to its default position, you can set the transformation on the control object to zero, and the arm will return to this default pose.

To freeze the transformations for the control object

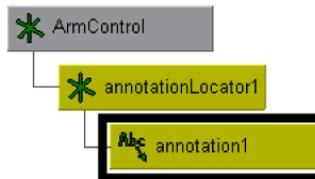
- 1 In the Hypergraph, select the node named `ArmControl`.
The control object is selected in the scene view.
- 2 In the main menu, select `Modify > Freeze Transformations > `.
- 3 In the Freeze Transformation Options window do the following:
 - Select `Edit > Reset Settings` to set the tool to its default settings.
 - Click `Freeze Transform`.

The transformations for `ArmControl` are set to zero. From this point onwards, you can reset its position to this location by zeroing its translations in the Channel Box.

To easily identify the locator as the control object in the scene view, label the control object using an *annotation*. An annotation is a text label that can be set to point at the labeled item. Annotations are useful because they always face the viewer, regardless of the orientation of the model in the scene view.

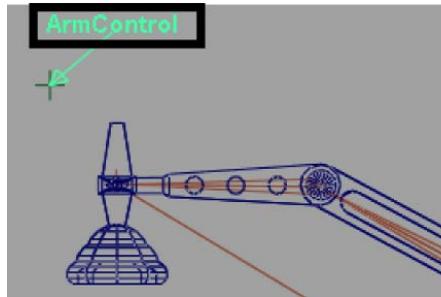
To label the control object in the scene view

- 1 In the Hypergraph, select the `ArmControl` node.
- 2 In the main menu, select `Create > Annotation`.
A input window appears.
- 3 Type `ArmControl` and click OK.
In the scene view, the word *ArmControl* appears near the locator/control object. In the Hypergraph, two new nodes appear in a hierarchy beneath `ArmControl`.



- 4 With the annotation still selected, select the Move Tool.
- 5 In the side view, drag only the annotation so it appears slightly offset from the locator object as shown below.

The annotation displays in any scene view so you can easily identify the control object for selection.



To simplify the display of the Arm Control node

- In the Hypergraph, right-click the ArmControl node. Choose Collapse from the pop-up menu. (You can also double-click a node to collapse or expand the hierarchy below it.)

The ArmControl node appears as a single node instead of the previous three node hierarchy. The small arrowhead on the lower left-hand corner of the node indicates that the node is collapsed.



The control object must be linked to the IK Handle to control the IK Handle.

Constraining an IK system

You can constrain the IK Handle to the control object (ArmControl) using a *point constraint*. A point constraint allows the transformation attributes of one object to be controlled by the transformations of another object. For example, you select and position ArmControl without having to touch the IK Handle. You also set the constraint so ArmControl maintains the current distance (*offset*) from the IK Handle. The offset ensures that the constrained IK handle doesn't move to the same position as the control object.

To constrain the IK Handle to the control object

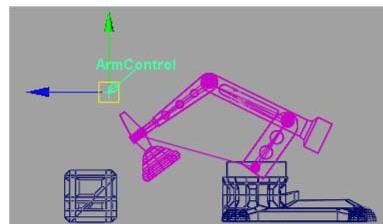
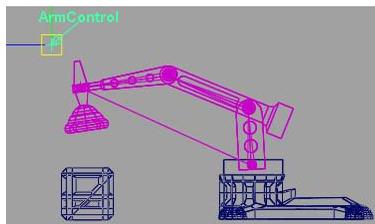
- 1 In the Hypergraph, select the ArmControl node and then Shift-select the ikHandle1 node.

The order of selection when applying a constraint is important: you must select the constraining object first, and the item to be constrained second.

- 2 In the main menu, select **Constrain > Point > **.
The Point Constraint Options window appears.
- 3 In the Point Constraint Options window, do the following:
 - Select **Edit > Reset Settings** to set the tool to its default settings.
 - Turn on the **Maintain Offset** setting.
 - Click the **Add** button.

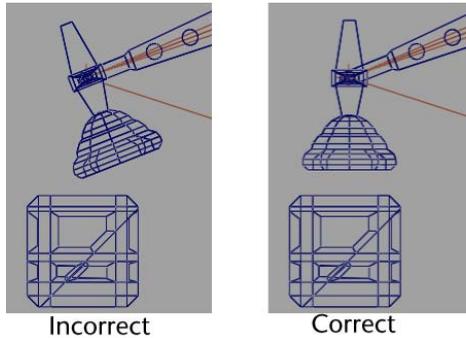
The IK handle is constrained to the control object. In the Hypergraph, a constraint node is created beneath the ikHandle1 node.

- 4 In the Hypergraph, select only the ArmControl node.
- 5 In the Toolbox, select the Move Tool.
- 6 In the side view, drag the manipulator to reposition the control object. (Experiment with a few different positions in the side view only.)
When you drag the manipulator, the control object moves, which in turn moves the IKHandle. Because the IK handle controls the skeleton and the mechanical arm model, they also move. The offset between the control object and the IK Handle is maintained.



The Cargo Magnet does not continuously point downwards when rotated. Its orientation is based on the rotation it had in its original position.

The mechanical arm will be easier to pose if you set Cargo Magnet to be oriented downwards towards the floor of the scene regardless of the orientation of the other components of the arm.



You make CargoMagnet point downwards using an *orient constraint*. An orient constraint constrains the orientation (rotation) of the X, Y, and Z axes of one object to match those of the constraining (target) object.

In the next steps, you reset the position of the mechanical arm to its default position, and then apply an orient constraint to the last joint in the IK system so that its orientation matches the control object. The effect is that CargoMagnet points downwards regardless of any other movement in the scene.

To set the IK system to its default position

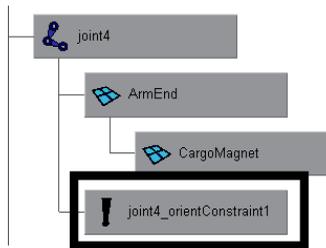
- 1 In the Hypergraph, select the ArmControl node.
- 2 Open the Channel Box by clicking the Show/Hide Channel Box icon  .
- 3 In the Channel Box, set the X, Y, and Z transformations for ArmControl to 0.
Zeroing the transformations on ArmControl resets the mechanical arm to its home position because of the freeze transformations step you performed in the previous section.
- 4 Close the Channel Box by clicking the Show/Hide Channel Box icon.

To apply an orient constraint for the IK system

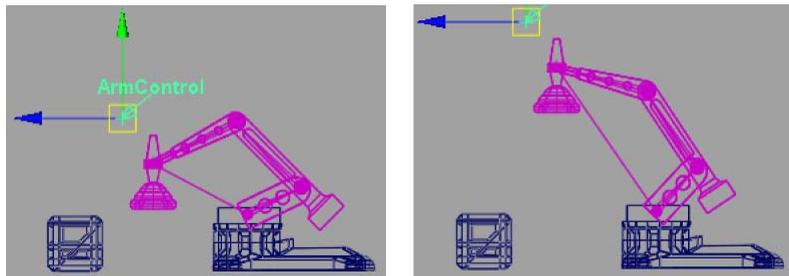
- 1 In the Hypergraph menu, select View > Frame All.
The Hypergraph displays all the nodes in the scene.

- 2 In the Hypergraph, select ArmControl and then Shift-select the node named joint4. (You may need to dolly and track the Hypergraph view to view these nodes.)
- 3 In the Main Menu bar, select Constrain > Orient >  .
- 4 In the Orient Constraint Options window, do the following:
 - Select Edit > Reset Settings to set the tool to the default settings.
 - Click the Add button.

Joint4 is constrained to have the same orientation as the control object. In the Hypergraph, the constraint node is created below the joint4 node in the hierarchy.



- 5 With only the ArmControl node selected, use the Move Tool to reposition ArmControl in the side view.



CargoMagnet maintains an orientation facing the floor of the scene regardless of how the mechanical arm is positioned in the side view.

In the next section, you learn how to limit the range of movement for the IK system.

Limiting the range of motion of an IK system

It's possible to move ArmControl so that the mechanical arm fully extends to a straight position or orients itself to other positions that you don't necessarily want.

You can limit the range of motion of the arm to ensure that the IK system poses in a predictable manner. In the following section, you limit the motion for the IK system as follows:

- Lock the translation of the control object (ArmControl) so that it can only move in Y and Z. By locking the X channel so it cannot be selected or modified, the arm will not be able to move from side to side.
- Lock the rotations on the swiveling base for the arm so it only rotates about its Y axis, and not about its X or Z axes.
- Limit the translations for the Arm Control so the mechanical arm cannot fully extend to a straightened position.

To lock the Translate X channel for ArmControl

- 1 In the Hypergraph, select the ArmControl node.
- 2 Open the Channel Box to view the transform channels for ArmControl.
- 3 Set all Translate or Rotate channels to 0, if they are not already.
- 4 In the Channel Box, select the Translate X channel by clicking on its name.
- 5 Right click on the word Translate X.
A drop-down menu appears.
- 6 From the drop-down menu, choose Lock Selected.
The numerical field for Translate X is dimmed indicating that the channel is locked. If you try to move ArmControl in X, it will not be possible unless you unlock the Translate X channel.

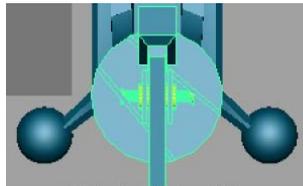
To lock the Rotate X and Z channels for SwivelBase

- 1 In the Hypergraph, select only the SwivelBase node.
- 2 Open the Channel Box to view the rotation channels for SwivelBase.
- 3 Set all Translate or Rotate channels to 0, if they are not already.

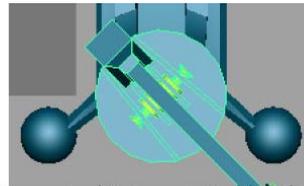
- 4 In the Channel Box, select the Rotate X and Rotate Z channels by Ctrl-clicking on their names.
- 5 Right-click on either of the selected names and select Lock Selected from the drop-down menu that appears.

To confirm that the Rotate X & Z channels are locked

- 1 In the Hypergraph, ensure only the SwivelBase node is selected.
- 2 In the view, select Panels > Orthographic > top.
- 3 In the view, select Shading > Smooth Shade All.
- 4 Dolly the top view so you can fully view the mechanical arm as shown below.
- 5 Using the Channel Box, rotate SwivelBase by typing 45 in the Rotate Y channel and pressing Enter.



Rotated like this



Expected it to rotate like this

SwivelBase rotates in the view but the mechanical arm does not rotate. You first might think something is wrong because SwivelBase and all of the arm's skeletons and surfaces are parented into the same hierarchy.

NOTE You may sometimes discover that something doesn't work in the manner you originally anticipated when setting up an IK system. When this occurs, it's useful to stop and diagnose why something is working the way it is (or not working the way it should). In that way you can retrace your actions to determine where the problem resides.

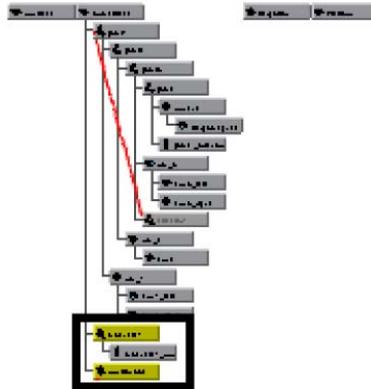


ArmControl and IK Handle are not parented into the SwivelBase hierarchy. Because ArmControl and IK Handle control the movements of the mechanical

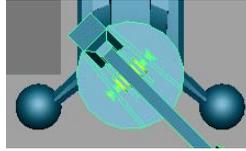
arm, they must be parented into the SwivelBase hierarchy in order for the arm to rotate when SwivelBase is rotated.

To parent the IK Handle into the skeleton hierarchy

- 1 In the Hypergraph, select the SwivelBase node.
- 2 In the Channel Box, reset the Rotate Y channel for SwivelBase to 0.
Setting any previous transformations to zero is necessary before parenting the IK Handle beneath SwivelBase in the hierarchy.
- 3 Unselect the SwivelBase node before performing the following steps.
- 4 In the Hypergraph menu, select View > Frame All.
The Hypergraph displays all the nodes for the scene.
- 5 In the Hypergraph, Shift-select the nodes named ikHandle1, ArmControl, and SwivelBase.
The order of selection is important. You want to Shift-select the items that will be the child nodes first, and then Shift-select the item that will be the parent node last.
- 6 In the main menu, select Edit > Parent (Hotkey p).
In the Hypergraph, IKHandle and ArmControl are now child nodes for the SwivelBase node in the skeleton hierarchy.



- 7 In the Hypergraph, select only SwivelBase.
- 8 Using the Channel Box, set Rotate Y for SwivelBase to 45.



When SwivelBase is rotated, the mechanical arm rotates because of its relationship within the hierarchy.

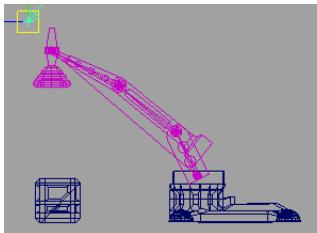
- 9 Reset the Rotate Y value for SwivelBase back to 0.

Finally, you want to limit how far Arm Control can translate away from the base of the model so the arm does not extend to a fully straightened position.

To determine the settings for the translation limits you need to examine the position of the mechanical arm in various positions while simultaneously viewing the translation values in the Channel Box.

To determine translation limits for the Arm Control

- 1 In the Hypergraph, select ArmControl.
- 2 Ensure the Channel Box is open so you can view the translate channels for ArmControl.
- 3 In the view, select Panels > Orthographic > side.
- 4 Dolly the side view so you can fully view the mechanical arm.
- 5 Select the Move Tool.
- 6 In the side view, middle-click and drag the mouse diagonally until the mechanical arm is fully extended as shown below.

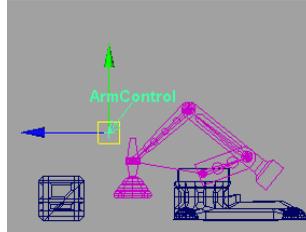


- 7 In the Channel Box, observe the number that displays in the Translate Y and Z channels.

When the arm is diagonally articulated above the cargo box, it is almost fully extended when the translation values are roughly as follows:

- Translate Y: 7.5
- Translate Z: 1

- 8 In the side view, press the middle mouse button and drag the mouse diagonally until the mechanical arm is almost fully compressed as shown below.



- 9 In the Channel Box, observe the numbers that display in the Translate Y and Z channels.

When the arm is articulated to a compressed pose, the translation values are roughly as follows:

- Translate Y: -13
- Translate Z: -13

These minimum and maximum values will be used in the steps that follow as the minimum and maximum translation limits for the ArmControl.

To set translation limits for the Arm Control

- 1 In the Hypergraph, select the ArmControl node.
- 2 Open the Attribute Editor (Window > Attribute Editor or click on the

Show/Hide Attribute Editor icon  on the Status Line).

The Attribute Editor displays the nodes associated with Arm Control.

- 3 In the Attribute Editor, click the ArmControl tab.
If the ArmControl tab is not visible, use the left and right scrolls arrows to scroll the view sideways to view it.

- 4 In the Attribute Editor, expand the Limit Information attributes, and then the Translate attributes.
- 5 In the Translate attributes, click the Trans Limit Y and Z check boxes for both the Min and Max values, then set the following values:
 - Trans Limit Y (Min): -13
 - Trans Limit Y (Max): 7.5
 - Trans Limit Z (Min): -13
 - Trans Limit Z (Max): 1
- 6 Hide the Attribute Editor.
- 7 Select ArmControl.
- 8 In the side view, move ArmControl to test the range of movement for the IK system.
The mechanical arm moves with a limited range of motion.

Simplifying the display of a hierarchy

The number of nodes in the IK system's hierarchy at this point may appear a bit overwhelming, but it allows the posing of the mechanical arm to be efficient and effective. You can simplify the display of the nodes by collapsing portions of the hierarchy.

To collapse the display of the hierarchy

- 1 In the Hypergraph, right-click on joint1, and choose Collapse from the pop-up menu.
The child nodes below joint1 disappear, simplifying the display of the hierarchy. The small arrowhead on the corner of the joint1 node indicates that a hierarchy exists below joint1. If you later need to view or select these nodes you can expand them again.
- 2 Collapse the ArmControl and IKHandle nodes.
The display of the hierarchy is simplified as shown below.



The IK system is now rigged and ready for animation.

Applying parent constraints on an IK system

To animate the cargo box so that it is picked up by the mechanical arm you need a method of constraining the cargo box to the mechanical arm so that it moves with the arm at specific times in the animation. This is accomplished using a *parent constraint*.

When a parent constraint is applied to an object, the constrained object behaves as if it were a child node of the constraining object.

An object can have multiple constraints applied to it. The influence each constraint has on the object can be changed by modifying the weight attribute for the parent constraint. The weight attribute is an animatable parameter and can be set to be on or off at specific frames.

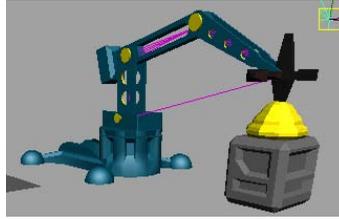
In this lesson, the cargo box is constrained to both the ArmControl and the Platform on the floor. By animating the weight attributes for these two parent constraints, the cargo box will move from one location to another depending on which constraint is influencing it at specific times in the animation.

To set up the parent constraints you need to position the constrained object at the locations where the influence will begin or end, and then set the parent constraints.

To set parent constraints for the cargo box

- 1 In the side view menu, select Shading > Smooth Shade All.
- 2 In the side view menu, select Panels > Perspective > persp.
- 3 In the Hypergraph, select ArmControl.
- 4 In the Channel Box, set the Translate Y and Z values as follows:
 - Translate Y: -9
 - Translate Z: 0

The arm is repositioned so that the cargo magnet is directly above the cargo box as shown below. Track, tumble, or dolly the view if required so you can see the mechanical arm unobstructed.



- 1 In the Hypergraph, with ArmControl still selected, Shift-select the CargoBox node.

The order of selection is important when you apply constraints. You must select the constraining object first, and the item to be constrained second.

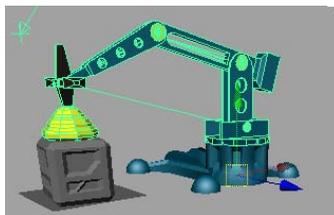
- 2 In the main menu, select Constrain > Parent > .
- 3 In the Parent Constraint Options window, do the following:
 - Select Edit > Reset Settings to set the tool to its default settings.
 - Click the Add button.

CargoBox is constrained to ControlArm. A new constraint node has been created below the CargoBox node in the Hypergraph. If you reposition the mechanical arm, CargoBox moves with it because of the constraint that is now applied.

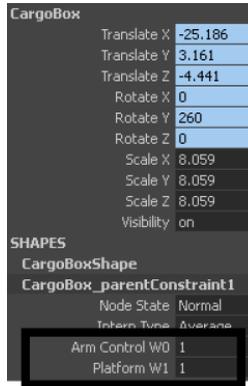


- 4 In the Hypergraph, select only SwivelBase.
- 5 In the Channel Box, set the Rotate Y value for SwivelBase as follows:
 - Rotate Y: -100

The mechanical arm and the cargo box are positioned as shown below.



- 6 In the Hypergraph, select Platform and then Shift-select the CargoBox node.
- 7 In the main menu, select Constrain > Parent.
CargoBox is constrained to Platform. This is the second constraint for the CargoBox.
- 8 In the Hypergraph, select only CargoBox.
- 9 In the Channel Box, view the parent constraints for CargoBox.



CargoBox has two parent constraints applied to it: Arm Control and Platform. In the Channel Box, the numerical settings beside ArmControl W0 and Platform W1 represent the influence weight each parent constraint has on CargoBox. At this point, each influence weight is set to a value of 1. If you move the mechanical arm, the cargo box will move somewhere between its first and second position. This occurs because each constraint has equal influence on the cargo box at this point.

Before you can animate the objects in the scene, you also need to reset the default weight values for each constraint, so that one parent constraint has full influence on CargoBox when the other does not.

To set the weighting for the parent constraints

- 1 In the Hypergraph, select only CargoBox.
- 2 In the Channel Box, set the following:
 - Platform W1: 0
- 3 Select only SwivelBase.

- 4 In the Channel Box, set the Rotate Y value to 0.
The mechanical arm and cargo box are rotated back.
- 5 In the Hypergraph, select only CargoBox.
- 6 In the main menu, select Constrain > Set Rest Position.
Setting a rest position for CargoBox sets a position for the cargo box when no constraints are acting upon it. If the rest position is not set, CargoBox will be repositioned to the second constraint position.
- 7 In the Channel Box, set the following:
 - ArmControl W0: 0

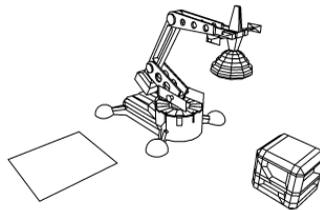
At this point, both parent constraints are set to have no influence on the cargo box because their weight attributes are set to zero. When you animate the components in the scene, the weight attributes for the parent constraints will be set and keyframed depending on which constraining object (Arm or Platform) is required to have full influence.

In the next steps, you plan the action for animation of the IK system.

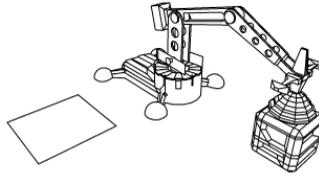
Planning an animation for an IK system

Now that the mechanical arm and cargo box are fully set up for animation it's a simple matter to pose the arm and set keyframes. Prior to setting any keyframes you should take a few moments to plan the action for the scene.

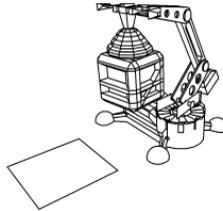
Planning the action usually involves sketching a simple storyboard, and writing out a brief timeline for the action for each component. In this lesson we've provided a simple version for you. The basic premise for the action is as follows:



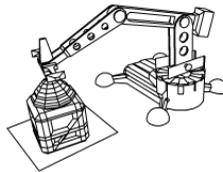
- The cargo box is positioned directly in front of the mechanical arm.



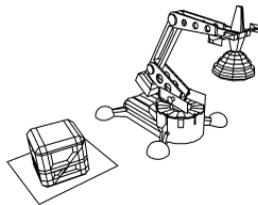
- The mechanical arm extends, and positions the cargo magnet to be directly touching the top of the cargo box.



- The mechanical arm extends upwards, lifting the cargo box, and rotates to the side.



- The mechanical arm lowers the box, placing it in a new position.



- The mechanical arm extends upwards, leaves the cargo box in the new location, and rotates back to its original position.

The table below breaks down the action indicating what action occurs for each object at the specified keyframes.

Frame	Mechanical Arm	Cargo Box
1	Collapsed position	Initial position
30	Vertically extended position	Initial position
60	Vertically lowered so that cargo magnet is positioned on top of cargo box	Initial position
80	Vertically lifted to extended position with cargo box	Cargo box lifted with mechanical arm
100	SwivelBase rotated along with arm to side	Cargo box repositioned along with mechanical arm
120	Vertically lowered along with cargo box to second position above surface	Cargo box is lowered along with mechanical arm to second position
140	Vertically lifted to extended position	Second position
160	SwivelBase rotated with arm to original position	Second position
180	Collapsed position	Second position

To pose the mechanical arm you select either the ArmControl or SwivelBase nodes and set a keyframe depending on which component you want to pose:

- ArmControl controls the movement of the mechanical arm.
- SwivelBase controls the rotation of the arm.
- CargoBox's position is based on the parent constraint weights.

Animating an IK system

The animation sequence occurs over a 180 frame period. Before you begin to set keyframes you set the playback range in the Time Slider.

To set the playback range in the Time Slider

- In the Animation Time and Range Slider, set the following start and end times for the animation:
 - Playback Start Time: 1
 - Playback End Time: 180



To simplify the keyframe workflow, you first set keyframes for the mechanical arm, and then set keyframes for the weight attributes of each parent constraint for the cargo box.

To set keyframes for the mechanical arm

- 1 In the Time Slider, go to frame 1.
- 2 Select only ArmControl.
- 3 In the Channel Box, set the following values:
 - Translate Y: 0
 - Translate Z: -13
- 4 From the main menu, select Animate > Set Key.

To set the remaining keyframes for the arm and the swivel base, use the table below as a guide, and set the attributes in the following order:

- Set the frame using the Time Slider.
- Set the attributes for the object in the Channel Box.
- Set the keyframe (Animate > Set Key).

NOTE Don't be concerned if the list of keyframes indicates to set a key twice for the same position over two frames. This is necessary to animate the parent constraint weights over a period of one frame.

Frame	Select	Set attribute	Select
1	ArmControl	TransY: 0 TransZ: -13	Set Key
30	ArmControl	TransY: 7 TransZ: -13	Set Key
59	ArmControl	TransY: -9 TransZ: 0	Set Key
60	ArmControl	TransY: -9 TransZ: 0	Set Key
80	ArmControl	TransY: 6 TransZ: -13	Set Key
80	SwivelBase	RotateY: 0	Set Key
100	SwivelBase	RotateY: -100	Set Key
100	ArmControl	TransY: 6 TransZ: -13	Set Key
120	ArmControl	TransY: -9 TransZ: 0	Set Key
121	ArmControl	TransY: -9 TransZ: 0	Set Key
140	ArmControl	TransY: 6 TransZ: -13	Set Key
140	SwivelBase	RotateY: -100	Set Key
160	SwivelBase	RotateY: 0	Set Key
160	ArmControl	TransY: 6 TransZ: -13	Set Key
180	ArmControl	TransY: 0 TransZ: -13	Set Key

Once the keyframes for the mechanical arm are set you can use the playback controls to playback the motion for the mechanical arm.

To playback the animation using the playback controls

- 1 Click play on the Time Slider playback controls to play the animation.
The mechanical arm animates over 180 frames. The cargo box does not move because the weight attributes for the parent constraints are set to zero and have not yet been set for the animation.
- 2 Click stop on the Time Slider playback controls to stop the playback.

In the next steps, you set keyframes for the parent constraint weight attributes so the cargo box is lifted, moved, and placed in the second location. You set keys for the weight attributes on the parent constraint node.

To set keyframes for the parent constraint weights

- 1 In the Time Slider, go to frame 1.
- 2 In the Hypergraph, select the node for the parent constraint directly below CargoBox.



- 3 In the Channel Box, set the following values (if they are not set already):
 - ArmControl W0: 0
 - Platform W1: 0
- 4 In the Channel Box, select the ArmControl and Platform channels by shift-clicking on the names.
- 5 Right-click on either one of the selected names.
A drop-down list appears.
- 6 From the drop-down list, choose Key Selected.
Key Selected sets a keyframe for the selected attributes only.

To set the remaining keyframes for the parent constraint weights, use the table below as a guide, keeping in mind that you set the attributes in the following order:

- Set the frame using the Time Slider.
- Set the parent constraint attributes in the Channel Box.
- Right-click on the selected names and select Key Selected.

Frame	Select	Set attribute	Select
1	CargoBox parent-Constraint	ArmControl W0: 0 Platform W1: 0	key selected
59	CargoBox parent-Constraint	ArmControl W0: 0 Platform W1: 0	key selected
60	CargoBox parent-Constraint	ArmControl W0: 1 Platform W1: 0	key selected
120	CargoBox parent-Constraint	ArmControl W0: 1 Platform W1: 0	key selected
121	CargoBox parent-Constraint	ArmControl W0: 0 Platform W1: 1	key selected

To playback the animation using the playback controls

- 1 Click play on the Time Slider playback controls to play back the animation.
The mechanical arm moves through the motion of the animation and picks up and moves the cargo box to the new location.
- 2 Click stop on the Time Slider playback controls to stop the playback.
You have completed this lesson.

Beyond the lesson

In this lesson, you completed the animation of a mechanical arm model that picked up and moved a cargo box. You learned how to:

- Construct a model hierarchy to provide relationships between the components for posing and keyframing.

- Construct a single chain skeleton to pose the arm model using Inverse Kinematics.

It is common to create characters with multiple joint chains. For example, two and four-legged characters would use multiple joint chain skeletons to assist with posing their hierarchical structure. See Skeleton hierarchy in the Maya Help. For a tutorial on creating a multiple chain skeleton see Lesson 1: Skeletons and Kinematics ([Introduction](#) on page 314).
- Parent the components of the mechanical arm model into the skeleton hierarchy.

Parenting is one method for combining a model into a skeleton hierarchy. When using characters that have bending or deforming features a more common technique is to use Skin > Bind Skin. For more information see Skeletons and skinning in the Maya Help. For a tutorial on smooth skinning see Lesson 2: Smooth Skinning ([Introduction](#) on page 330).

An easy method to parent one node to another using the Hypergraph is to drag one node over top of the other using your middle mouse button.
- Create an IK system with a control object for manipulation.

It is very common to create characters with multiple joint chains and multiple IK handles to control them. Control objects simplify the selection and posing of the character.
- Use three types of constraints (Point, Orient, Parent) to control components of the IK system and other components in the scene.

Constraints are useful for animating the attributes of one object based on the attributes of another. Other types of constraints not covered in this lesson include aim and scale constraints. See Aim constraints and Scale constraints in the Maya Help.

Using the parent constraint, the platform could also be animated to move the cargo box as if it were on an elevator, or a conveyor belt.
- Apply limits to the movement of the IK system.

In this lesson, you constrained the translation of the control object in order to constrain the movement of the IK. It is not recommended that you set limits on the rotation of an IK system as it limits how the solver calculates the position of the joints. See Setting up joints for posing and animation in the Maya Help.

Although not covered in this lesson, it is possible to simplify the display of attributes in the Channel Box so that only the animatable attributes associated with a character are displayed.
- Pose and set keyframes for an IK system.

Character Setup

7

Introduction

A typical 3D character can be made up of many surfaces and components. To ensure that the character animates in the way that you want, it is important to carefully plan the process of *character setup*.

Character setup or *rigging* is the general term used for the preparation of 3D models with their accompanying joints and skeletons for animation.

Depending on the model to be animated, character setup can involve the following techniques:

- Creating a skeleton with joints that acts as a framework for the 3D character model. You set limits on the joints so they rotate in a convincing manner. When you animate the character, you position the character's joints using either forward or inverse kinematic techniques (FK or IK).
- Binding the 3D surfaces to the skeleton so that they move together. The process of binding may also include defining how the character's joints bend or how the skin surfaces bulge to simulate muscles.
- Defining and setting constraints for particular animated attributes in order to restrict the range of motion or to control an attribute based on the movement of another.
- Grouping surface components such as CVs into sets called *clusters* so that parts of the character can be animated at a more detailed level.

This chapter introduces you to the most common character setup features:

- Lesson 1 Skeletons and kinematics: [Introduction](#) on page 314

- Lesson 2 Smooth skinning: [Introduction](#) on page 330
- Lesson 3 Cluster and blend shape deformers: [Introduction](#) on page 341

Preparing for the lessons

- 1 To ensure the lessons work as described, select the Animation menu set. Unless otherwise noted, the procedures in this chapter assume the Animation menu set is selected.
- 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](#) on page 25.
- 3 Make sure that Soft Selection is turned off by opening the Attribute Editor with the Select Tool active and unchecking the box marked Soft Select.

Lesson 1: Skeletons and kinematics

Introduction

After you model the surfaces that make up a human, animal, or other character, you create a skeleton and bind it to the character's surface (called *skin*).

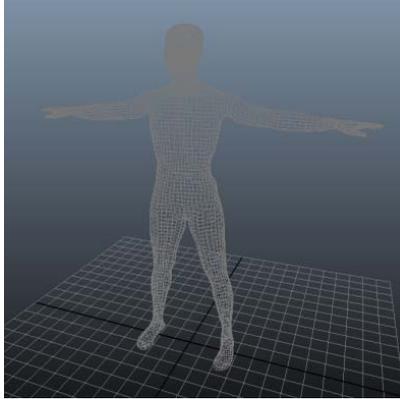
A skeleton provides a structure for animating the character. When you create a skeleton in Maya, you create a series of bones with joints in the skeletal locations where you want the character to bend or twist. (You can animate a skeleton without having a skinned character.)

In this lesson, you learn the techniques for creating a simple skeleton for a human character. You will learn how to:

- Create joints for a character.
- Name joints in the Hypergraph.
- Use symmetry when creating a character.
- Parent joints into an existing skeleton.
- Use Inverse Kinematics (IK) techniques to pose a skeleton.

Learning how to bind a character to a skeleton is described in the next lesson.

Open the scene for the lesson



In this lesson, you'll load a scene that contains a human character. During the lesson, you'll create and animate a skeleton for the character.

To open the scene

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 314.
- 2 Open the scene file named `Skeleton.mb`.

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

```
GettingStarted/CharSetup/Skeleton.mb
```

The character displayed in the scene, Jackie, is a template, so you cannot unintentionally select it during the lesson. Jackie was modeled in the Da Vinci pose, the position that's easiest for creating a skeleton.

Creating joints

A skeleton is made of bones and joints. When you create a skeleton in Maya, you create a series of joints in the skeletal locations where you want the character to bend or twist.

A common technique for creating a skeleton is to create several independent joint chains—one for each arm, one for each leg, one for the spine/head—then

group the chains together to create a single skeletal hierarchy. In the steps that follow, you create the joints for the legs.

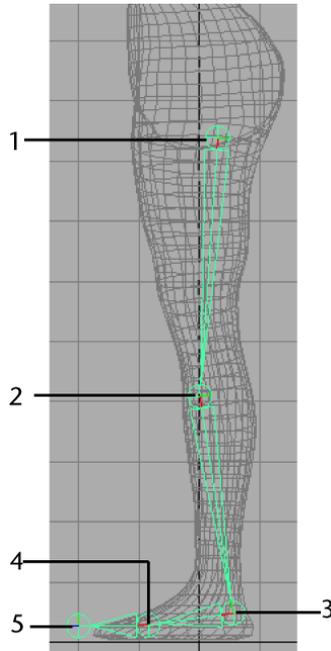
To create joints for the legs

- 1 Select Window > Settings/Preferences > Preferences. In the Preferences window, click the Kinematics Category. Enter 0.4 for the Joint size, then click the Save button.

This displays the joints smaller when you create them in the next steps. At the default size, 1.0, the size of the joints makes them hard to position accurately for this character.

NOTE The default joint size of 1.0 has been used to create the joints pictured in the following illustrations. The joints you create may appear smaller than those pictured in this lesson.

- 2 Select Skeleton > Joint Tool. This is the tool for creating the joint chains that make up a skeleton.
- 3 In a side view, click at the hip, knee, ankle, ball of foot, and toe to create joints at these positions (see the following figure). Make sure the knee joint is in a position that creates a slight forward bend. The forward bend ensures that you will be able to animate the leg easily in a direction natural for a leg.



TIP Press Alt+b (Windows) or Option+b (Mac OS X) to toggle between default background colors in the scene view.

- 4 Press Enter (Windows and Linux) or Return (Mac OS X) after creating the toe joint. This completes the joint chain.
- 5 Select Window > Hypergraph: Hierarchy.

The Hypergraph is a convenient place to select, rename, and parent objects. It is similar to the Outliner, but it has features tailored for character setup. For example, it depicts all parent-child relationships in an easy-to-read indented format.

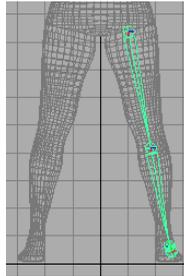
The Hypergraph shows the default names given to the joints just created: joint1, joint2, and so on. The joints have a hierarchical relationship. Joint1 is the parent of joint2, which is the parent of joint3, and so on. Joint1 is the root of the hierarchy. If you reposition joint1, you reposition the whole *joint chain*.

In the scene view, joints are represented by spherical icons. *Bones* separate the joints, and are represented by elongated pyramid icons. The narrow part of a bone points in the downward direction of the hierarchy.

The reason you create the hip joint first and the toe joint last is to have the hip at the top of the hierarchy and the toe at the bottom. You'll usually want the toe (and other joints) to move whenever you move the hip, but not necessarily vice versa. In general, joint chains emanate from the interior of the character outward.

- 6 Rename the joints as `left_hip`, `left_knee`, `left_ankle`, `left_ball`, and `left_toe`. To rename a joint, right-click the joint name in the Hypergraph and select `Rename` from the drop-down menu. Type the new name and press `Enter`.
- 7 In the front view, click the `left_hip` joint to select it. Move it along the X-axis to the center of the top of the left leg. (In this lesson, *left* and *right* refer to directions from Jackie's point of view, not from your view of the scene.)

As mentioned before, when you move a joint, all joints lower in the hierarchy move with it. If you press `Insert` (Windows and Linux) or `Home` (Mac OS X) while a joint is selected, you can move the joint without moving joints below it in the hierarchy. (To exit this mode, press `Insert` or `Home` again.)



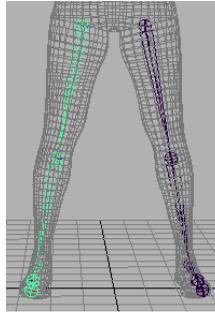
NOTE You may need to rotate `left_hip` so that the skeleton fits inside the leg. It's unnecessary to fit the skeleton perfectly inside the character, as it won't be displayed when you render an image from the scene.

- 8 To create the joints for the other leg, you can save time and ensure symmetry by duplicating the existing leg joint chain with mirroring. With `left_hip` selected, select `Skeleton > Mirror Joint >` . In the options window, turn on `YZ` for `Mirror Across`.

Jackie's legs straddle the `YZ` plane, so mirroring the joint chain across the `YZ` plane positions the duplicate joint chain in the desired location. This operation illustrates that Jackie's original position affects the ease with which you can create the skeleton. Had Jackie been positioned away from

the origin, you would not have been able to use Mirror Joint conveniently to duplicate the leg's joint chain.

- 9 Enter left in the Search For field and enter right in the Replace With field. The Replacement names for duplicated joints feature automatically replaces the names of all duplicate joints with the specified joint prefix.
- 10 Click Mirror.
The right leg bones and joints appear in a mirrored position.



In the next steps, you create a joint chain for the spinal column. You also extend a joint from the upper neck region of the joint chain so that you can animate the jaw.



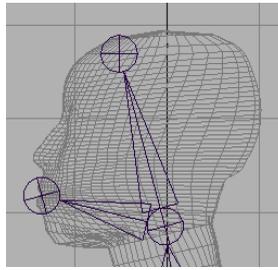
To create joints for the spine and jaw

- 1 In the side view, use the Joint Tool as follows to create a series of joints at the locations shown in the figure:
 - Start at the base of the spine near the existing hip joints (`left_hip` and `right_hip`) and end at the top of the head.
 - Make sure you create the first joint a little bit away from the existing hip joint displayed in the front view, otherwise the first joint will be connected to the hip joint.
 - Press Enter (Windows and Linux) or Return (Mac OS X) when you are finished creating the joint chain.

With the exception of the joints at each end of the joint chain, the joints are located where the character is likely to bend or twist at the spine and neck.

The S-shaped curvature of the joint chain resembles Jackie's spinal curvature. This makes it easier to animate the character's torso and neck naturally.

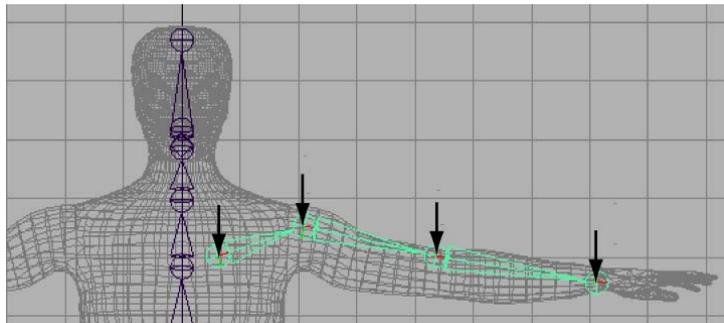
- 2 Starting at the base of the spine, name the joints `back_root`, `pelvis`, `lower_back`, `mid_back`, `upper_back`, `lower_neck`, `upper_neck`, and `crown`.
- 3 To set up the skeleton for jaw movement, extend a joint from the `upper_neck` joint as follows:
 - With the Joint Tool selected, click the `upper_neck` joint in the Hypergraph to select it.
 - Click to create a new joint near the lips, and press Enter or Return.
 - Name the new joint as `jaw`.



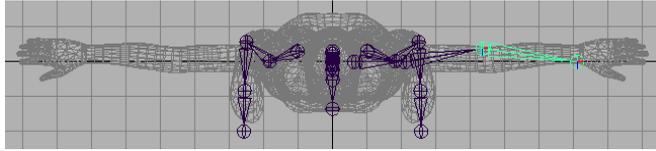
Creating joints for the arms is similar to creating joints for the legs.

To create joints for the arms

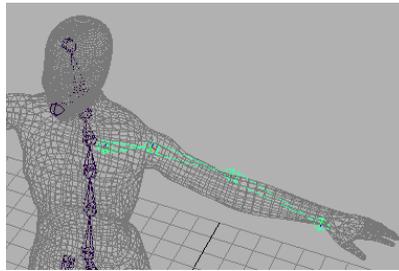
- 1 In the front view, create a series of joints at the locations shown in the figure. Start at the pectoral region (near the `upper_back` joint) and end at the wrist.



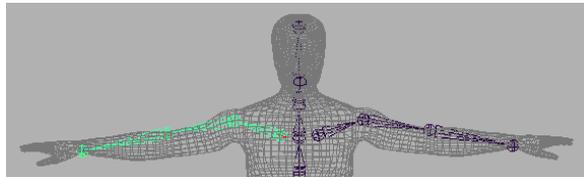
- 2 Name the joints `left_arm_root`, `left_shoulder`, `left_elbow`, and `left_wrist`.
- 3 In the top view, select the `left_elbow` joint, select the Move tool, press Insert (Windows and Linux) or Home (Mac OS X), then move the joint to the back of the arm. Press Insert or Home again.



Moving the joint to the back of the arm creates a bend at the elbow. This will make it easier to animate the character's arm in the direction an arm naturally bends.



- 4 In the perspective view, select `left_arm_root`. Select Skeleton > Mirror Joint. This creates a copy of the left arm's joint chain for the right arm and renames them in the process.
The right arm bones and joints appear in a mirrored position.



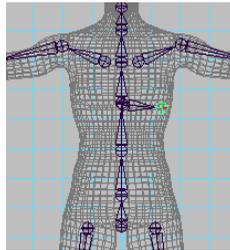
Adding joints to a skeleton

When you bind a character to a skeleton (described in the next lesson), the skeleton provides a structure for the character's skin and prevents the skin from collapsing as you pose the skeleton.

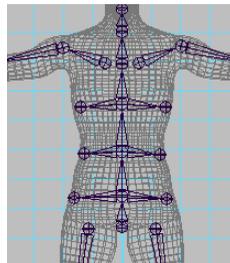
It's useful to add extra joints in areas of the character where you want the surface to keep its volume upon deformation. In the next steps, you'll add ribs to Jackie's skeleton. Although you will not bind the character in this lesson, the technique of adding ribs is important to learn so that when you do bind a character, the skin in the torso area will not collapse as you pose shoulder and spinal rotations. The rib joints are for structure only, not for posing the skeleton.

To add ribs to the skeleton

- 1 In the Hypergraph, click the mid_back joint to highlight the joint in the front view. Remember the location of the joint in the front view so you can select it later.
- 2 Select Skeleton > Joint Tool.
- 3 In the front view, click on the mid_back joint to select it again.
- 4 Click to the side of the joint to add a rib bone, then press Enter (Windows and Linux) or Return (Mac OS X).



- 5 With the same technique as in the prior steps, create two more ribs for the left side. The ribs extend from the mid_back, lower_back, and pelvis joints.



- 6 For easy identification in the Hypergraph, you can optionally name the ribs as desired. For instance, name them `left_top_rib`, `left_mid_rib`, `left_bottom_rib`, and so on.
- 7 Mirror each of the left rib joints to create ribs for the right side of the skeleton.

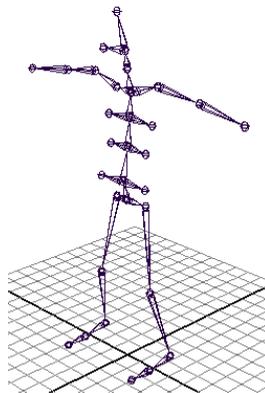
Creating a skeleton hierarchy

So far you have created five separate joint chains: one for the spine, and one for each arm and each leg. You need to create a single hierarchy from the five joint chains so you can move all of them as a single unit. To create the single hierarchy, you parent the arms and legs to the nearest joint in the spine.

To parent the arm and leg joint chains to the spine

- 1 In the Hypergraph, middle-drag `left_arm_root` to `upper_back`.
When you parent one joint to another, Maya creates a bone that connects the parent to the child. In this case, the bone connects `upper_back` to `left_arm_root`.
- 2 Middle-drag `right_arm_root` to `upper_back`.
- 3 Middle-drag `left_hip` to `back_root`.
- 4 Middle-drag `right_hip` to `back_root`.

Now if you need to move the entire skeleton, you can move `back_root`—the root of the hierarchy. The completed skeleton is shown in the following figure (with Jackie hidden).



Forward and inverse kinematics

There are two techniques for posing a skeleton: forward kinematics and inverse kinematics.

Forward kinematics (FK)

To pose a character with forward kinematics, you rotate each joint individually until you get the desired positioning. For example, to move a hand to some location, you must rotate several arm joints to reach the location.

When you animate a skeleton posed with forward kinematics, Maya interpolates the joint rotations starting with the root joint, then the root's child joints, and so on down through the skeleton's action hierarchy. Maya proceeds forward through the action hierarchy, starting at the root joint.

Forward kinematics is intuitive for creating simple arc motions, but it's tedious if you are animating a complex skeleton. It's also not intuitive for specifying goal-directed motion. For example, to move a hand to some location, it's not obvious how to rotate the joints in an arm.

Inverse kinematics (IK)

With IK, you create an extra control structure, an *IK handle*, for certain joint chains such as arms and legs. An IK handle lets you pose and animate an entire joint chain by moving a single manipulator.

As you pose the IK handle, it automatically rotates all the joints in the joint chain. For example, if you move a hand to a doorknob, the other joints in the arm rotate to accommodate the hand's new positioning.

IK is more intuitive than forward kinematics for goal-directed motion because you can focus on the goal rather than on how you need to rotate each joint to achieve that goal.

Posing and animating using inverse kinematics

Next, you create IK handles that you'll later use to pose the arms and legs. The next steps describe some initial setup you should perform before creating the IK handles.

To set up the character prior to creating IK handles

- 1 Select the root of the hierarchy, `back_root`, and then select `Skeleton > Set Preferred Angle`.

This sets the current joint angles throughout the skeleton as the preferred angles. This is a useful step after you complete a skeleton. Maya thereafter uses the current bend in the knees and elbows as the preferred initial rotation direction of these joints during inverse kinematics (IK) posing. This makes it easier to pose the character with motion that is natural for a human character.

- 2 Select Jackie in the Outliner and, from the main menu, select Display > Hide > Hide Selection. (You must select Jackie from the Outliner because Jackie is a template object.) By hiding Jackie, you'll lessen scene clutter as you pose the skeleton in the following steps.

To create, pose, and animate IK handles for the legs

- 1 Select Skeleton > IK Handle Tool >  .
- 2 In the Tool Settings window, make sure Current Solver is set to ikRPsolver. This type of IK solver has characteristics that work well for this application.
- 3 In the perspective view, click left_hip and left_ankle.
This creates an IK handle that lets you control all joints from left_hip through left_ankle. The handle's main manipulator is at the left_ankle.
The IK handle is the selected object after you create it. If you unintentionally cancel the selection of the handle, you can select it again by clicking the left_ankle—the last joint you clicked after using the IK Handle Tool. Whenever you select a handle, make sure you do not select joints or other objects along with the handle. You can check the Outliner to confirm your selection.
- 4 Go to the start of the playback range.
- 5 With the IK handle selected, select Animate > Set Key to key the leg's current position at the first frame.
- 6 Go to frame 12.
- 7 In the side view, use the Move tool to drag the IK handle up and to the left (see illustration), as if Jackie were stepping up a staircase. The foot and knee move while the hip stays in place.



- 8 Set another key for the IK handle.
- 9 Go to frame 24. Move the IK handle back to its prior position. Set another key.
- 10 Play the animation to see the leg step up and down during the first 24 frames.
This completes a simple animation of the leg using an IK handle to control its position.

To practice additional IK techniques

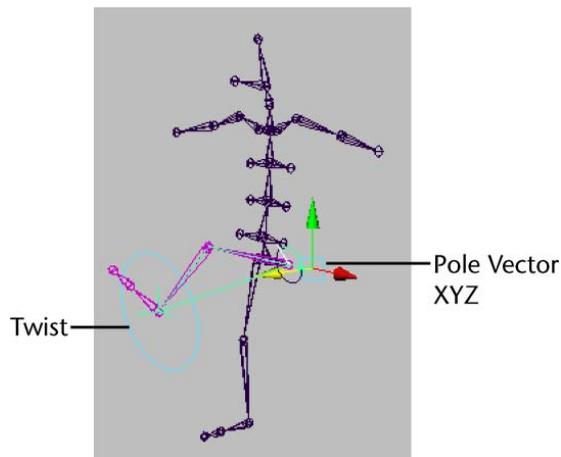
- 1 Go to the start of the animation.
- 2 In a perspective view, practice posing the leg in various directions by moving the IK handle.

No matter how far you drag the handle manipulator, the joints of the leg will not stretch beyond the straight leg position. This is desirable; you do not want the size of a skeleton to change as you pose it. However, if you drag a leg joint, the bone hierarchically above that joint will lengthen. This is why it's important to check that you haven't selected a joint with the IK handle before you move the handle.

As you drag the handle to some positions, you might notice the leg joints flip abruptly (see the next figure for an example position). It's therefore hard to control the leg positioning in this region.

The default IK handle (*IK Rotate Plane handle*) has manipulators you can use to avoid joint flipping. With the handle selected, select **Modify > Transformation Tools > Show Manipulator Tool**. Move the Pole Vector XYZ manipulator to a slightly different position (see the following figure). If this doesn't solve the problem, rotate the Twist manipulator to rotate the leg. You can key the Pole Vector XYZ and Twist attributes to fixed values to avoid the flipping as the character moves.

A more precise way to avoid joint flipping, which requires some initial setup, is to use a Pole Vector Constraint. For more information see Pole Vector constraints in the *Character Setup* guide.



- 3 Repeat the preceding steps for the right leg. (Create an IK handle for the right_hip to right_ankle, then practice posing and animating the handle.)

NOTE If you want to move the entire skeleton, group the back_root and all IK handles under a single node, select the group node, and then use the Move tool. With this grouping, the motion of the entire skeleton won't conflict with the keys you set for the handles.

To create, pose, and animate IK handles for the arms

- 1 Select **Skeleton > IK Handle Tool**.
- 2 Click left_shoulder, and then click left_wrist. This creates an IK handle for the left arm.
- 3 Select **Skeleton > IK Handle Tool**.

- 4 Click `right_shoulder`, then click `right_wrist`. This creates an IK handle for the right arm.
- 5 Practice posing and animating the handles.

Posing and animating using forward kinematics

As the last part of the lesson, you can optionally practice posing and animating the character by rotating joints that are not controlled by IK handles—the ball joints of the feet, and joints from the `back_root` through the `upper_neck`. Rotating such joints is forward kinematics.

With the exception of the `back_root`, you'll invariably rotate rather than move the joints that aren't part of an IK handle. If you move the joints, the bones lengthen. This deforms bound skin undesirably unless you are creating cartoon-like distortions.

Because you've already set keys for IK handles, some joint rotations on non-IK handle joints will be restrained by the IK handle positioning. If this interferes with the intended poses, you can remove all prior animation from an IK handle by selecting `Edit > Keys > Delete Keys`. You can also use `Delete Keys` to remove all animation from keys you've set with forward kinematics.

Beyond the lesson

In this lesson you learned how to:

- Create a skeleton with bones and joints.
- Pose the skeleton using Inverse Kinematics.

Additional things to consider when working with skeletons:

- The appropriate number of joints in a skeleton depends on the anatomical parts of the character you want to manipulate. More joints means finer control at the expense of greater complexity.
- As you created the skeleton in this lesson, you ended the arm's joint chain at the wrist. This prevents you from animating hand motion. If you need to animate hand motion or even finger motion, you would need to make additional joints and IK handles. The same applies to foot and toe motion.
- When you create a skeleton, you can animate a character bound to it to produce natural skin deformations. Although you animated an unskinned

skeleton in this lesson, it's more common to animate a skinned skeleton. Binding a character is the topic of the next lesson.

- It's typically best to animate the entire skeleton from pose to pose at desired frames. It's hard to get desired results by animating one limb for a frame range, another limb for a frame range, and so on.

There are many other ways to work with skeletons not described in this lesson:

- You can blend or switch between IK and forward kinematics on joints controlled by an IK handle.
- There are other types of IK handles that provide different controls for manipulating parts of a skeleton. Especially noteworthy is the IK spline handle, which makes it easy to animate the twisting, wavy motion in tails, necks, spines, snakes, and so on.

For more details on these and other features, please refer to the Maya Help.

Lesson 2: Smooth skinning

Introduction

After you create a skeleton, you *bind* it with the character's surface so that the surfaces move with the skeleton during animation. Binding is also called skinning, and a character's surface after binding is called a *skin*.

It is important that the character's skin deforms naturally as the skeleton moves. Near joints, the skin bulges or indents when you rotate the joints.

In Maya, the skin deforms because the surface's vertices (or CVs) move in response to the rotation of adjacent joints. The vertices are known as skin points. This is useful for animating elbows, shoulders, necks, and so on.

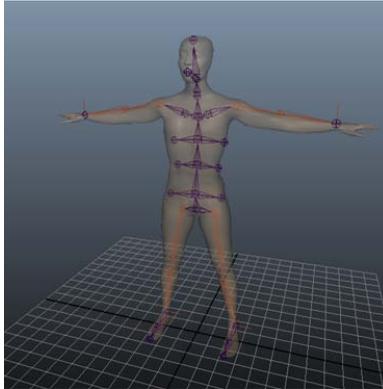
By default, the influence a joint has on a skin point's movement depends on how close it is to that joint. You can edit skin point weighting to change the default movement.

In this lesson you learn how to:

- Bind a skeleton using a smooth bind technique.
- View and modify skin weights using the Paint Skin Weights Tool.

- Use influence objects to enhance the skin deformation of a character.

Open the scene for the lesson



In this lesson, you will load a scene containing a human character and skeleton. Each leg and arm of the skeleton has an IK handle that lets you pose the limbs conveniently. If you completed the prior lesson, the character, skeleton, and IK handles will be familiar.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 314.

- 2 Open the scene file named `SmoothSkin.mb`.

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

```
GettingStarted/CharSetup/SmoothSkin.mb
```

The character in the scene, Jackie, is displayed in a transparent fashion so you can see the skeleton beneath the surface.

- 3 Select `Window > Settings/Preferences > Preferences`. In the Preferences window, click the Kinematics Category. Enter 0.4 for the Joint Size, then click the Save button.

This displays smaller joints. At the default size of 1.0, the large joints obstruct your view of the skin.

Smooth binding a skeleton

You use Jackie as the surface to be bound to the skeleton.

To bind the skeleton using smooth skinning

- 1 Select Jackie.
- 2 Shift-select the back_root joint.

The back_root joint is at the base of Jackie's spine. If you are not sure which joint it is, select it in the Hypergraph (Window > Hypergraph: Hierarchy).
- 3 Select Skin > Bind Skin > Smooth Bind > .
- 4 In the Smooth Bind Options window, set Max Influences to 3, then click Bind Skin.

A Max Influences value of 3 specifies that three joints influence each skin point. By default, the joint closest to the point has the most influence. The second most influential joint is that joint's parent or child joint, whichever is closest to the point. The third most influential joint is the nearest parent or child of the second joint. The influence drops with the distance from the joints. The amount of influence each joint has on any skin point is the *skin weight*.

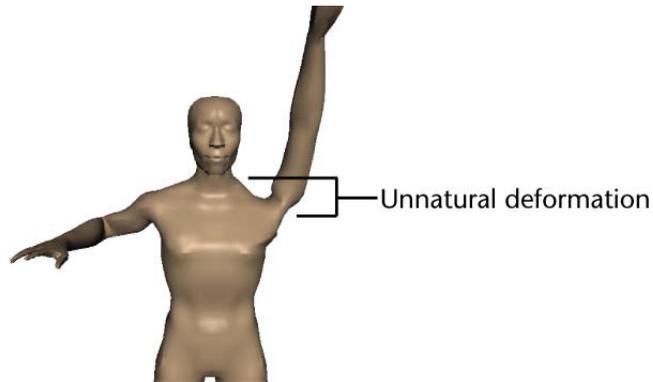
The default skin weights create smooth deformations of the skin at elbows, knees, and elsewhere as the nearby joints rotate.
- 5 Select and move any or all of the four IK handles to pose the arms and legs in various positions in order to get used to posing with the IK handles. To select an IK handle, drag a selection box around a wrist or ankle joint. To move a handle, use the Move tool to drag the handle in the desired direction.

NOTE You can also select the IK handles in the Hypergraph. The handles are indented under jackieSkeleton. Each is named for a leg or arm, for example, ikHandleLeftLeg. For more details on IK handles, see [Lesson 1: Skeletons and kinematics](#) on page 314.

- 6 If you prefer to turn off the skin's transparency so you can see the skin more clearly, turn off Shading > X-Ray.

Do not be concerned that the skeleton pokes through the skin, as the skeleton is not displayed in a rendered image.

As you pose the arms and legs, examine the skin in the regions where joints bend. For many poses, the skin looks natural. For others, the skin folds, compresses, or bulges unnaturally. The pelvis, shoulders, and torso are common problem areas for various poses. For example, if you move an arm straight up, the shoulder compresses as in the following figure.



There are two ways to improve the smooth skin deformations:

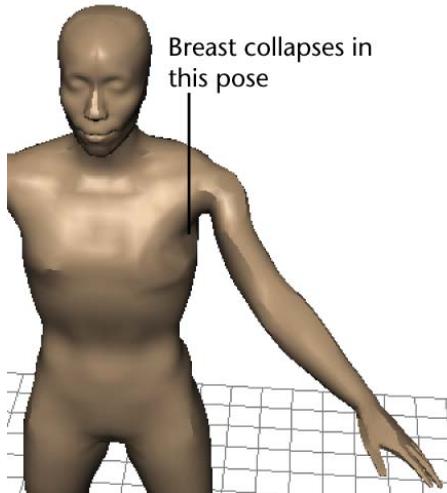
- Add an influence object to eliminate collapsing regions or to create muscle bulge. For example, you can use an influence object to make the shoulder region look more natural in the prior pose.
- Edit skin weights—to remove minor lumps or indented regions. For example, you can fill out an undesirable concave region at the chest that appears when you move the arm to certain poses. The following procedures explain the techniques.

Skin weighting and deformations

In the next steps, you'll learn how joints and skin weights influence deformations at Jackie's chest. In a subsequent section, you'll improve uneven deformations.

To see how skin weights affect a skin's deformation

- 1 Pose the left arm similar to the following figure. The left breast becomes irregularly shaped.



- 2 Select Jackie.
- 3 Select Skin > Edit Smooth Skin > Paint Skin Weights Tool >  .
- 4 In the Influences list of the Tool Settings window, select any joint, for instance, pelvis.

The grayscale color of Jackie's skin indicates how much influence the selected joint has on the skin deformation. In general, a white region of skin is influenced nearly entirely by the selected joint. A gray region is influenced significantly by one or two additional joints, and a black region indicates skin not influenced by the selected joint at all.

TIP If you prefer, you can turn on Use Color Ramp in the Paint Skin Weights Tool to view the influence feedback in color. This can make it easier to see small weights.

Each point on Jackie's surface is influenced by three joints, as specified by the Max Influence setting in a prior step. However, one or two of the three joints might have so little influence as to be insignificant.

In this case, the left breast becoming irregularly shaped as you pose the arm indicates that one or more of the joints is exerting too much or too little influence on the surface. In the next steps, you will determine which joints are influencing the irregular region.

- 5 Do the following to select the vertices in the problem area:
 - In the Paint Skin Weights Tool, turn on Paint Select mode (below the Influences list).
 - Right-click Jackie's skin and select Vertex from the marking menu.
 - Paint to select the vertices around Jackie's shoulder.
- 6 In the Paint Skin Weights Tool, click the Show Influences  button. All joints that influence the selected vertices are selected in the Influences list.

TIP If you don't immediately see the selected influences in the list, scroll to find which influences are selected.

Depending on which vertices you selected, the Influences list shows you that the main joints influencing this region are from the mid and upper back, and the left arm.

- 7 Now that you have narrowed down which joints are influencing the shoulder, select each entry in the Influences list individually to view the shades of gray on Jackie's skin.

The `left_arm_root` is the main influence. The `upper_back`, `mid_back`, and `left_top_rib` also have influence, though not exactly in the same region.

Modifying skin weights

You can modify the skin weights for any of the influencing joints to alter the irregular region. If you don't get the desired results when you change the skin weights of one of the influencing joints, you can undo your changes and try another influencing joint.

In general, it's best to modify the skin weights of the most influential joint first, then work with less influential joints if you don't get the desired results. Modifying skin weights requires experimentation.

In the next steps, you'll modify the skin weights for the `left_arm_root` joint.

To modify skin weights using the Paint Skin Weights Tool

- 1 Select `left_arm_root` in the Influences list.
- 2 Set the Mode to Paint, then set the Paint Operation to Add.

3 Set the Value to 0.1.

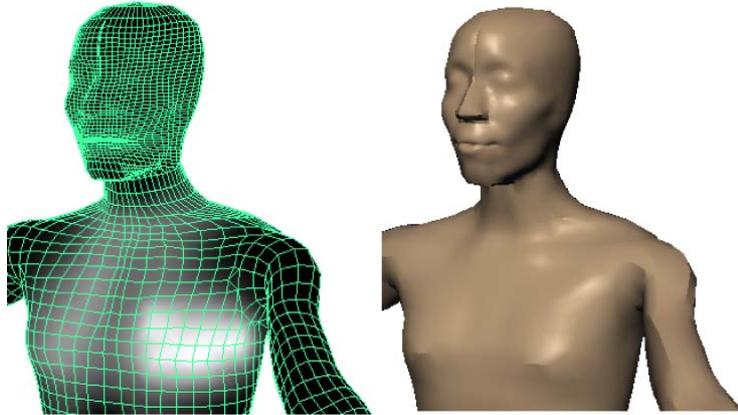
4 Drag to paint on the irregular region of the breast.

As you stroke a region, a red circle icon indicates the area affected by the brush stroke. Each stroke adds 0.1 (10%) to the skin weight. The weight for a skin point has a maximum value of 1 (fully white).

TIP To change the size of the red circle icon, change the Radius(U) value in the Paint Skin Weights Tool.

Repeat the strokes several times until the irregular region becomes smoother. The jagged, asymmetrical wireframe at the breast becomes more symmetrical as the skin becomes smoother. The region whitens, which indicates increased influence from the left_arm_root joint. Note that increasing the influence of one joint lessens the influence of the other influencing joints.

5 To check the shape with full-color shading, click the Select Tool  in the Maya Toolbox. (You might also prefer to cancel the selection of Jackie to eliminate the display of the highlighted wireframe.)



6 To continue modifying the skin weights, select Jackie and select Skin >

Edit Smooth Skin > Paint Skin Weights Tool, or click  in the Maya Toolbox.

There are several additional tool settings you might find useful as you modify skin weights:

- The Smooth paint operation smooths the weights of points by averaging the stroked points with the weights of the surrounding region. This is useful if you add to the weight values in some region but the points become unevenly weighted. Uneven weights are indicated by mottled gray skin.
 - The Weight Hammer button  lets you select vertices in a region with undesirable deformation then click to quickly flatten the area. This tool checks the weight values of neighboring vertices and applies the same weight value to the selected vertices.
 - You can use the Scale operation with a Value less than 1 to scale down the point weights of a stroked region.
 - You can select any of the Profile icons to specify the region affected by strokes. The two leftmost icons are the most commonly used shapes.
- 7 After you eliminate the irregular region from the left breast, move the arm to different positions to see if the breast's shape stays natural in various poses.
- It is common for a surface to look good in one pose but not in another. Strive to make the surface look good in the poses that you are likely to use during animation.
- 8 If you want more practice, pose the right arm as you posed the left arm and fix the corresponding irregular region in the right breast. This time, fix the region by modifying the weights for the upper_back joint rather than the right_arm_root. As mentioned previously, you can modify point weights of alternative joints to get a similar effect.

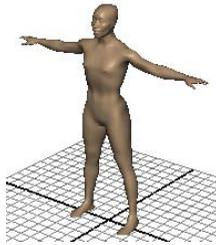
Influence objects

You can create an object and use it to influence the shape of smooth skin. The object, called an influence object, acts like a surgical implant against which the skin deforms. For example, you can create a sphere and use it to simulate a muscle or bone that bulges as you pose the character in certain positions. You can also use an influence object to smooth deformation and maintain volume in regions that collapse while bending.

In the next steps, you'll use a sphere as an influence object to simulate an elbow jutting out as Jackie's arm bends. To use an influence object, the skeleton must be in the bind pose—the original pose at which the skin was bound to Jackie.

To return the skeleton to the bind pose

- 1 Click the Select Tool and cancel the selection of Jackie.
- 2 Turn off Modify > Evaluate Nodes > IK Solvers.
After you use an IK handle to pose a character, you must do this step before you can return the skeleton to the bind pose.
- 3 Select `back_root`, the root of the skeleton.

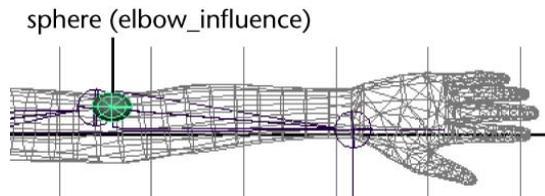


- 4 Select Skin > Go To Bind Pose.
- 5 Select the `ikHandleLeftArm` in the Hypergraph. Use the Move Tool to move the handle manipulator slightly in any direction. When you release the mouse, note that the arm does not move and that the IK handle manipulator snaps back to the desired position at the end of the joint chain. Repeat this step for any other IK handles you posed previously.
- 6 Turn on Modify > Evaluate Nodes > IK Solvers.
You must do this step to enable the IK solvers so you can manipulate IK handles in the scene again.

To position a sphere to be used as an influence object

- 1 Right-click Jackie's surface (anywhere but on the skeleton) and select Actions > Template from the marking menu.
By making Jackie a template, you won't accidentally select Jackie in subsequent steps.
- 2 Create a sphere (select Create > Polygon Primitives > Sphere >) and name it `elbow_influence`.

- 3 Scale the sphere to be slightly smaller than any of Jackie's joints (see illustration below).
- 4 Move the sphere so that its surface is positioned where the elbow would jut out, but still within the skin. You can optionally reshape the sphere's curvature to resemble the tip of an elbow jutting out from a bent arm. For example, you can scale the sphere in one dimension to squash its shape or you can move individual vertices. A top view of the sphere's shape and position follows:



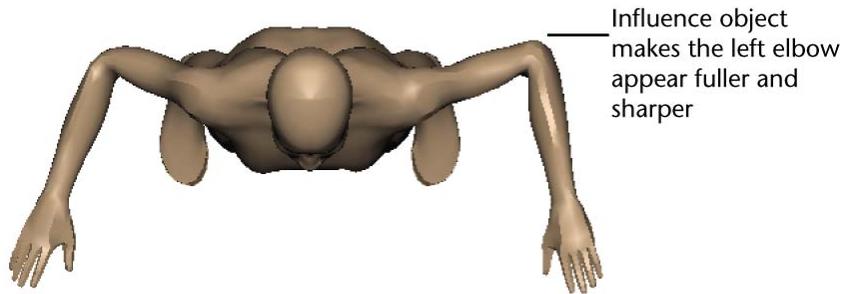
The exact position and scale of the sphere is unimportant. You can make slight position and scale adjustments later to enhance the deformations that result from its use.

To make the sphere an influence object

- 1 In the Hypergraph, select Jackie and Shift-select elbow_influence.
- 2 Select Skin > Edit Smooth Skin > Add Influence > . In the Add Influence Options, select Edit > Reset Settings, and then click the Add button.
Maya completes the operation within a few moments. To ensure the influence object stays in the correct position at the elbow, you will parent it to the left_shoulder joint.
- 3 In the Hypergraph, Shift-select elbow_influence, elbow_influenceBase, and left_shoulder joint, in that order.
The order of selection is important. The parent object should be selected last.
- 4 Press p to parent the elbow nodes under the left_shoulder joint.
The elbow_influence nodes appear in the skeleton hierarchy below the left_shoulder joint.
The elbow_influenceBase is known as a base object. It stores vertex position information for the influence object. The reason for parenting the base object to the left_shoulder joint is so that the influence object's vertices stay in the correct position at the elbow whenever the elbow

moves to a new position. Other than this step, you do not work with the base object.

- 5 Right-click Jackie and select Actions > Untemplate. It is easier to see subsequent skin deformations when Jackie is displayed with smooth shading rather than as a template.
- 6 From the menus at the top of the top view, turn off Show > Joints to avoid having the joints obstruct your view of the skin.
- 7 Use each arm's IK handle (ikHandle1 and ikHandle2) to pose the arms in the following position.



The left elbow looks more natural than the right elbow. To tune the elbow's deformation during bending, change the position, scale, and rotation of `elbow_influence`.

Do not be concerned if the influence object pokes through the skin. It won't be displayed when you render the animation.

Beyond the lesson

In this lesson you learned how to:

- Bind a skeleton to a single surface.
You can bind a skeleton to multiple surfaces or even to a selection of polygonal vertices or NURBS CVs or subdivision surfaces.
- Use the Paint Skin Weights Tool to prevent a region from collapsing upon deformation.
To control skin weights with more precision than shown in the lesson, you can modify skin weights numerically with the Component Editor. If you smooth skin multiple surfaces which have been seamed together, there

are many tool settings for the Paint Skin Weights Tool that make the task easier.

- Use an influence object to make an elbow stick out upon bending. Another common use of an influence object is to simulate a muscle bulging during joint rotation. The technique requires use of Set Driven Key to link the bulging influence object's scale values to the rotation of the appropriate joint. A brief example of this technique is provided in the Maya Help for Smooth Skin.

Skin weights and influence objects cannot overcome all modeling problems. For instance, if you further separate Jackie's legs sideways, you will see an undesirable fold at the hips. This occurs because of the asymmetrical arrangement of polygonal edges in the hip region of the original model. To fix the problem, you would need to detach the skin, alter the original model, then smooth bind the model again.

Maya has an alternative skinning method, rigid skinning, which gives results similar to smooth skinning but requires use of different enhancement tools: flexor and lattice deformers. In general, smooth skinning gives more natural deformations with less effort than rigid skinning. If you skin multiple seamed surfaces, however, processing is faster with rigid skinning than with smooth skinning.

Smooth skinning is just one of the techniques for perfecting a character's deformations during animation. You can use Maya's other deformers alone or in addition to smooth skinning to achieve the results you want.

For more information refer to Smooth skinning in the *Character Setup* guide.

Lesson 3: Cluster and blend shape deformers

Introduction

Facial animation is an integral component of character animation. The face of a character can be animated to impart a range of emotions and expressions. Maya has two deformation tools that ease character setup for facial animation: cluster deformers and blend shape.

Cluster deformers enable you to control a set of an object's points (CVs, vertices, or lattice points) with varying amounts of influence to create a target shape for an animation.

A Blend Shape deformer provides an interface for blending between various target shapes, to control the range of movement for the cluster on an object or face.

In this lesson, you will be introduced to cluster deformers in order to shape a character's mouth into a smile pose. You will learn to use the Blend Shape feature so you can animate the face from a neutral pose to a smile. In this lesson, you learn how to:

- Select CVs in a region and create a cluster deformer.
- Use the cluster deformer handle to adjust the position of a cluster deformer.
- Use the Paint Cluster Weights Tool to refine the cluster weight values.
- Create a Blend Shape deformer to control the blending between target shapes.

Open the scene for the lesson

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

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 314.
- 2 Open the scene file named `ClusterBS.mb`.

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

`GettingStarted/CharSetup/ClusterBS.mb`



Model by Cory Mogk

- 3 Select Shading > Smooth Shade All and then Shading > Hardware Texturing, and Lighting > Use All Lights to view the head fully textured (Hotkey 7).

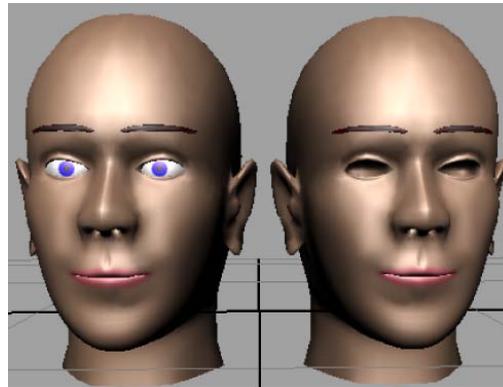
Creating a target object for a blend shape

In the next steps, you duplicate the face of the head to create a target object for the blend shape. Later in the lesson, you apply a cluster to the duplicated face (target object) so you can create a smile. The facial expression on the target object is what the original base object will match.

- 1 Using the Outliner, select only the face. That is, select only the head surface without the eyes.

In the Outliner, the face is named baseFace and is indented under head.

- 2 Select Edit > Duplicate Special > . In the options window, select Edit > Reset Settings, turn on World for the Group Under option, then click the Duplicate Special button. By grouping the duplicate under the world, you create an object independent of the original.
- 3 Move the duplicate face to a position where both faces are visible for comparison, for instance, to the right side of head.



Head

Duplicated face

- 4 Rename the duplicate face as smilingFace.
This duplicate face will be modified so it can be used to influence the original.

Creating a cluster deformer on a target object

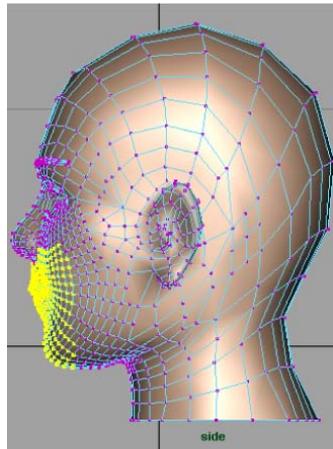
In the next steps, you create a cluster deformer on the duplicated face (target object) to reshape the expression into a smile. The cluster deformer on the target object will be used to change the unsmiling baseFace (base object) into a smiling face.

A cluster deformer creates a *set* whose members consist of selected points (CVs, vertices, or lattice points). You assign a percentage weight to each point, indicating how much you want each point to be affected by any translation, rotation, or scale of the cluster set. When you transform the cluster, the points are transformed according to the percentages you have specified. A cluster is useful for stretching, moving, or compressing part or all of a surface.

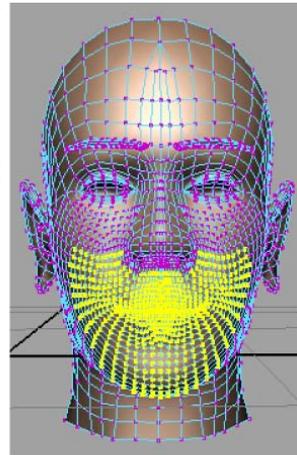
To create a cluster deformer

- 1 Position the mouse pointer over the face, then right-click and select Vertex from the marking menu.

Next, you'll select the smilingFace vertices as shown in the illustration, which are roughly the vertices at the chin, cheeks, and lips, but not the nose.



Side view

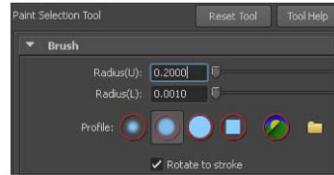


Persp view

- 2 An easy way to select the vertices is to use the Paint Selection Tool. Open the Paint Selection Tool by double-clicking the icon in the Toolbox to display the Paint Selection Tool settings.



Paint Selection Tool

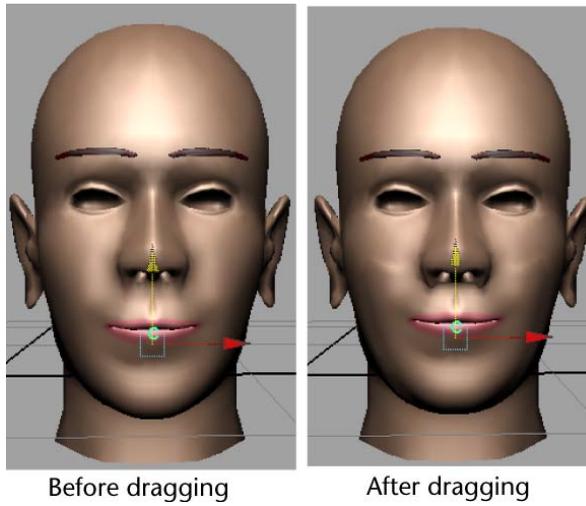


Paint Selection Options

- 3 In the settings window, set Radius(U) to 0.2. Drag the mouse over the desired vertices to select them. Ctrl-drag (Windows and Linux) or Control-drag (Mac OS X) to cancel the selection of unwanted vertices. The region of selected vertices does not need to be exact, although you want to avoid selecting vertices at the back of the head inadvertently. Also make sure you don't inadvertently miss a few vertices in the region you do want to select.

The objective is to select all vertices where a smile might deform the face. You might want to look in a mirror to see which parts of a face move while smiling. It's better to select too many vertices than too few. It's easier to work with too many vertices than too few.
- 4 Close the Tool Settings window when you are finished.
- 5 Select Create Deformers > Cluster.

This puts the vertices into a cluster—a set of points you can move as single entity. The cluster's handle appears in the view as a C icon.
- 6 In the Outliner, make sure the cluster2Handle is selected.
- 7 Use the Move tool to drag the cluster handle up along its Y-axis a small amount until smilingFace is deformed as follows:



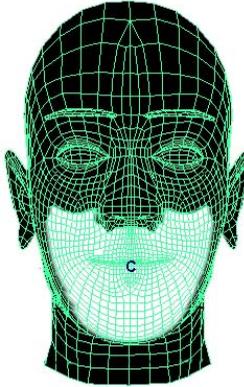
The cluster set on smilingFace should be slightly higher than the cluster on the original face. This will become the basis for the blend shape deformer.

Editing cluster weights

In this section you will use the Paint Cluster Weights Tool to further refine the smilingFace. As you saw in previous steps, the points in a cluster move when you move the cluster handle. The distance a particular point moves depends on its *weight*. Lower weights cause less movement, while higher weights cause exaggerated movement. In the next steps, you create a smile by editing the weights.

To edit cluster weights using the Paint Cluster Weights Tool

- 1 Select smilingFace.
- 2 Select Edit Deformers > Paint Cluster Weights Tool > . This displays the smilingFace in grayscale.



The white area shows the points that make up the cluster. The whiteness also indicates the weights—how much the points move in response to the movement of the cluster handle. White indicates a cluster weight of 1—the points move the same distance as the handle. By default, each point has a weight of 1.

Black indicates a cluster weight of 0. The points do not move in response to cluster handle movement. Note that regions that are not part of the cluster are also black.

Although there are no gray regions currently, gray means the movement is partial. The lighter the gray, the more the movement. Gray regions indicate a cluster weight between 0 and 1.

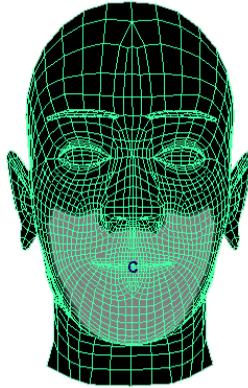
3 In the Paint Attributes section of the Tool Settings window, set the following:

■ Paint Operation: Replace

■ Value: 0.5

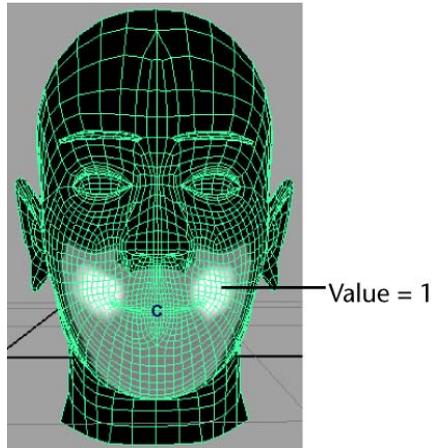
4 Click Flood.

This gives all the cluster points a weight of 0.5. The white region is now gray:



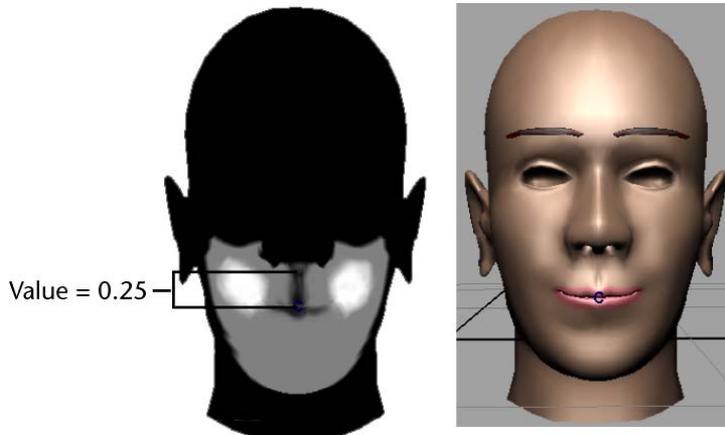
The selected vertices move down a bit, as the weight of 0.5 lessens the effect of the prior movement of the cluster by 50%.

- 5 To check the shape with full-color shading, click the Select Tool. (You might also prefer to cancel the selection of smilingFace to eliminate the display of the highlighted wireframe.)
- 6 To return to the Paint Cluster Weights Tool, select smilingFace and Edit Deformers > Paint Cluster Weights Tool > again.
- 7 In the Tool Settings window, set the following:
 - Radius(U): 0.1
A Radius(U) of 0.1 sets the radius of the region affected by the stroke. The red circle icon on the face displays the radius.
 - Opacity: 0.3
An Opacity of 0.3 scales down the effect of painting weight values so you can build up values gradually with repeated strokes.
 - Paint Operation: Replace
A Replace operation paints the Value specified on the points stroked.
 - Value: 1
A Value of 1 sets the weight value you paint on the points.
- 8 Paint the region at the corners of the mouth with repeated strokes until the face looks roughly like the following figure. If your results differ from the figure significantly, undo the strokes and try again.



While painting weights, it's useful to intermittently check the shape with full-color shading rather than with the grayscale shading. To do this, click the Select Tool to turn off the Paint Cluster Weights Tool. (You might also prefer to cancel the selection of smilingFace to eliminate the display of the highlighted wireframe.)

- 9 Enter a Value of 0.25 and paint the central region above and below the lips as follows:



Because the corners of the mouth have a high weight and the central region around the lips have a low weight, the corners move up more in response to the prior positioning of the cluster handle. This creates a smiling mouth. The cheeks and chin have a medium weight, so they also

move up slightly also. (For many people, skin at the chin and neck moves during smiling.)

- 10 In the tool settings editor for the Paint Cluster Weights Tool, set the Paint Operation to Smooth. Paint any region where the surface has become irregular. Irregular regions are typically indicated by jagged wireframe curves (isoparms) or where a grayscale color makes an abrupt change to a lighter or darker color. Smoothing averages the weights of the stroked points with the weights of the surrounding region. The Value setting has no effect on smoothing.
- 11 Replace and smooth weights in other regions of the face until you are satisfied with the smile. For example, you might want to reduce the weight values at the side of the nose and immediately below the nose.
Your smilingFace should now have a slight smile as a result of your cluster weight edits.
- 12 Hide the Paint Attributes Tool settings window.

Creating a blend shape

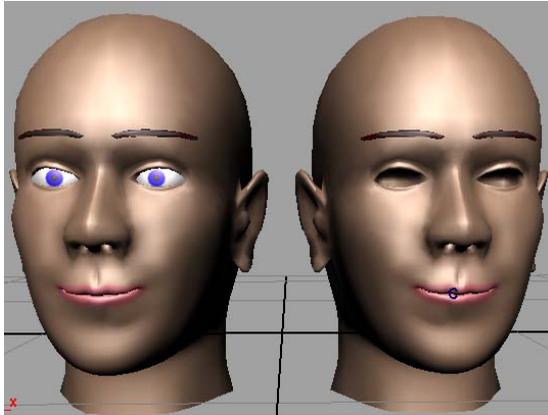
In this section, you create a blend shape deformer to control the smile on baseFace (base object) using the smilingFace (target object). A blend shape deformer is ideal for facial animation, where you need a number of facial positions to be readily available for use in an animation sequence. With a blend shape deformer, you can set up a character's face to blend between a smile, frown, smirk, and so on.

In the next steps, you create a blend shape deformer to change the baseFace into the smile of smilingFace.

To create a blend shape deformer

- 1 In the Outliner, select smilingFace and Shift-select baseFace, specifically in this order.
The smilingFace (the first selection) is called the target object, while baseFace (the second selection) is called the base object. The objective of a blend shape operation is to morph the base shape into the target shape.
- 2 Select Create Deformers > Blend Shape >  .
- 3 In the Blend Shape node text box, type `blendShape`, then click Create.

- 4 Select Window > Animation Editors > Blend Shape. This displays the Blend Shape editor, which has a slider for changing the base into the target shape, and buttons for setting keys.
- 5 Cancel the selection of the faces so you can see the surfaces without the obscuring highlighted wireframe.
- 6 Drag the slider from 0 to 1 to morph the baseFace into the target, smilingFace.



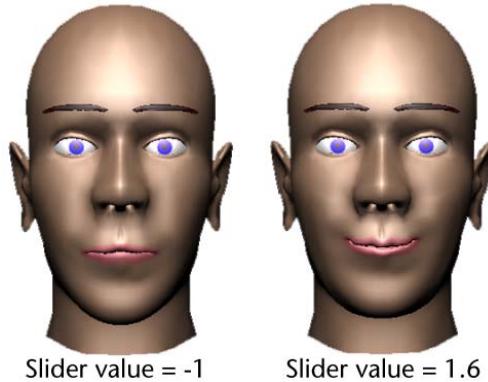
Note that you can click the Key button below the slider to set an animation key for the shape of the face at the current frame. By setting a few keys with different slider values at different points in the timeline, you can animate from the neutral expression to a full or partial smile or vice versa.

When you key the shape, Maya applies the key to the blendShape node that was created when you selected Create Deformers > Blend Shape > . If you need to select the blendShape node, for instance, so you can see or delete the keys in the Time Slider, click the Select button in the Blend Shape editor.

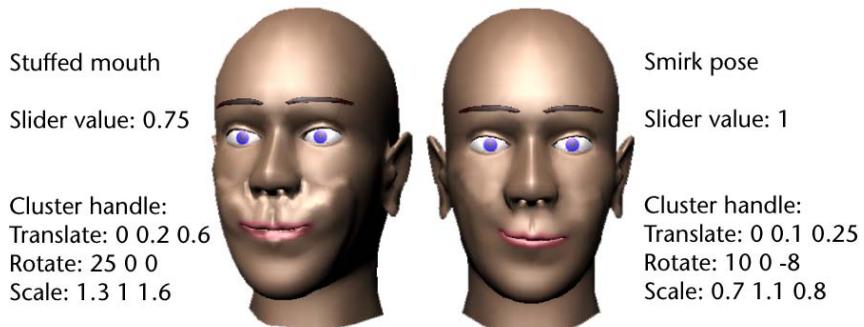
Refining deformation effects

The combination of a cluster and blend shape is ideal for facial animation because it lets you tune various subtle deformations. A few common techniques follow.

- In the box below the slider of the Blend Shape editor, you can enter a numerical value below 0 to invert the deformations, or above 1 to amplify the deformations. For example, -1 creates a frown, while 1.6 creates a brimming smile:



- With or without changing the Blend Shape slider, you can move, rotate, and scale the cluster handle to modify the blend shape. The following figure shows some examples. The face you created will be slightly different because your original smile is different.



- You can use the Paint Cluster Weights tool again to change cluster weights when the cluster and blend shape editing deforms certain regions

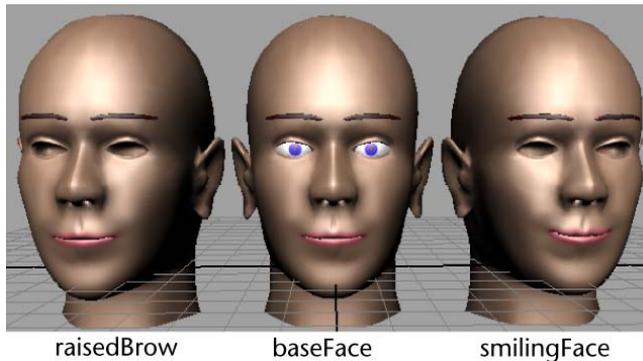
undesirably or doesn't give the exact look you desire. You can also use Ctrl - Z (Windows and Linux) or Control-z (Mac OS X) to undo any undesired changes.

Adding target objects to an existing blend shape

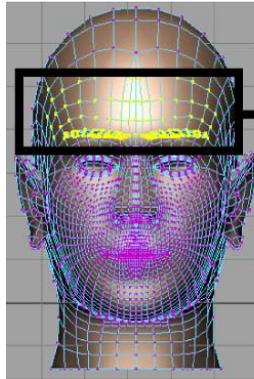
In the next steps, you duplicate the face again and reshape the duplicate into a new facial expression. You then add the new face to the blend shape node to create another slider in the Blend Shape Editor.

To create another facial expression

- 1 Set the Blend Shape editor slider value to 0 to return baseFace to the position it had at the beginning of the lesson.
- 2 Make a duplicate of baseFace, name it raisedBrow, and move it to the left of baseFace so that all three faces are visible for comparison.



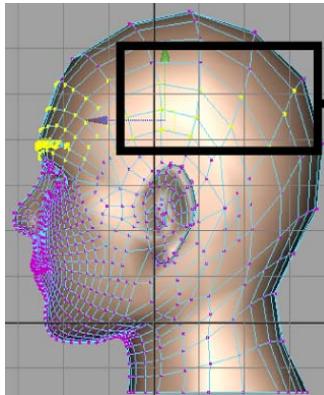
- 3 Make sure raisedBrow is selected.
- 4 In the perspective view, select Show > Isolate Select > View Selected. Select this menu item again in the front view.
This displays only the selected object (raisedBrow) in the views. This is necessary to avoid selecting unwanted vertices in the next steps. The view's label indicates isolate is turned on.
- 5 In the front view, position the pointer over the face, right-click and select Vertex. Drag a selection box around the vertices in the region of the eyebrows and forehead as shown below.



Drag a selection box around these vertices

For these vertices, you need to select the vertices by dragging a selection box rather than by using the Paint Selection Tool. The eyebrows have vertices that lie behind its outer surface. The Paint Selection Tool selects only vertices at the outer surface. Dragging a selection box selects all vertices in the boxed region, including vertices that lie behind the outer surface.

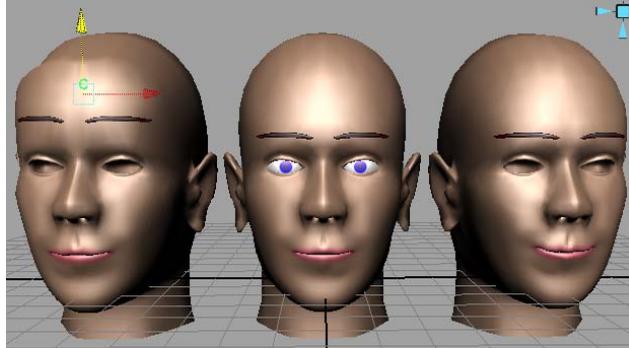
- 6 In the side view, hold down the Ctrl (Windows and Linux) or Control (Mac OS X) key and drag a selection box around the vertices at the side of the head to turn *off* their selection (see the following figure).



Hold down the Ctrl or Control key and drag a selection box around these vertices to cancel their selection.

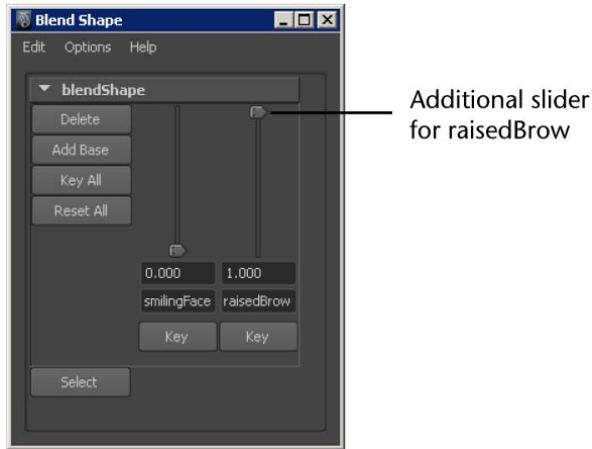
- 7 Switch to wireframe display mode and dolly the camera from various close-up views to make sure you select all the vertices on and under the eyebrow. If you miss a few vertices, subsequent deformations will not work correctly. Also, make sure you don't select vertices at the side or back of the head.

- 8 Select Create Deformers > Cluster.
- 9 In the perspective view, turn off the selection of Show > Isolate Select > View Selected. Do this again in the front view.
By turning off these menu selections, Maya displays all objects in the views again.
- 10 In the perspective view, drag the cluster up along its Y-axis a small amount until raisedBrow looks like the face on the left:

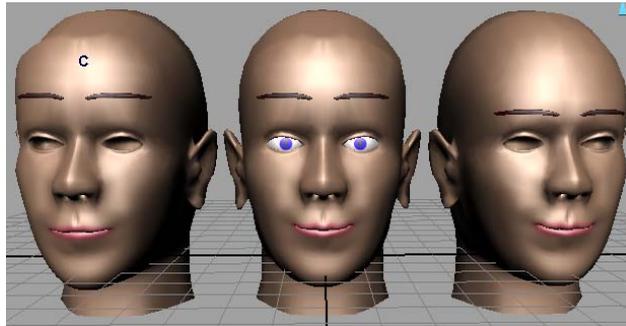


To add the new target object to the blend shape

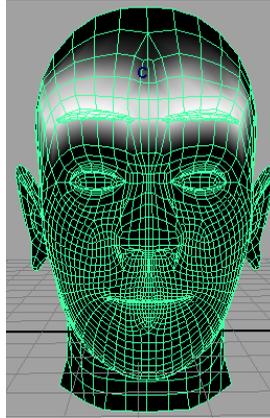
- 1 In the Outliner, select raisedBrow and Shift-select baseFace (the order of selection is important). The raisedBrow is the target, while baseFace is the base object.
- 2 Select Edit Deformers > Blend Shape > Add > . In the options window, turn on Specify Node and enter blendShape in the BlendShape Node box. Click the Apply and Close button.
When you created the blend shape for the smile, Maya created a node named blendShape that contains the slider attributes that adjust the blend into the smilingFace target. The Add operation creates a blend shape for the raisedBrow and adds it to the blendShape node. This adds a slider to the node's Blend Shape editor for adjusting the raised brow deformations.
- 3 To display the Blend Shape editor, select Window > Animation Editors > Blend Shape.



- 4 You can use the sliders alone or in combination to create a smile with raised eyebrows, a frown with lowered eyebrows, and so on.



- 5 You can optionally select raisedBrow and edit the cluster weights to tune the deformation of the eyebrow region as desired. See [Editing cluster weights](#) on page 346 for details. An example weighting follows:



TIP After you create a blend shape, you can optionally hide or delete a target object (in this lesson, `smilingFace` and `raisedBrow`). If you delete a target, you improve Maya processing time but lose the capability to manipulate the cluster handle. For versatility, many animators hide the target rather than delete it. Hiding the target is necessary when you render the scene. It is also useful when you want to unclutter the scene view.

Beyond the lesson

In this lesson, you learned how to:

- Use two clusters and a blend shape deformer to manipulate facial expressions.

In professional productions, it's common to create several more clusters with a blend shape deformer for finer control of facial expressions. For example, you can create clusters for a number of mouth positions that you can use to simulate the phonemes of speech.

When you used certain blend shape slider values, you may have noticed that the face looked good in all regions except for a few vertices. To fix such regions, delete the cluster, move the problem vertices of the target face to a better position, then create a blend shape for the improved target face.

As much as possible, use a target shape that's not extremely different from the base shape. If you need to use an extremely different target shape, create a blend shape with multiple targets. Each of the multiple targets must have a shape that is progressively more like the extreme target shape.

Before you create the blend shape, turn on In-Between in the Create Blend Shape options window, and select the multiple targets in this order: least extreme shape difference first, most extreme shape difference last.

Although the lesson showed how to blend individual objects, you can also blend hierarchies of objects. See the Maya Help for more information.

There are some additional notes on clusters which are not explained in this lesson:

- You can add or remove vertices from a cluster with Edit Deformers > Paint Set Membership Tool.
- You can use the Component Editor to edit weight values more precisely than by painting.
- Avoid changing the number of vertices (or CVs) for a surface after you apply a cluster or other deformer. Unexpected deformations may occur.

Polygon Texturing

8

Introduction



Texture maps let you modify the appearance of your 3D models and scenes in Maya. *Texture maps* are images you apply and accurately position onto your surfaces using a process called *texture mapping*. When an image is texture mapped onto a surface, it alters the appearance of the surface in some unique way. Texture maps let you create many interesting visual effects:

- You can apply labels and logos to your surfaces.
- You can apply surface relief details and features to a surface instead of having to model the details on the surface directly.
- You can use illustrations as texture maps to create interesting backdrops in your scenes.

Most shading attributes for a surface material can be altered by a texture map. For example, color, specular, transparency, and reflectivity are examples of attributes that can be modified by a texture map.

Texture mapping is a key component in the 3D production workflow. Many production environments employ texture artists whose only role is to create and apply the texture maps to 3D models.

Preparing for the lesson

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

- 1 Select File > New Scene to create a new scene.
- 2 Make sure the Construction History icon (below the menu bar) is on:
 (If it is turned off, it has a large X across it.)
- 3 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](#) on page 25.
- 4 Select the Polygons menu set. Unless otherwise noted, the directions in this chapter for making menu selections assume you've already selected the Polygons menu set.
- 5 Make sure Display > UI Elements > Help Line is turned on. You will use the Help Line while modeling.
- 6 Ensure that the interactive creation option for primitives is turned off by selecting Create > Polygon Primitives > Interactive Creation. The option is off when a check mark does not display beside the item's name in the menu.
- 7 Make sure that Soft Selection is turned off by opening the Attribute Editor with the Select Tool active and unchecking the box marked Soft Select.

modifying the UV texture coordinates so that the texture map appears correctly on the surface.

In this lesson you learn how to:

- Assign a 2D texture map to a polygonal model.
- Map UV texture coordinates (UVs) to a polygonal surface.
- Correlate the UVs between the scene view and the UV Texture Editor.
- Use the UV Texture Editor to visualize how the UV texture coordinates from a three-dimensional model relate to an assigned two-dimensional texture map.
- Determine basic UV layout requirements.
- Set preferences to let you visualize the texture borders in both the UV Texture Editor and the scene view to better understand how the texture map is placed on the polygonal model.
- Select and reposition UV texture coordinates within the UV Texture Editor using the transformation tools to make the UVs match a pre-defined texture map.
- Sew UV texture coordinates to existing UV shells.

Creating a cracker box model

You begin the lesson by creating a model of a cracker box using a polygonal cube primitive. The dimensions you'll use for the cracker box are based on a texture map you'll apply to the model in this lesson.

To create a cracker box model using a cube primitive

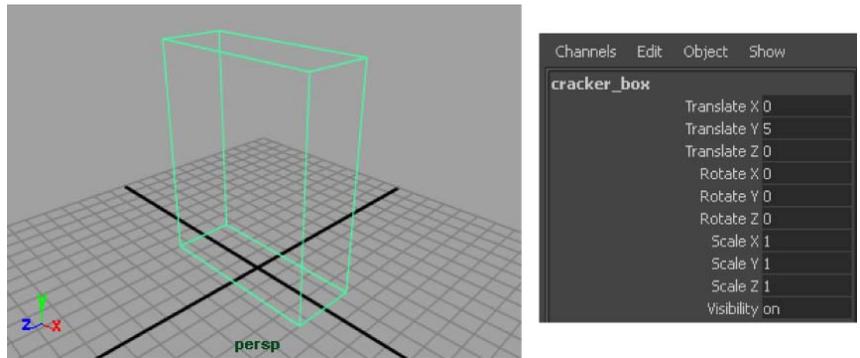
- 1 Select Create > Polygon Primitives > Cube > .
- 2 In the Polygon Cube Options window, select Edit > Reset Settings, set the following options, and then click Create:
 - Width: 8
 - Height: 10
 - Depth: 3
 - Width divisions: 1

- Height divisions: 1
- Depth divisions: 1
- Axis: Y
- Create UVs: On
- Normalize: Off

A cube primitive in the shape of a rectangular box appears in the scene view.

When the Create UVs option is on it ensures that the primitive object is created with a set of default UV texture coordinates. All polygon primitives in Maya provide an option for creating UV texture coordinates at the time of creation.

- 3 Using the Channel Box, set the Translate Y attribute to 5 so that the cracker box is repositioned to rest on the X axis.
- 4 Rename the pCube1 primitive to: `cracker_box`.



Applying a texture map to a polygon mesh

In Maya, surfaces appear in a rendered image based on the shading materials you apply to them. The default shading material that gets assigned to surfaces provides basic attributes such as color, transparency, and incandescence.

Depending on your final image requirements, you can enhance the visual impact of an object by adding one or more texture maps to the assigned shading material. One basic texture map you can apply is a bitmap image also

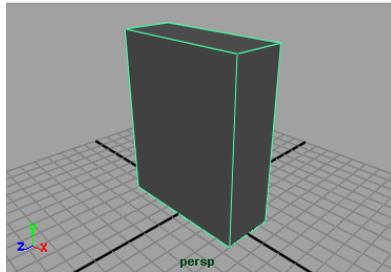
referred to as a *file texture*. In this lesson, you apply a texture map we've created for your use on the cracker box.

For more basic information on shading materials and texture maps see the *Rendering* chapter of this guide and the Maya Help.

To assign a shading material to the cracker box

- 1 In the scene view, select Shading > Smooth Shade All from the panel menu.

This displays the cracker box with a default gray, smooth shading material, lit using default lighting. Turning on Smooth Shade All displays the assigned shading material you apply to the cracker box in the steps following.

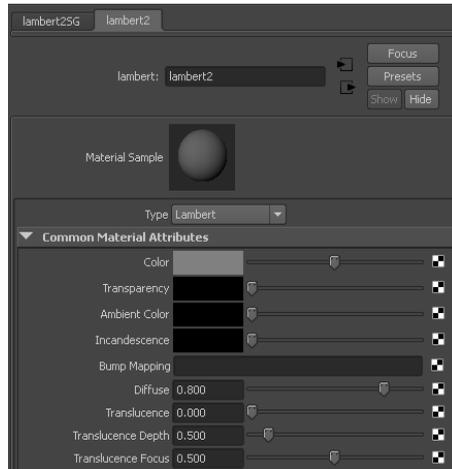


- 2 In the scene view, right-click on any region of the cracker box and select Assign New Material from the pop-up menu.

The Assign New Material window appears.

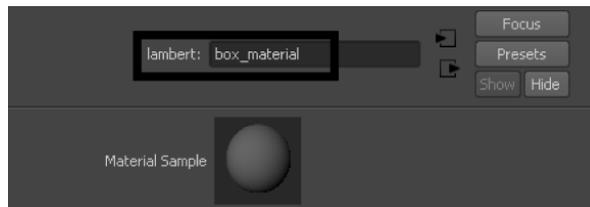
- 3 In the list of materials, select Lambert.

A new Lambert shading material is created and assigned to the cracker box. The Attribute Editor appears and displays the various attributes for the new Lambert shading material.



The new Lambert material is identical to the default gray material that was previously assigned to the cracker box. However, it's always a good practice not to modify the default shading material in the scene. Instead, you should always create a new shading material and modify it to suit your requirements.

- 4 In Attribute Editor, double-click in the *lambert2* box and then type: `box_material` to rename the shading material.



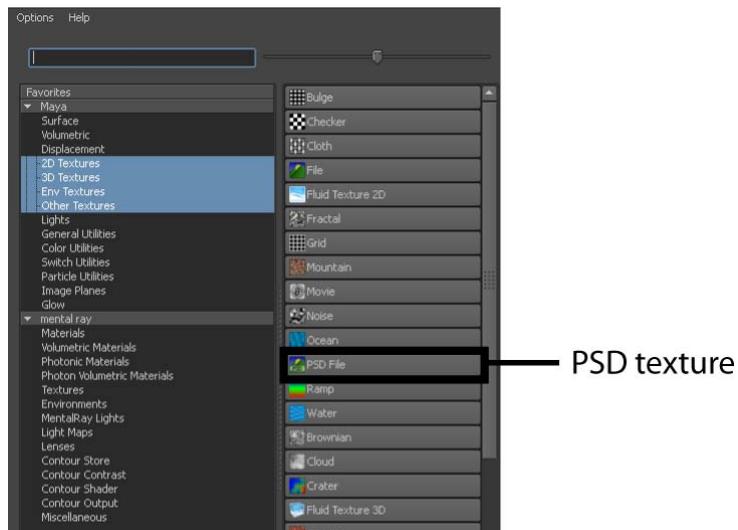
Renaming the shading material lets you easily identify it later on when you need to edit it. Naming items uniquely in Maya is a useful habit.

- 5 In the Attribute Editor, move the *Color* slider fully to the right so that the color box appears white.



The shading material assigned to the box model also updates in the scene view so it appears in a brighter tone.

- 6 Click the Map button located to the right of the Color slider.
The Create Render Node window appears and lists the various texture options you can apply (or connect) to the color channel of the box material.
- 7 Click PSD File from the list of 2D Textures.

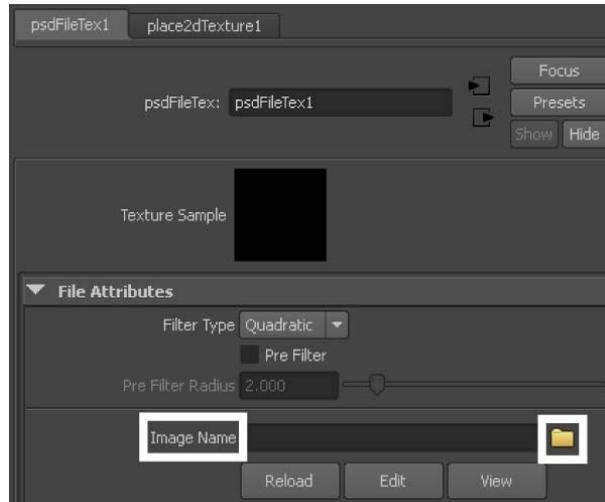


Choosing this option specifies that you want to apply an Adobe® Photoshop® format file as a texture. A PSD format file lets you keep the various components of your texture map on multiple layers. This is useful when you need the ability to edit a component of the texture during production.

The Attribute Editor updates to display the options for the PSD File texture.

TIP If the Attribute Editor suddenly appears empty, it indicates that no objects are selected in the scene. Simply select the box model again in the scene to display the attributes for the box.

- 8 Click the Browse button (folder icon) to the right of the Image Name attribute to specify the image file for the texture map.



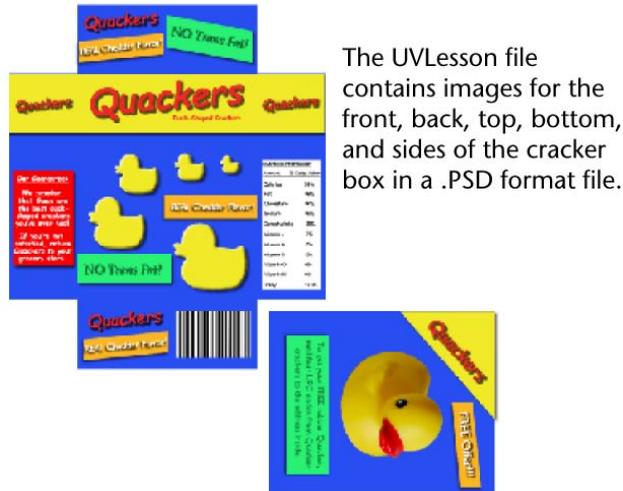
A browser window appears with the path set to the default project directory. By default, Maya looks for source images for texture maps in a `sourceimages` folder whenever the project folder is set. Maya can use file texture images from anywhere on your workstation or local network. For example, you can have images on a central disk drive that is shared among users in a production environment.

- 9 Select the image file named `UVLesson.psd`.

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

`GettingStarted/UVMapping/sourceimages`

The image contains 2D artwork for the front, back, top, bottom, and sides of the cracker box that was generated in Adobe® Photoshop®.

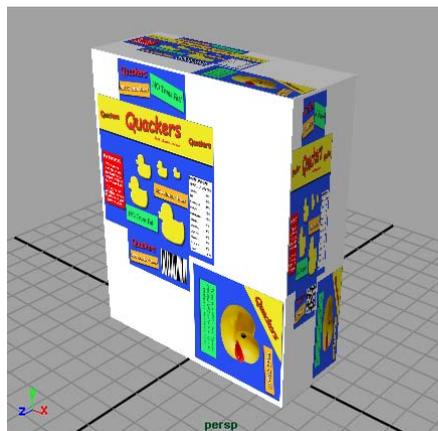


The UVLesson file contains images for the front, back, top, bottom, and sides of the cracker box in a .PSD format file.

A small preview of the image appears in the Texture Sample box in the Attribute Editor.

When a texture is applied to a shading material its attributes get added to the existing attributes for shading material. That is, all of the existing shading attributes remain unchanged except for the attribute that gets modified by the texture. In this case, the grey color is substituted by the image of the cracker box artwork.

- 10 In the scene view, select Shading > Hardware Texturing from the panel menu to display the texture map on the cracker box model.



The texture map does not appear correct on the box because the default UVs for the cube primitive don't correlate to the texture map that was supplied.

The texture map does not appear correctly on the cracker box model. The texture map was designed so that specific sides of the box receive specific regions of the image. Instead, the complete image is being placed on every side of the object.

To correct this problem you must modify the default UV texture coordinates for the model so they match the layout of the texture image. To do this, you use the UV Texture Editor.

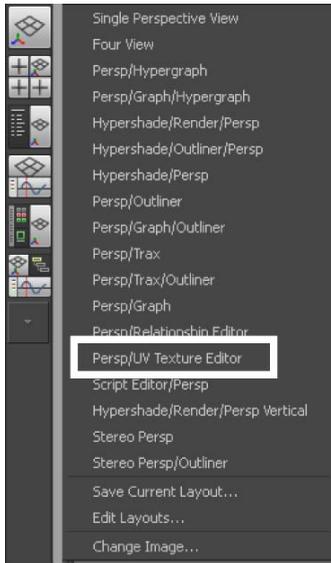
Viewing UVs in the UV Texture Editor

The UV Texture Editor lets you view and interactively edit the UV texture coordinates for polygon and subdivision surface types. UVs appear laid flat within the UV Texture Editor's 2D view. It also lets you display the 2D image for the texture map in relation to the UVs. These features are critical for accurate and efficient placement of texture maps on polygon and subdivision surface types.

In this lesson, your goal is to ensure that the shape and placement of the UVs match the image as it appears in the 2D view of the UV Texture Editor. This will ensure that the map appears correctly on the cracker box model in the 3D scene view.

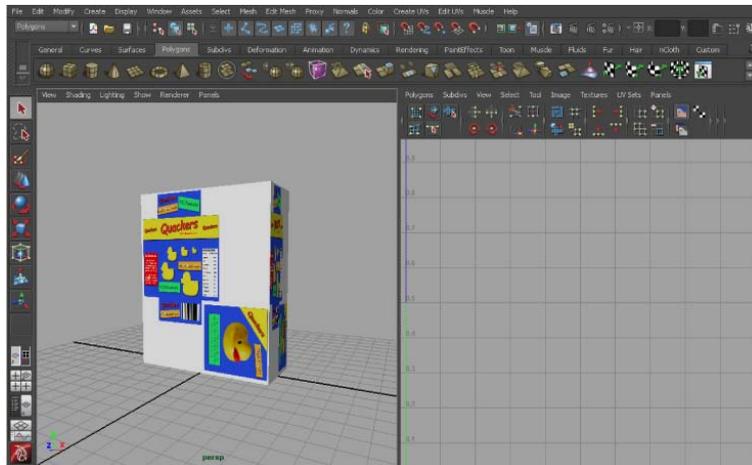
To open the UV Texture Editor in a two pane layout

- In the scene view, right-click any of the Quick Layout buttons on the Toolbox to display the pop-up menu of Quick Layout shortcuts and select *Persp/UV Texture Editor* from the list.



Right-click any of the Quick Layout buttons to view the pop-up menu and select Persp/UV Texture Editor

The panel layouts update to display the Perspective view in the left pane and the UV Texture Editor in the right pane simultaneously. (You can close the Attribute Editor if it is still displayed.)



Perspective view

UV Texture Editor

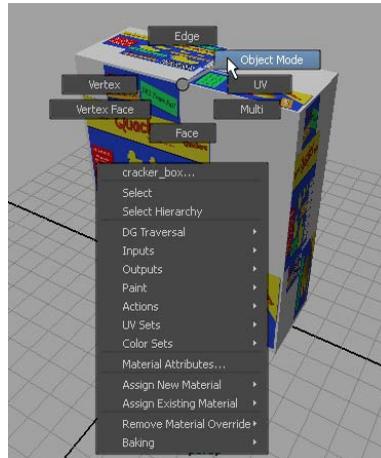
This two pane layout is helpful for two reasons:

- The two view layout shows you how one item selected in the 3D view relates to the UVs displayed in the 2D view of the UV Texture Editor and vice versa, without having to open and close the views repeatedly.
- When you edit UVs for a surface mesh in the UV Texture Editor, you can immediately see the effect of the texture map on the model in the 3D scene view.

UVs do not initially appear in the 2D view of the UV Texture Editor until you select an object or change the selection mode for an object in the scene view.

To view UVs in the UV Texture Editor

- 1 In the scene view, right-click any region of the cracker box model and select Object Mode from the pop-up menu.

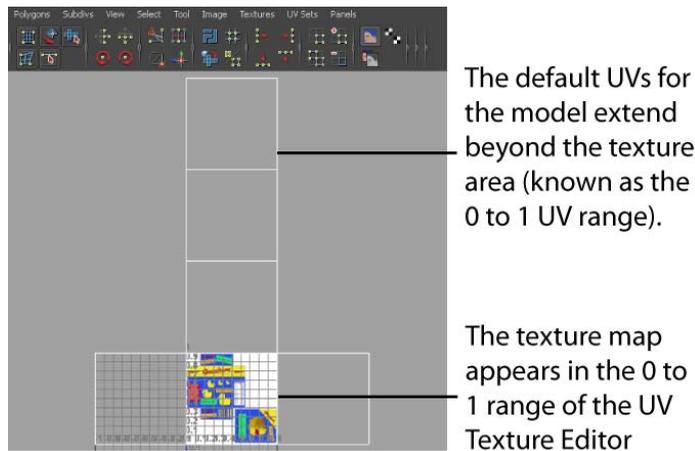


- 2 Select the cracker box model.
- 3 In the UV Texture Editor, dolly the view outwards so you can see the UVs for the cracker box as shown below. (To dolly, press the Alt key and drag the mouse to the left while holding down the right button on your mouse.)

The UV texture coordinates for the cracker box model appear in the 2D view of the UV Texture Editor as a flattened, two-dimensional representation. The UVs appear highlighted with lines connecting the UVs to indicate the region of the texture the UVs represent.

The image you specified as the texture map for the box also appears in the 2D view of the UV Texture Editor in the upper right quadrant of the 2D cartesian graph called the *UV image range* or *UV Texture Space*. The coordinates for this quadrant range from 0,0 to 1,1 and represent the texture space for the surface. How the UVs appear in this quadrant in relation to the displayed image has a direct bearing on how the texture gets mapped onto the surface.

TIP If the texture map for the cracker box doesn't appear in the 2D view, select Image > Update PSD Networks in the UV Texture Editor to refresh the 2D view of the UV Texture Editor. Update PSD Network is normally used to refresh a PSD texture in Maya after you have modified the PSD texture in Adobe® Photoshop®.



In this example, the UVs for the cracker box appear like a box where all six sides have been cut open and then unfolded flat.

The texture map does not appear correctly on the cracker box for a number of reasons:

- The UVs for the cracker box extend well beyond the default 0 to 1 UV range for the texture map in the 2D view of the UV Texture Editor. As the texture map displays within the 0 to 1 range, the UVs should also be positioned to fit within the 0 to 1 UV range, in most cases. Otherwise, the texture map repeats on the surface mesh, as it does in this case.
- The position of the UVs do not match the specific regions of the image we've provided for the texture map. The regions of the image show

the front, back, top, bottom, and sides of the box. The UVs should specifically match these regions to display the texture correctly. UVs do not automatically align themselves to a texture, you must manually reposition them.

- The shape of the UVs don't match the aspect ratio of the cracker box model in the scene view: -10 (Height) X 8 (Width) X 3 (Depth). This is because the default UVs for a Maya cube primitive are created based on a predetermined default shape and do not get updated if the shape or scale of the primitive is modified later on.

There are a number of things you can do to correct these issues depending on the situation. For this lesson, you will correct the UV and texture map misalignment by doing the following:

- Map a new set of UVs for the cracker box model that better matches the individual faces of the cracker box. (While the existing UVs could be modified, you'll learn how to create new UVs in this lesson that will better match the size and scale of the 3D model.
- Ensure the new UVs fit within the 0 to 1 UV range in the UV Texture Editor.
- Reposition the UVs so that they correlate to specific locations on the 2D image using the UV Texture Editor. This will ensure that the various sides of the box receive the correct regions of the texture map.

Mapping UV texture coordinates

It is often necessary to create new UVs for a surface mesh in order to texture map it correctly. These situations include:

- Texture mapping a surface that doesn't have existing UVs. This can occur when you import 3D models from other software applications that don't create UVs.
- When the UVs for a surface are badly jumbled or are missing some UVs. This can occur when a surface has been edited or modified in some way and it becomes hard to determine what UVs may be missing as a result.
- When you need a unique set of UVs for a particular purpose. For example, if you want to paint a texture map directly onto a 3D surface, you may want to map UVs that allow you to paint using the 3D paint tools

in Maya. Alternatively, you may want to create a unique set of UVs specifically for baking textures or light maps.

Maya lets you create UVs for polygonal and subdivision surfaces using a process called *projection mapping*, also referred to as *mapping UVs*. Maya provides several projection mapping types that map what gets viewed by a particular projection to a flat 2D view that can be subsequently correlated to your texture map using the UV Texture Editor.

In this lesson, you use a feature called *Automatic Mapping* to create new UVs for the cracker box model. Automatic mapping lets you specify the number of planes that will be used for the UV projection.

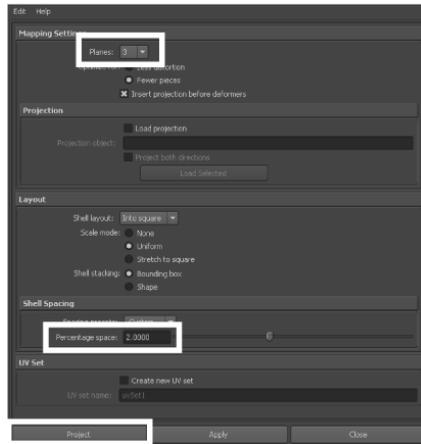
Automatic Mapping is normally used when mapping UVs that require multiple projections of organic shaped models as it has the advantage of producing UVs that are proportional in scale to the world space area of the mesh faces of the surface.

Automatic mapping is useful when your mapping requirements are for non-overlapping UVs which fit within the 0 to 1 texture space, but which do not have to be contiguous; for example, when using the 3D Paint tool, Maya® Paint Effects™, Maya® Fur™ or Maya® Hair™.

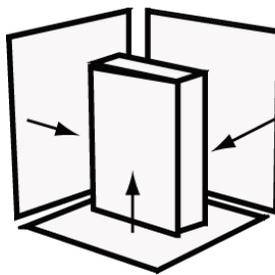
You use Automatic Mapping in this lesson because it can produce a UV projection from three planar directions simultaneously which makes it a good choice for the shape of the cracker box model. That is, you can map UVs for the front and back, top and bottom and the two sides in one operation. This projection type is referred to as a *triplanar projection*.

To map UVs using Automatic Mapping

- 1 In the perspective view, select the cracker box as an Object.
- 2 From the Polygons menu set, select Create UVs > Automatic Mapping >  .
The Automatic Mapping Options window appears.
- 3 In the Automatic Mapping Options window, select Edit > Reset settings, set the following options, and then click Project:
 - Planes: 3
 - Percentage Space: 2



Setting *Planes* to 3 creates UVs based on projections from three separate directions. Setting the *Percentage Space* option to 2 sets the size of the space that appears between each of the separate UV projections when they are laid out in the UV Texture Editor.



A triplanar UV projection creates UVs based on 3 projection planes.

When the projection is complete the new projected UVs from the triplanar projection appear in the UV Texture Editor.

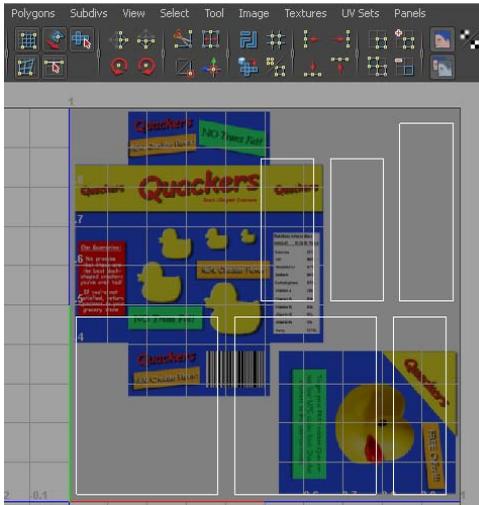
NOTE Whenever a UV projection is performed on a surface, any existing UVs for the surface get replaced by the UVs from the new projection. To avoid replacing existing UVs you can explicitly save the UVs in a UV set. For more information on UV Sets see the *Modeling* guide in the Maya Help.

- 4 In the UV Texture Editor, dolly the view so it shows only the 0 to 1 UV range area that displays the image for the texture map and the new UVs (as shown below).
- 5 In the UV Texture Editor, select Image > Dim Image.



Dim Image button

The intensity of the image map is reduced so you can view the UV borders on the projected UVs more easily. The new UVs appear in the UV Texture Editor as shown below.



- The six rectangular UV shapes, referred to as *UV shells* or simply *shells*, now match the aspect ratio for each of the six sides of the cracker box better than the earlier UVs.
- The UV shells fit within the 0 to 1 UV texture range. Automatic mapping fits the UVs to the 0 to 1 range by default.

The texture map still does not appear correctly on the cracker box because the UV shells need to be repositioned so they align with the corresponding components of the image map.

While the shape of each UV shell is now recognizable as being associated with the cracker box, it is not apparent which components in the UV Texture Editor correspond to a particular face on the cracker box.

In the next section you learn how to select and reposition the UV shells so they better match the image for the texture map.

Working with UVs in the UV Texture Editor

When texture mapping a polygon or subdivision surface type model it is often necessary to modify the UV components for a model so they match the texture map. The UV Texture Editor provides many tools for working with UVs. In this section you learn how to:

- Select UVs from either the scene view or the UV Texture Editor and accurately correlate them. Selecting UV components is critical to modifying them accurately.
- Sew UV shells together to reduce the number of UV components you need to work with.
- Move and rotate UVs within the UV Texture Editor to make the UV shells align to the cracker box texture map.

Selecting UV components

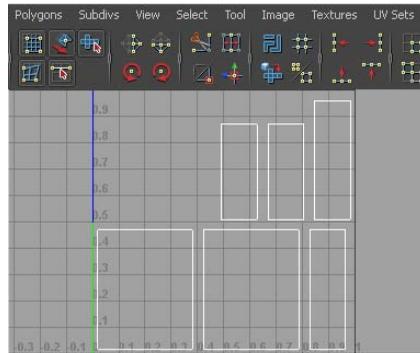
You can select UV components from either the scene view or within the UV Texture Editor. Displaying both the 3D scene view and the 2D view of the UV Texture Editor lets you easily correlate which UV components are associated with a particular edge or vertex on the 3D model.

This is very helpful to understand the layout and orientation of your UV texture coordinates in relation to the image used for a texture map and how the map appears on the model in the 3D scene view.

To select UVs for a 3D model in the scene view

- 1 In the UV Texture Editor, select Image > Display Image to turn off the display of the texture image.

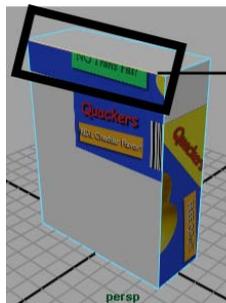
Display Image works by turning the display of the assigned texture on or off in the 2D view of the UV Texture Editor. Turning the image off temporarily aids in viewing only the UV components. This will be useful in the steps that follow.



- 2 In the scene view, right-click on the cracker box model and select *UV* from the pop-up menu that appears.
- 3 In the scene view, select one UV on any corner of the cracker box.
Selecting a UV is very similar to selecting a vertex on an object. That is, you select a point that resides in exactly the same position as the vertex. When you select a UV in the scene view, the corresponding UV is also selected in the UV Texture Editor.

NOTE If more than one UV gets selected in the UV Texture Editor when you select a UV in the scene view it indicates that multiple UV components share the same UV on the 3D model (similar to how polygon faces share common vertices). This is important to remember when you move one of these UV components in the UV Texture Editor and other UVs also move.

- 4 In the scene view, right-click on the cracker box model and select *Edge* from the pop-up menu that appears.
- 5 In the perspective view, select the top front edge of the cracker box.



Select the top front edge of the cracker box in the perspective view.

When you select this edge in the scene view, notice that two corresponding shell edges also get selected in the UV Texture Editor. This indicates the following:

- The selected edge is shared by two faces; the front and the top.
- Selecting the top front edge in the scene view, identifies the corresponding front and top UV shells for the box. (Indicated by the selection in the UV Texture Editor.)

This is useful when you need to understand how the various UV shells relate to each other, especially when you have many UV shells for a model in the scene; for example, when you sew two UV shell edges together.

In the steps that follow you select edges in the UV Texture Editor so you can sew them together.

Sewing UVs

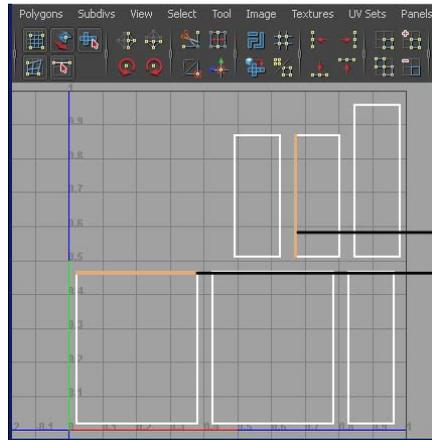
Sewing UV shells together merges the UV shells along a shared edge that you specify. Sewing UVs is useful for the following reasons:

- You can move and modify the UV shells as larger contiguous texture units. This makes it more efficient to match the UVs to the texture in many situations.
- Texture maps can appear more uniform across texture borders when the texture is applied to one or more sewn UV shells compared to many separate UV shells. Sewing shells together reduces the chance that an unwanted texture mismatch occurs along the texture edge.

To sew shell edges together in the UV Texture Editor

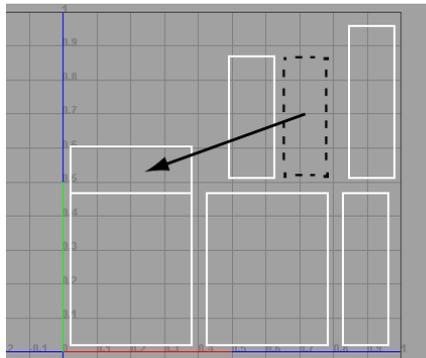
- 1 In the scene view, ensure that only the top front edge of the cracker box is still selected.

When the top front edge is selected in the scene view, the top edge of one of the large UV shells is also selected as well as the left side of the middle UV shell in the top row of the UV Texture Editor. For the cracker box texture map, you need to sew these together so they correlate to the map correctly.



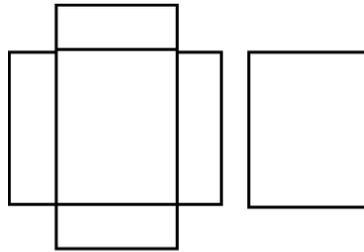
Selecting the top front edge of the cracker box in the perspective view selects edges on two UV shells in the UV Texture Editor.

- 2 In the UV Texture Editor, select Polygons > Move and Sew UV Edges. The small rectangular UV shell is repositioned to match the top of the large UV shell and the two shells are combined into one. When you later select and move this sewn shell, it will move as one piece. When two UV shells are selected for sewing, the smallest UV shell is moved to the larger.

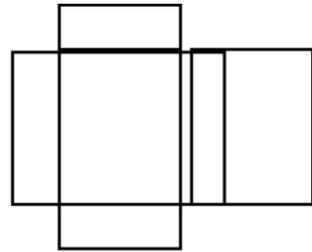


Move and Sew repositions one of the selected UV shells and merges the two shells into one.

- 3 Repeat steps 1 and 2 for the remaining three edges on front face of the cracker box keeping in mind the following:
 - Select only one edge at a time in the scene view.
 - When you've completed the move and sew operation, your UV Texture Editor should show the front UV shell with the corresponding top, bottom, and side shells sewn to it. When fully complete, you will have only two UV shells to match to the texture image for the cracker box.



Two UV shells remain after the Move and Sew operations.



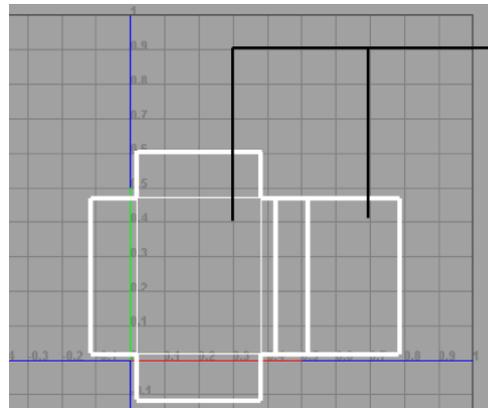
The UV shells will appear overlapping as shown above.

- To display the texture borders for the two shells, select Texture Border Edges. (You can also launch this window by right-clicking the Toggle Texture Borders button on the UV Texture Editor's toolbar.)



Toggle Texture Borders button

In the UV Texture Editor, the texture borders for two UV shells now appear with a thicker line.



Two separate UV shells remain after sewing (the shells overlap in this image).

Displaying the texture borders on the UV shells aids in visualizing where the borders of the texture map will exist on the surface.

Displaying texture borders on the UV shells is useful for visualizing where the texture borders exist on the 3D model in the scene view. That is, any portion of the texture image that appears outside of the texture border will not appear on the surface. It is particularly useful for troubleshooting texture mismatches when they occur. Displaying the texture borders can also help to identify

which UV shells are combined and which shells are not when you have many UV shells displayed in the UV Texture Editor.

If you want to be able to view areas where UV shells overlap in the UV Texture Editor select Image > Shade UVs. When Shade UVs is turned on any selected UV shells appear shaded in a semitransparent fashion. Areas where the shading appears more opaque indicate the regions of shell overlap.

In the steps that follow, you reposition the two UV shells in the UV Texture Editor so they coincide with the texture image.

Moving UV components

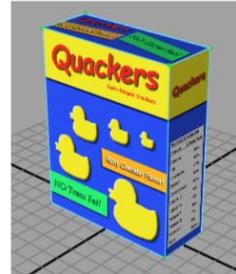
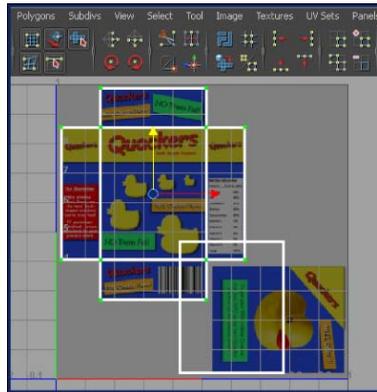
You manually reposition UV components in the UV Texture Editor using the tools within the Toolbox. To reposition an individual UV component you must first select it before selecting the Move Tool. To reposition an entire UV shell you must first select one individual UV from the shell it belongs to and then convert the selection to a shell before using the Move Tool.

To move UVs in the UV Texture Editor

- 1** In the UV Texture Editor, select Image > Display Image.
The texture image displays in the 2D view of the UV Texture Editor dimmed, as before. You need the image displayed so you can accurately reposition the two UV shells. Dimming the image aids in seeing the selected UVs.
- 2** In the UV Texture Editor, right-click and select UV from the pop-up menu that appears.
- 3** In the UV Texture Editor, select one UV on the UV shell you sewed earlier in the lesson.
The UV coordinate appears highlighted in the 2D view.
- 4** Convert the selection to a UV shell by pressing the Ctrl key and right-clicking in the 2D view of the UV Texture Editor. Choose *To Shell* from the pop-up menu that appears.
The selection changes from a single selected UV to selecting the entire UV shell.
- 5** From the UV Texture Editor menu, select Tool > Move UV Shell Tool > .
- 6** In the Move UV Shell Tool options window, turn off the Prevent overlap option which prevents UV shells from being moved so they overlap and then select Apply and Close.

The Move Tool manipulator icon appears over the selected UV shell.

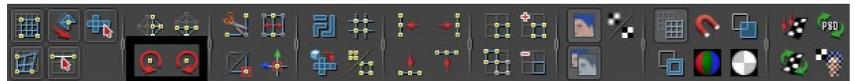
- 7 Drag the green (Y axis) and red (X axis) Move Tool manipulator handles to reposition the UV shell so it matches the displayed texture image.



Repositioning the UV shell to match the texture image repositions the texture map on the 3D model.

The position of the texture updates on the cracker box model in the scene view as you do this.

- 8 Select one UV on the other remaining UV shell.
The UV coordinate appears highlighted.
- 9 Convert the UV selection to a shell selection by pressing the Ctrl key and right-clicking in the UV Texture Editor. Choose *To Shell* from the pop-up menu that appears.
The selection changes from a single selected UV to selecting the entire UV shell.
- 10 In the UV Texture Editor's toolbar, click one of the Rotate UVs buttons to rotate the UV shell so it appears in the same orientation as the texture image. (Click the other Rotate UVs button to rotate it in the opposite direction.)



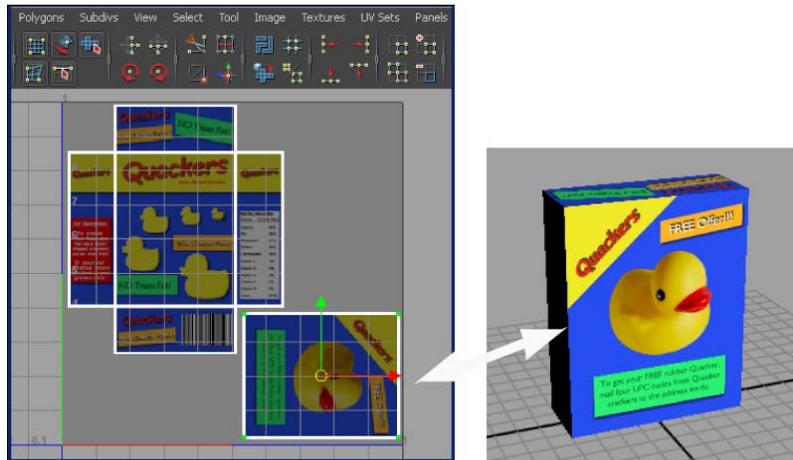
Click the Rotate UVs buttons to rotate the UV shell

- 11 Confirm that the texture appears correctly on the cracker box model in the scene view by tumbling the scene so you can view the back of the box.

TIP If the texture appears upside down, you can continue to rotate it another 180 degrees. If the texture appears reversed, that is, the letters are backwards, you can correct this by selecting Polygons > Flip from the UV Texture Editor menu and choosing the appropriate option setting.

- 12 From the UV Texture Editor menu, select Tool > Move UV Shell Tool. A move manipulator icon appears over the selected UV shell.
- 13 Drag the green (Y axis) and red (X axis) manipulator handles to reposition the UV shell so it matches the displayed texture image.
The texture map should now appear correct on all sides of the cracker box when you're finished.

TIP You can confirm that the UV shells are positioned accurately by dollying in closer on the 2D view of the UV Texture Editor. If a minor correction is required, simply select and move the UVs or move the entire UV shell.



Repositioning the remaining UV shell to match the texture image repositions the texture map on the back face of the 3D model.

TIP The exact UV boundaries can be viewed more clearly when the anti-aliasing for the texture image is temporarily turned off. To do this, select Image > Display Unfiltered to view the texture image without anti-aliasing.

If you need to move a single UV so it precisely matches a particular location on the texture map you can turn on Pixel Snap (Image > Pixel Snap). When you select and move the UV using the Move Tool, it will reposition the UV coordinate based on pixel boundaries.

Beyond the lesson

In this lesson you were introduced to a few of the basic techniques required to assign and accurately position texture maps on polygonal models using UV texture coordinates:

- UV coordinates are essential for applying and accurately positioning texture maps on polygonal and subdivision surfaces.
- UVs can be explicitly created for a surface mesh using a variety of UV mapping techniques. (Planar mapping, spherical mapping, cylindrical mapping, and Automatic mapping.)
- UVs are represented as 2D coordinates and are viewed and edited with respect to the 2D texture map using the UV Texture Editor. The UV Texture Editor provides many other tools for editing UVs.
- UVs are usually positioned to fit within the 0 to 1 UV coordinate space unless the texture map requirements are such that the map needs to repeat across the surface mesh.
- It is useful to display both the 3D scene view and the 2D UV Texture Editor simultaneously when texture mapping so that a correlation between the 3D model and 2D UV coordinates can be made.
- You can visualize texture borders using Texture Border Edges.
- You can view the regions where UVs overlap each other in the UV Texture Editor by selecting the UV shells and selecting Image > Shade UVs.

You can open the PSD file you used in this lesson using Adobe® Photoshop® to learn how layers were used for the texture map. You can edit one or more of the layers to experiment with how the texture can be modified. After you modify and save the file in Photoshop®, remember to select Image > Update

PSD Network in the UV Texture Editor so the texture map gets updated on the model in the Maya scene.

If you would like to learn how to render an image of the cracker box model see the *Rendering* chapter of this guide as well as the Maya Help to learn more about lighting, shading, and rendering.

This lesson used a simple polygonal model to introduce the fundamental concepts. When you work with more complex polygonal and subdivision surface models, you can also:

- layout overlapping UV shells that occur as a result of UV projection mapping to ensure they fit within the 0 to 1 texture space using Edit UVs > Layout.
- maximize the use of texture space by forcing the UV borders to be mapped to the full area of the 0 to 1 UV texture range using Edit UVs > Map UV Border
- align UVs to one another or snap them to the grid in the 2D view
- untangle a UV mesh that contains overlapping UVs using Edit UVs > Relax.
- modify a group of selected UVs using UV Lattice and UV Smudge tools

In general, you should begin texturing a model only after the model is fully complete. Otherwise, changes to the model may affect the associated UV texture coordinates which in turn will affect how the texture appears on the model.

In this lesson you matched the UVs to an existing texture image. Many texture artists create good UV layouts for their models prior to creating the actual 2D images for their texture maps. This is done only after the model is fully complete.

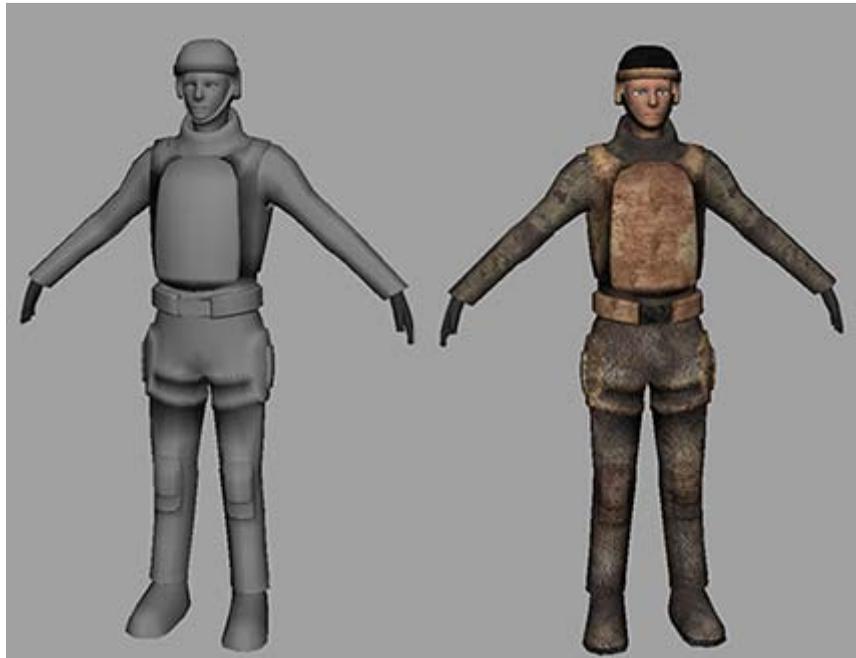
You can export a bitmap image of the 2D view of the UV Texture Editor to use as a guide for painting a texture in your favorite image creation software. In the UV Texture Editor, select *Polygons > UV Snapshot* to export the image.

If you use Adobe® Photoshop® for image creation and editing you can create a UV snapshot in a .PSD file format that creates the UV image on a separate layer. To create the .PSD format file while working in the UV Texture Editor, select *Image > Create PSD Network*.

If you want to learn more about a particular tool or feature that was used in this lesson, please refer to the Maya Help.

Lesson 2: UV unfolding

Introduction



While texturing a flat object is fairly straightforward, texturing an object with curves can pose some unique problems. How do you paint a 2D texture on a curved surface? How do you deal with a texture warping over bumps and grooves? If you try to break apart the UV mesh like Lesson 1, you can potentially end up with hundreds of small pieces.

To deal with these problems, artists typically use a workflow called unfolding. Unfolding a UV mesh refers to the process of cutting a seam in the UV mesh (a mesh made up of UVs similar to how a polygon mesh is made of vertices) and then unfolding along that seam. The process is similar to cutting a seam along a shirt and laying it flat on a table. By laying out the UVs this way, you can easily paint a texture on the 2D surface, which you can then wrap around the model.

In this lesson you learn how to:

- Divide a mesh into separate parts for texturing.
- Assign a basic checker pattern to judge the quality of a mapping.
- Create a planar mapping.
- Cut UV edges.
- Unfold a UV mesh using the Unfold option.
- Fix an unfolded mesh using the Smooth UVs tool.

Lesson Setup

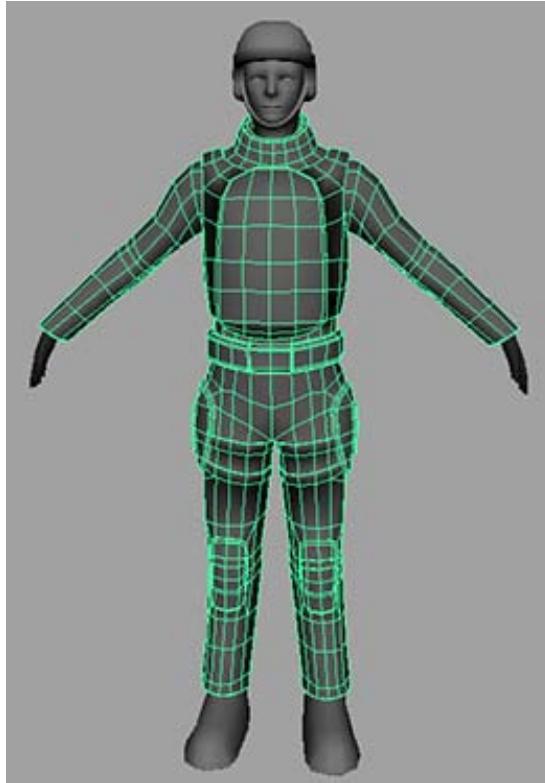
To ensure the lesson works as described, do these steps before beginning:

- Make sure you've done the steps in [Preparing for the lesson](#) on page 360.
- Select the Polygons menu set by selecting it from the pull down menu on the status line.
- Open the scene file named Soldier.ma.
This file can be found in the GettingStarted directory that you set as your Maya project.

`GettingStarted/UVMapping/Soldier.ma`

The scene contains a polygon soldier.

Dividing the mesh



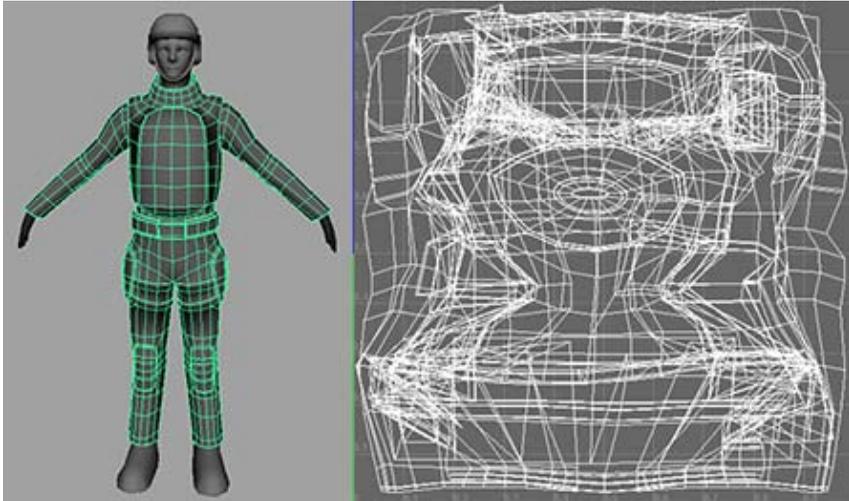
When working with UVs, it is useful to divide the workspace so you can see both the model and the UVs at the same time.

To view the soldier model and its UVs side by side

- ▶ In the panel menu, select Panels > Saved Layouts > Persp/UV Texture Editor.

The perspective view and UV Texture Editor appear side-by-side in the viewport.

If you select the polygon soldier you will notice that the UVs for the body, arms, and legs are scattered in a pattern that doesn't resemble the actual model. This isn't very helpful for painting a texture map.

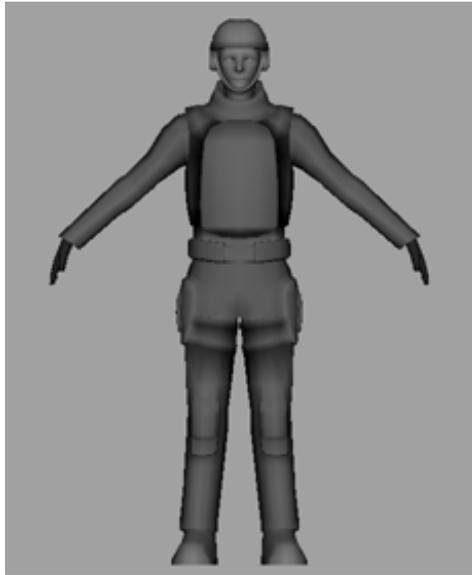


To simplify the job, you need to divide the mesh into multiple pieces that you can then texture individually. Generally, you want to divide the mesh into pieces that can be unfolded into approximate square or rectangle shapes. This requires some pre-planning.

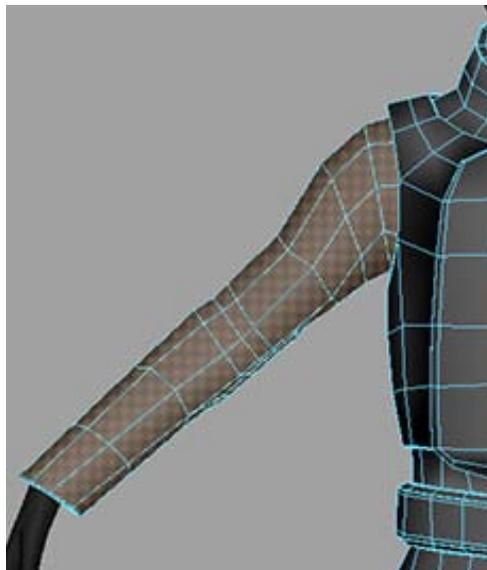
In this lesson you divide the body mesh into a torso, arms, and legs.

To divide the mesh into pieces

- 1 In the panel menu select Panels > Orthographic > front.
The perspective view changes to an orthographic front view of the soldier.



- 2 Right-click the mesh and select Face from the pop-up menu.
- 3 Drag-select the faces on the right arm up to the shoulder.



- 4 Select Mesh > Extract.

Now you change the selection mask back to Object, you can select the hand separately from the rest of the body.

- 5 Repeat steps 1-3 for the following areas (use the pictures as reference for the boundaries of each extraction).

Extract Point

Example

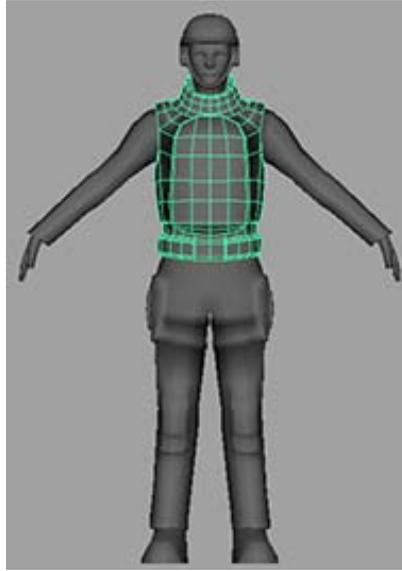
Left arm



Legs



This leaves you with individual meshes for the arms, legs, and chest. The head, feet, and hands have already been done for you. This division allows you to texture each part individually.

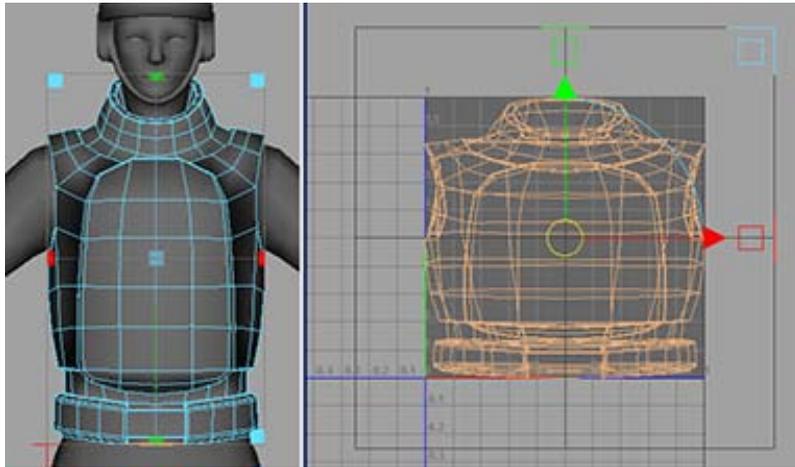


Creating a planar map

To texture an organic object, you must start by creating a planar mapping as a base, which you can then unfold.

To create a planar map for the torso

- 1 Right-click the torso and select Object mode from the pop-up menu.
- 2 Select the torso.
- 3 Select Create UVs > Planar Mapping >  .
The Planar Mapping Options window appears.
- 4 Select Edit > Reset Settings.
- 5 Set Project From to Z axis.
- 6 Click Project.



Maya creates a new UV mapping for the torso. If you look at the UV Texture Editor, you can see the new UVs look like a flattened version of the front of the chest. At first glance this may seem perfect, but there's a major problem here that you need to change the UV Texture Editor view to shaded mode to see.

To turn on shaded UV display

- 1 Select Window > UV Texture Editor.

The UV Texture Editor appears.

- 2 In the UV Texture Editor, click the Shade UVs icon (), or select Image > Shade UVs.

In shaded UV mode, overlapping faces appear red while non-overlapping faces appear blue.



Notice that all the faces for the torso are red. This is because the faces of the front and back of the torso are overlapping. If you tried to apply a texture to the UV mesh like this, you would get the exact same image on the front and the back of the torso.

To fix this, we need to unwrap the texture so that front and back no longer overlap.

Unfolding a UV mesh

Unfolding a UV mesh is the process of cutting a mesh so that you can lay its entire surface area flat. However, before you do this, you want to make sure your UVs are spread out evenly along the surface. You can do this by applying a basic checker pattern to the mesh.

To apply a checker pattern to the torso

- 1 Right-click the torso and select Assign New Material from the pop-up menu.
The Assign New Material window appears.
- 2 Select Lambert.
- 3 Open the Attribute Editor.
- 4 In the lambert tab, set the lambert field to checker_pattern.
- 5 Click the box next to Color.
The Create Render Node window appears.
- 6 Select Checker.

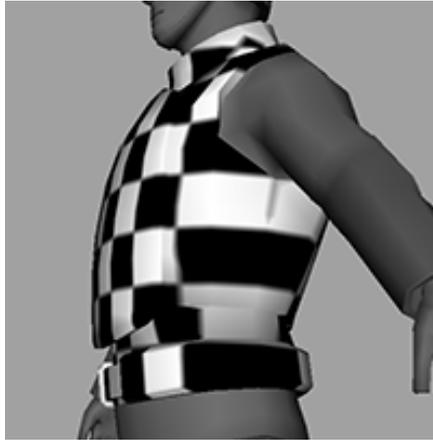
- 7 Press '6' to enter Smooth Shade mode.
The torso is covered in a checker pattern.



- 8 In the panel menu, select Panels > Perspective > persp.

From the front view, the checker pattern looks roughly even with very little distortion. This indicates that the spread of UVs is fairly even. An even spread of UVs is important to avoid texture warping when you apply your texture. In some cases you may need to experiment with different planar mappings to give you the best base to start with.

However, the planar projection isn't perfect. If you look at the sides of the torso, from the armpits to the waist, you can see some warping. This is the ideal place to cut the UV edges for unwrapping.



To cut the UV edges

- 1 Right-click the vest and select Edge from the pop-up menu.
- 2 Select all the edges in a line that runs from under the arm, down the side of the soldier's body to the bottom of the vest. You can do this by double-clicking edges along this path until they are all selected.



- 3 Shift-select the corresponding edge on the other side. An edge loop on both sides should be selected now.

- 4 In the UV Texture Editor, select Polygons > Cut UV Edges.

Now you are ready to unfold the mesh.

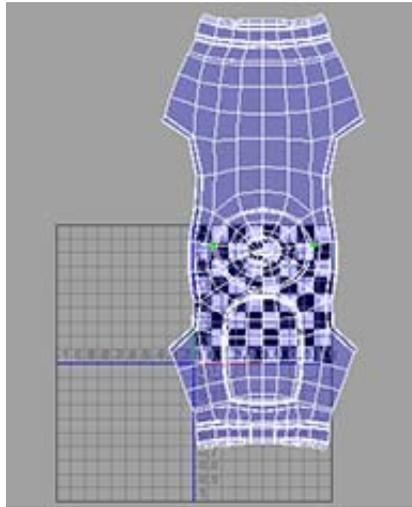
To unfold the torso

- 1 Right-click the torso and select UV.
- 2 Shift-select one UV at the inside tip of each vest strap, along the unfold line.



- 3 In the UV Texture Editor, select Polygons > Unfold >  .
The Unfold UVs Options window appears.
- 4 Select Edit > Reset Settings.
- 5 Set Pin UVs to Pin selected UVs.
A pinned UV does not move when a UV mesh is unfolded. In general, you want at least one pinned UV when unfolding because it gives the unfold process a point of reference. This is similar to placing a pin in a piece of fabric so that it does not unfurl out of control as you unfold it.
- 6 Click Apply and Close.

Maya lays the UV mesh out across the UV space. Parts of it may fall outside the 0-1 texture space, but that's fine for now. More importantly, notice that the pattern is much more even across the mesh now.



Unfortunately, while you've minimized the distortion, you've also create a visible seam where you cut the UV edges. Unfortunately this is unavoidable. In general it is best to put your seams in the least visible places. In this case, the seams will be obscured by the arms most of the time and therefore won't be as noticeable.



Adjusting the checker pattern

Although not always necessary, it is often helpful to increase the number of squares in your checker pattern. This can help you to more easily spot deformities in the pattern.

To increase the number of squares in the checker pattern

- 1 Right-click the torso and select Object Mode from the pop-up menu.
- 2 Select the torso in the perspective view.
- 3 Open the Attribute Editor and navigate to the checker_pattern tab.
- 4 Click the box next to Color.
- 5 In the place2dTexture tab, set Repeat UV to 10, 10.
The number of squares on the torso increases.

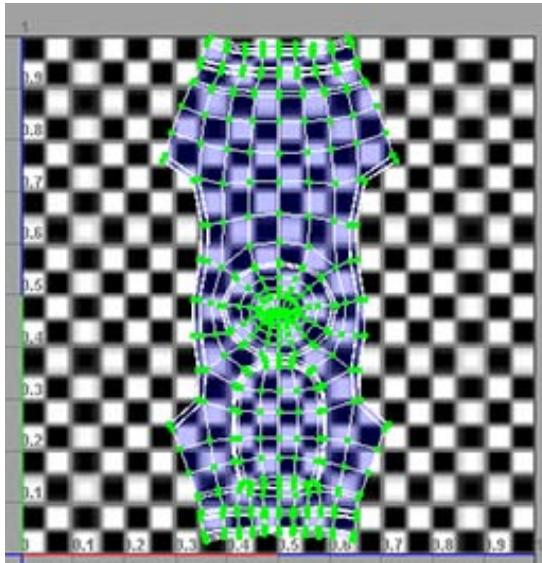


Outputting UVs

Now that you have your unfolded UVs, you need to export them so that you can draw a texture on them. You can do this by taking a snapshot of your UV layout.

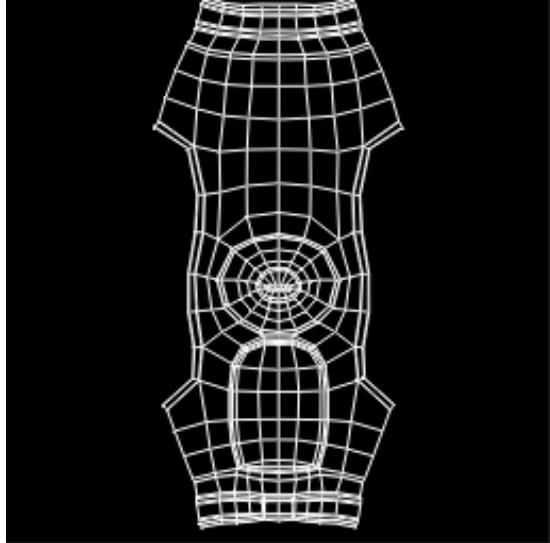
To export a UV snapshot

- 1 Right-click the torso in the perspective view and select UV from the pop-up menu.
- 2 Drag-select all the UVs on the torso.
- 3 In the UV Texture Editor, use the Scale Tool and Move Tool to transform the mesh so that it fits inside the upper-right square of the UV Texture Editor.



- 4 Right-click the torso in the perspective view and select Object Mode.
- 5 Select the torso.
- 6 In the UV Texture Editor, select Polygons > UV Snapshot.
The UV Snapshot options window appears.
- 7 Click the Browse button.
The file browser appears.
- 8 Navigate to `GettingStarted/UVMapping` and enter the name `torso_UVs`.
- 9 Click Save.
- 10 Set Image Format to Maya IFF.
- 11 Click OK.

Maya creates a file named torso_UVs.iff in the Getting Started folder. If you open this file, you can see the exported UV mesh.



You can now open this file using an external image editor and paint your texture using the mesh as a guide.

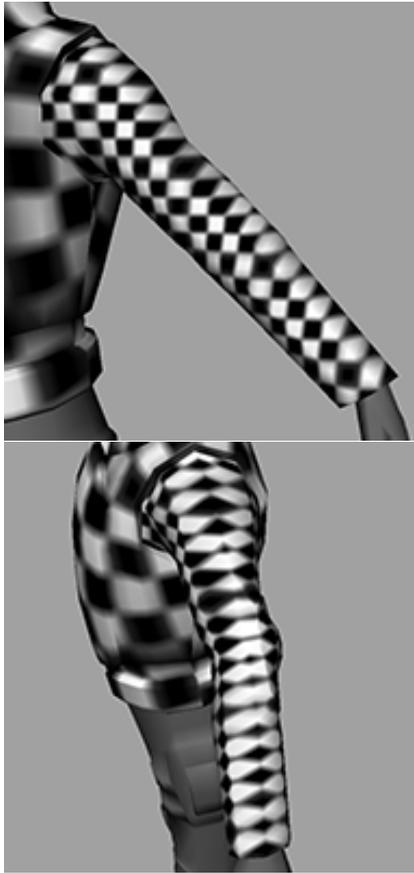
NOTE If your image editing software does not support Maya .iff images, you can change the image format in the UV Snapshot options window and then export the file again.

Unfolding with constraints

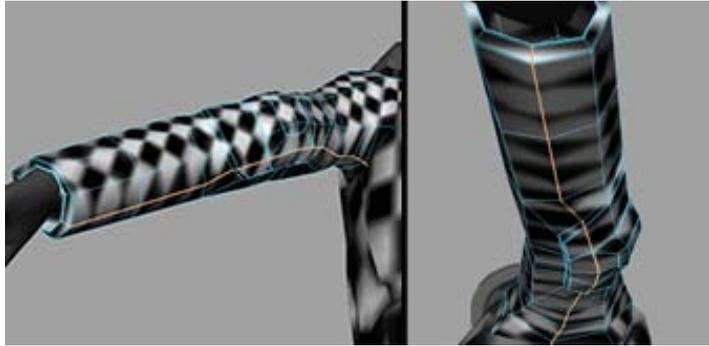
Sometimes you only need to unfold a UV mesh in one direction. This is especially useful when there is only a single seam.

To unfold the arms

- 1 Select the soldier's left arm.
- 2 Select Create UVs > Planar Mapping.
- 3 Right-click the arm and select Assign Existing Material > checker_pattern.

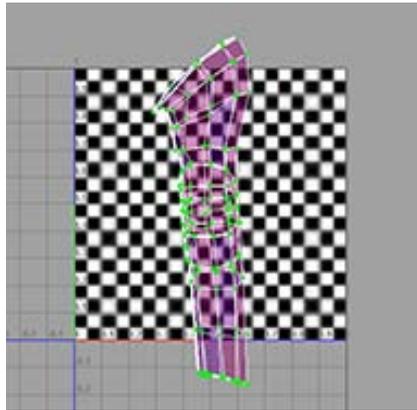


- 4 Right-click the arm and select Edge from the pop-up menu.
- 5 Select the edges on the bottom of the arm at the center of the distortion just under the elbow.

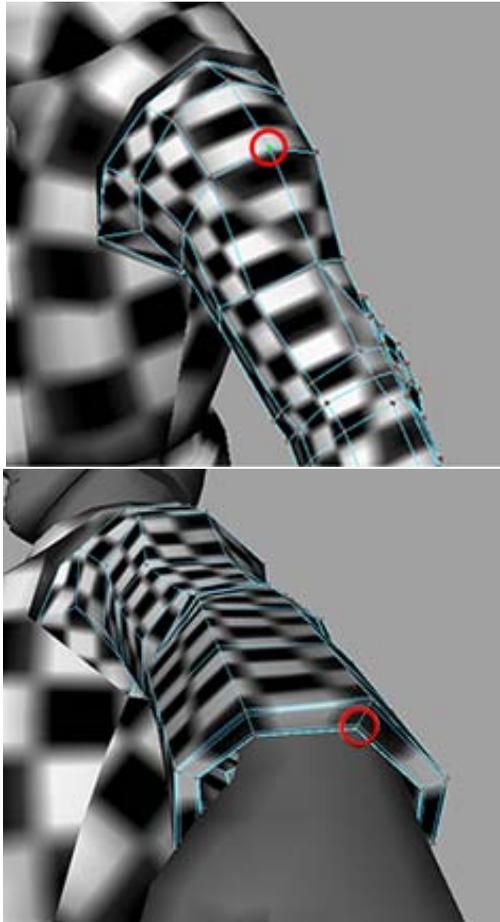


- 6 In the UV Texture Editor, select Polygons > Cut UV Edges.
- 7 Right-click the arm and select UV from the pop-up menu.
- 8 Drag-select all the UVs on the arm and click the Rotate clockwise button ().

The UVs rotate so the arm in the UV Texture Editor is vertical.

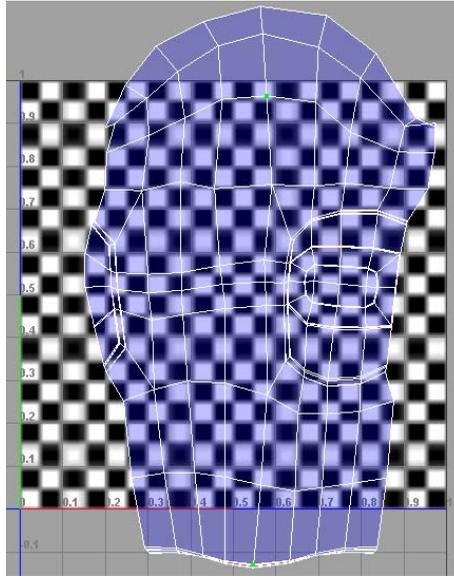


- 9 Shift-select one UV at the top of the arm's bicep and another UV at the tip of the soldier's cuff.



- 10 Select Polygons > Unfold >  .
The Unfold UVs Options window appears.
- 11 Set Unfold constraint to Horizontal.
- 12 Click Apply and Close.

The UV mesh for the arm is unfolded evenly up and down. This is better than the UV mesh we would get if we unfolded in all directions.



Now you can output these UVs to a file.

To export the UV snapshot

- 1 Right-click the arm in the UV Texture Editor and select UV from the pop-up menu.
- 2 Drag-select all the UVs on the UV mesh.
- 3 Select the Scale Tool.
- 4 In the UV Texture Editor, scale the UV mesh so that it fits inside the upper-right square of the UV Texture Editor.
This step is necessary because a UV snapshot only captures what is in the 1 x 1 UV texture space (the upper right corner).
- 5 Right-click the arm in the perspective view and select Object Mode.
- 6 Select the arm.
- 7 In the UV Texture Editor, select Polygons > UV Snapshot.
The UV Snapshot options window appears.
- 8 Click the Browse button.
A file browser appears.

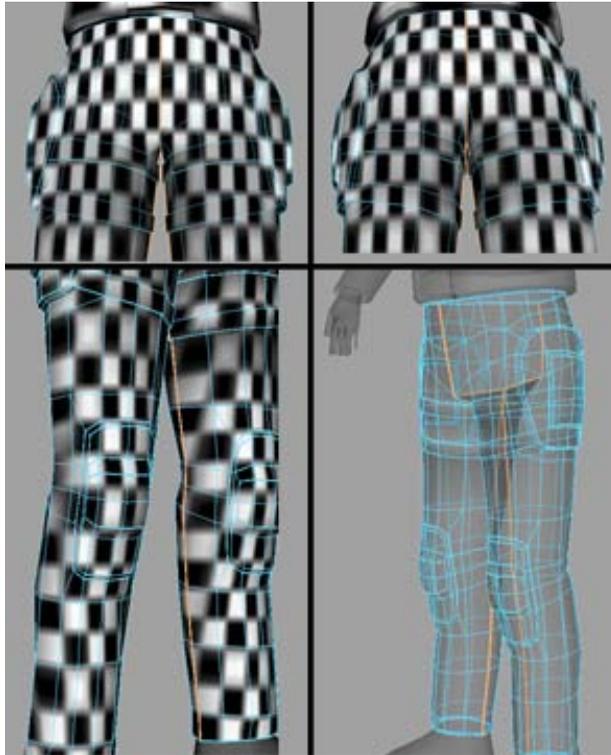
- 9 Navigate to `GettingStarted/UVMapping` and enter the name `left_arm_UVs`.
- 10 Click Save.
- 11 Set Image Format to Maya IFF.
- 12 Click OK.

Like the torso you did in the previous step, Maya outputs an .iff file containing the UV mesh. You can import this file into an image editor and use it as a template to create a texture.

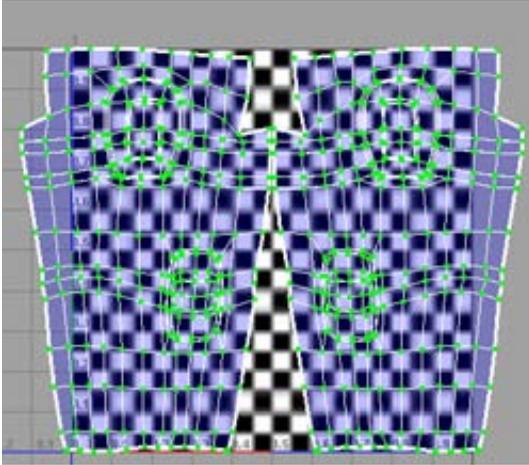
Now repeat the entire procedure for the right arm. Note that you do not need to output the UVs for the right arm because it will use the same texture as the left arm.

Although you could handle the legs the same way you handled the arms (cutting them into two pieces), you can also treat the entire lower body as a single mesh. You can follow the same procedure as above, except for the following:

- Select the edges down the rear until directly underneath the groin then select the edges on the inside of each leg. Cut these UV edges.



- You do not need to pin any UVs for the pants (but you must select the pants as an object to unfold the UVs).
- Like the arms, unfold the UV mesh horizontally.



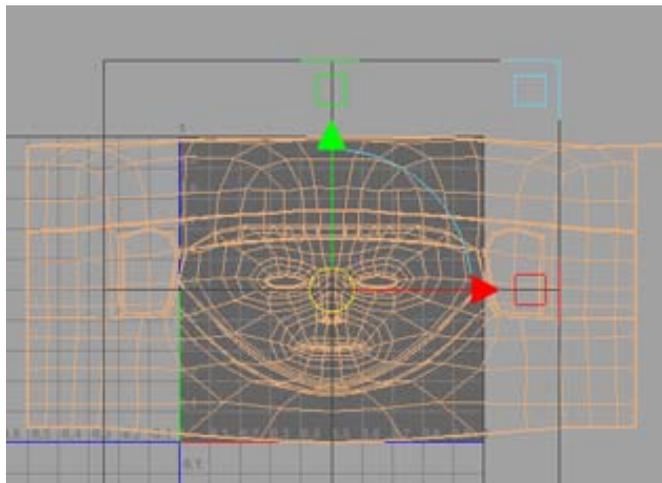
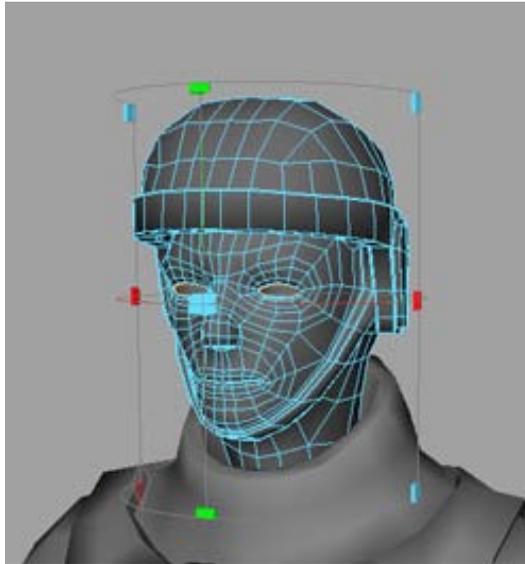
As with the torso and arms, you can now output these UV meshes to an external file to paint a texture on them. For information about how to do this, see [Outputting UVs](#) on page 400.

Sewing UV Edges

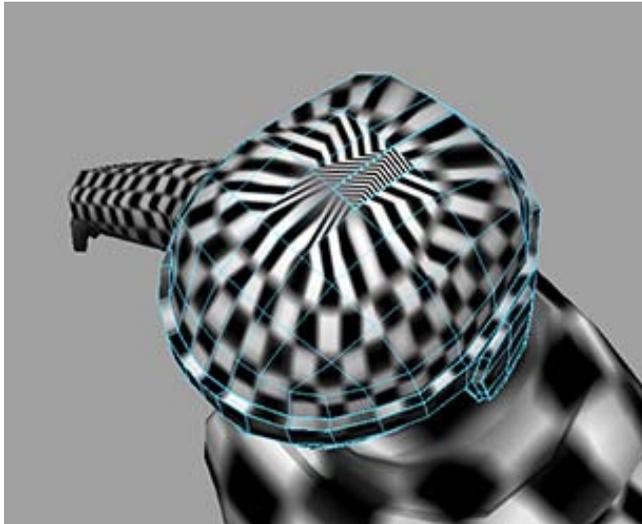
Depending on the mapping you start with, you may sometimes find that your UV mesh is broken up in ways that can ruin your final result. In these cases, you can manually sew UV edges back together.

To map the head to the UV space

- 1 Select the head and select Create UVs > Cylindrical Mapping.



- 2 Right-click the head and select Assign Existing Material > checker_pattern.



Notice that because you started with a cylindrical mapping, the checker pattern on the sides of the head aren't warped like in the previous examples. Unfortunately, if you tumble the camera you can see that the top of the head is now warped. In this particular case this is preferable and you will see the reason why later.

If you turn on texture border display by clicking the Texture Border Display

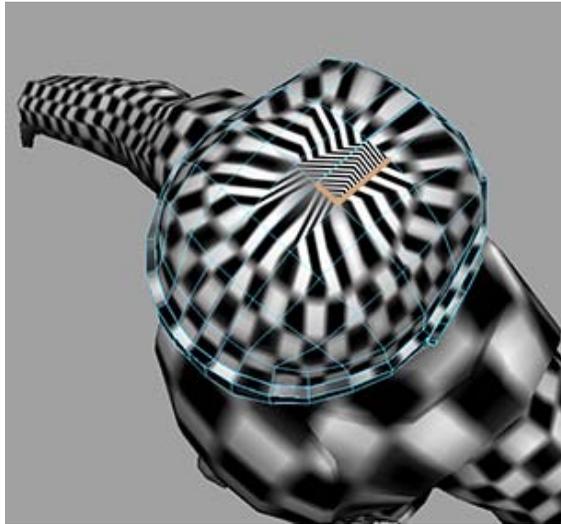
icon () in the UV Texture Editor, and re-examine the top of the helmet, you can see that a few edges that are not on the line of symmetry are colored thick blue. This indicates that they are on the border of the UV mesh and will cause a seam. You can fix this by sewing these overlapping UV edges together.

To sew UV edges

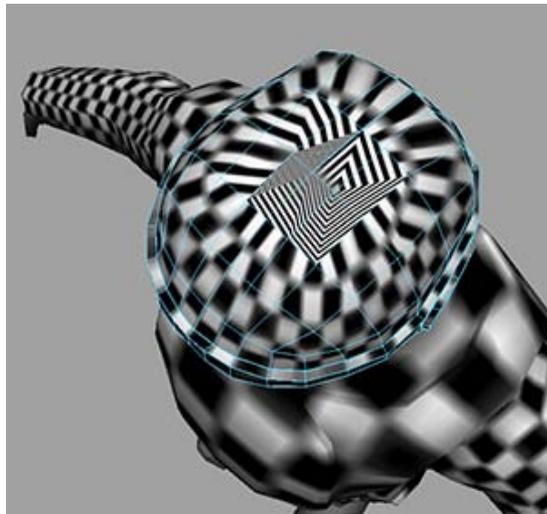
- 1 Right-click the helmet and select Edge from the pop-up menu.
- 2 Shift-select the two thick blue edges at the top of the helmet that are not on the line of symmetry.

NOTE If none of the edges appear thick blue, click the Texture Border Display

icon () in the UV Texture Editor



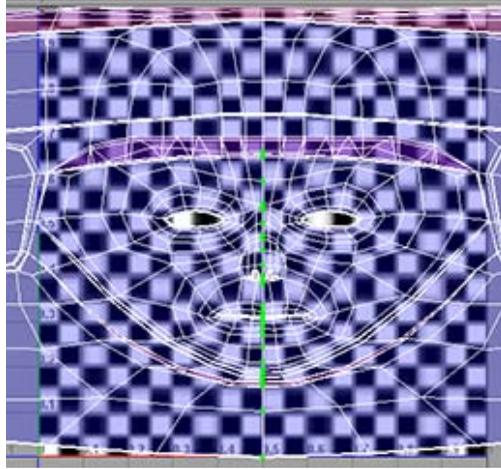
- 3 In the UV Texture Editor, select Polygons > Sew UV Edges. The thick blue edges change to match the other edges.



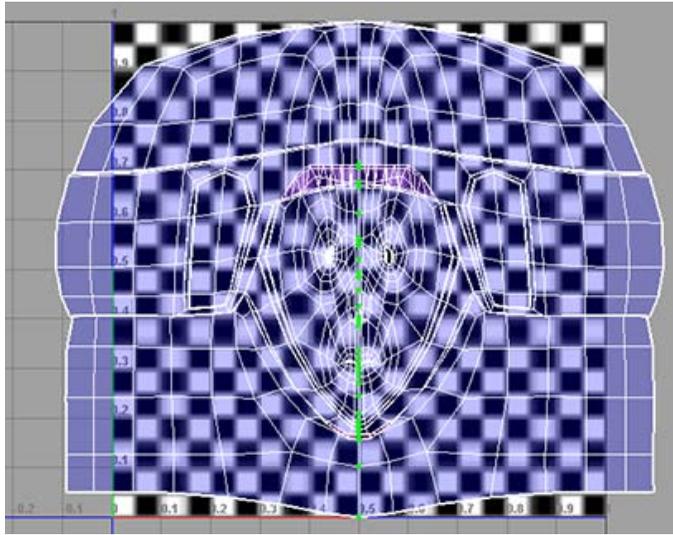
Notice that after sewing the UV edges together, the distortion has actually gotten worse. However, this is only temporary and is fixed when you perform the unfold.

To unfold the helmet's UV mesh

- 1 In the UV Texture Editor, right-click the face mesh and select UV from the pop-up menu.
- 2 Select the UVs down the center of the face starting from the tip of the helmet's rim to the bottom of the neck.



- 3 Select Polygons > Unfold > .
The Unfold UVs Options window appears.
- 4 Set Pin UVs to Pin selected UVs.
- 5 Click Apply and Close.
The face unfolds evenly across the texture space.

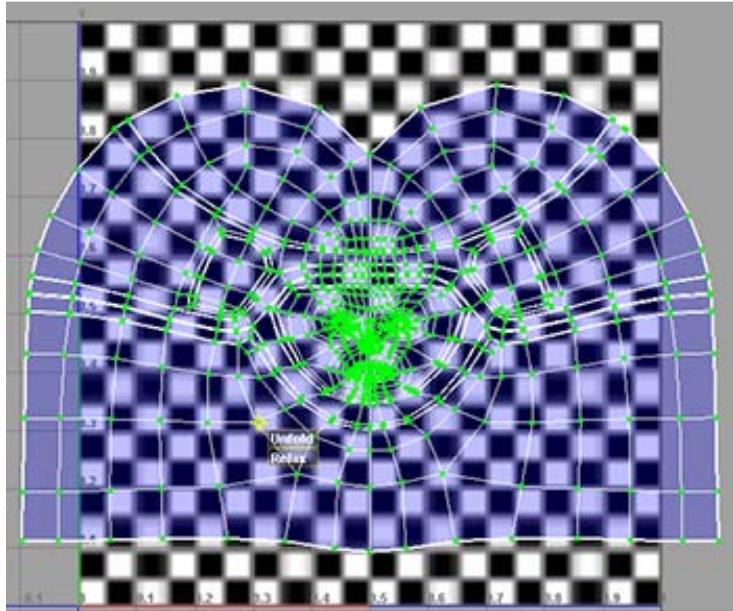


Smoothing/Relaxing a mesh interactively

Sometimes you only want to partially unfold a mesh. For this, you can use the UV Smooth tool to interactively unfold or relax the mesh.

To partially unfold the UV mesh

- 1 In the UV Texture Editor, select all the UVs.
- 2 Select Tool > Smooth UV Tool >  .
The Smooth UV Tool options window appears.
- 3 Select Edit > Reset Settings.
- 4 Set Space to UV Space.
- 5 Click Apply and Close.
Two controls appear over the UV mesh called Unfold and Relax.
- 6 Click Unfold and drag the mouse to the right.
The UV mesh unfolds as you drag the mouse to the left.
- 7 Continue to drag until the bottom of the UV mesh until it roughly resembles the picture below.



If you reach the edge of the screen you can release the mouse button, click the Unfold control and drag again.

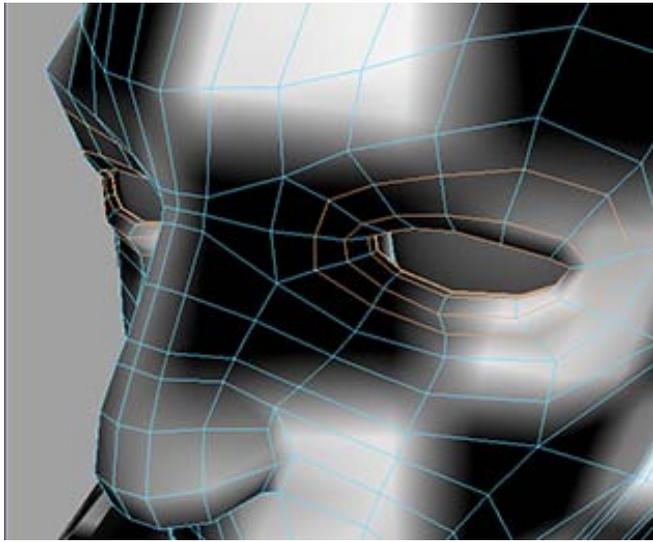
Fixing problem areas

You now have an unfolded UV mesh with no major overlapping faces. However, there are still a few minor problem areas where the mesh overlaps itself (particularly the eyes and mouth).

You can use the Smooth UV Tool to fix small imperfections in the shell with targeted unfolds or relax operations.

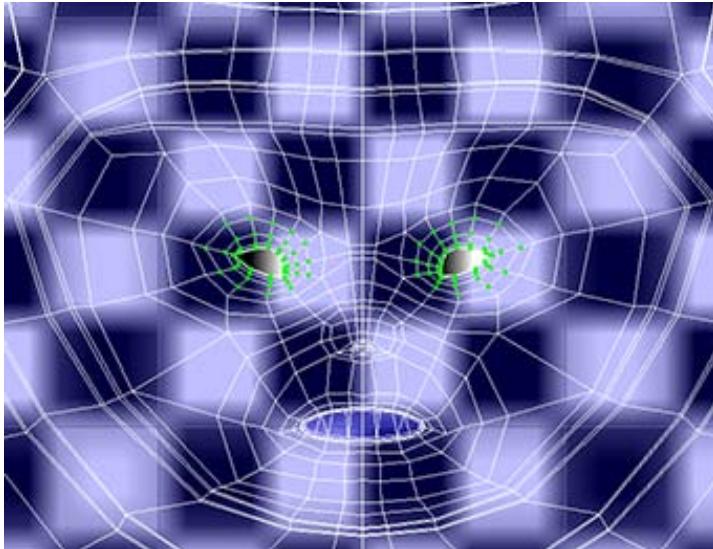
To fix the eyes

- 1 In the perspective view, right-click the soldier's head and select Edge from the pop-up menu.
- 2 Shift-select the four closest edge loops that surround each eye. You can do this quickly by Shift + double-clicking one edge in each loop.

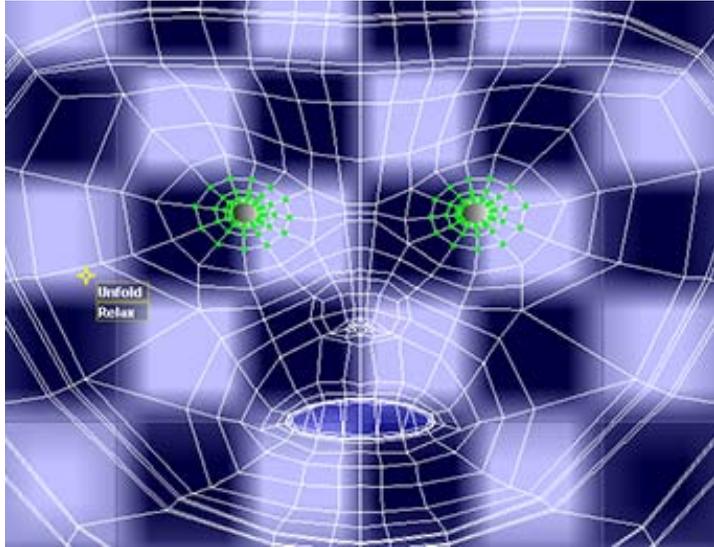


- 3 In the UV Texture Editor, Ctrl + right-click the UV mesh and select To UV from the pop-up menu.

The edges selected around the eyes are converted to the UVs selected around the eyes.



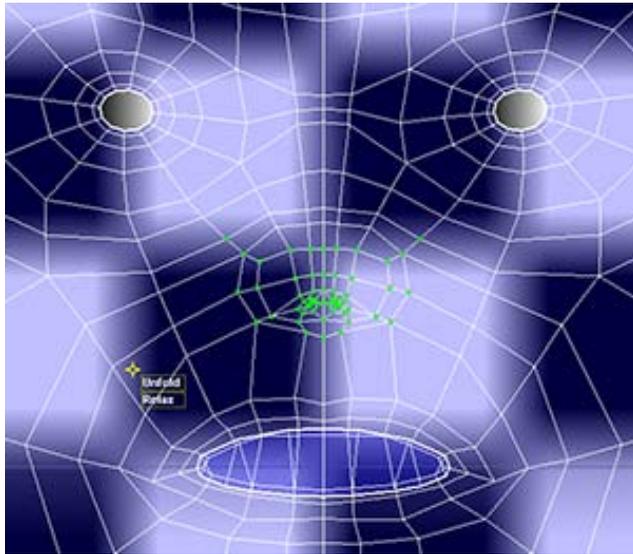
- 4 Click the Smooth UV Tool icon ().
- 5 Drag the Relax control to the right until the eyes appear round.
As you drag the Relax control, the selected UVs relax until they are round.



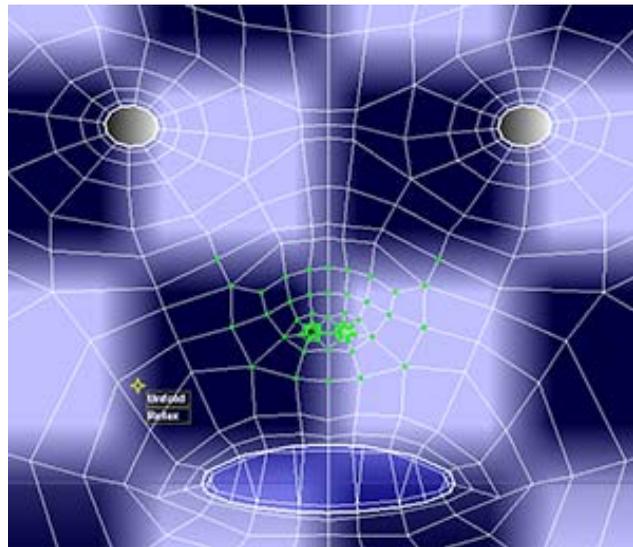
Now you can fix the nose and mouth in a similar manner.

To fix the nose

- 1 Select a group of UVs on and around the nostrils.

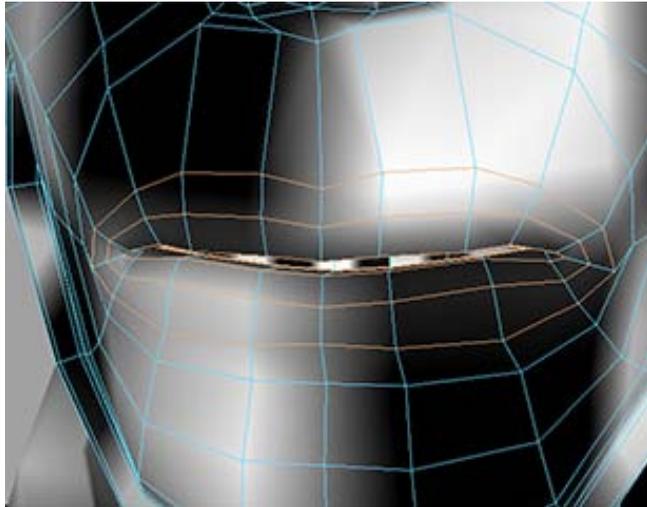


- 2 Drag the Relax control to the right until the UVs stop moving.

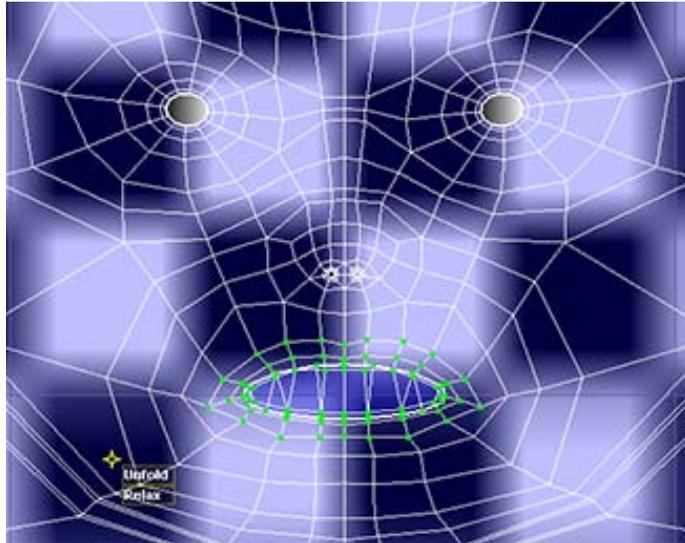


To fix the mouth

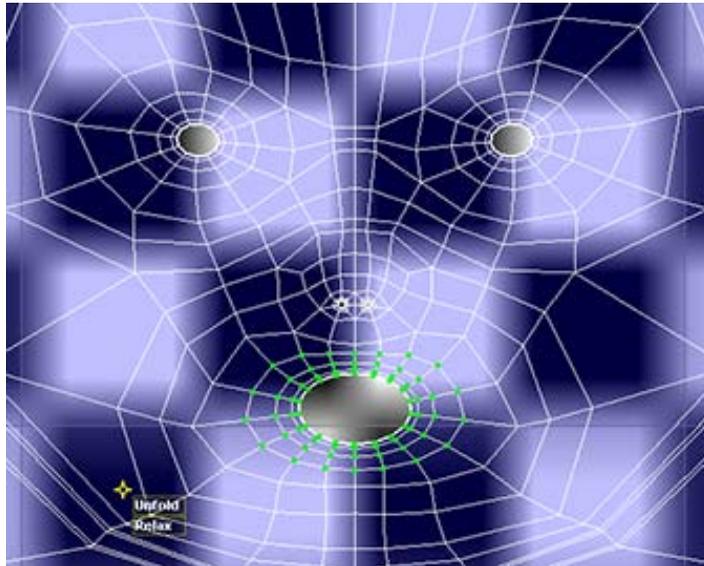
- 1 In the perspective view, right-click the head and select Edge from the pop-up menu.
- 2 Shift-select the 4 edge loops around the mouth. You can do this quickly by Shift + double-clicking one edge in each loop.



- 3 In the UV Texture Editor, Ctrl + right-click the UV mesh and select To UV from the pop-up menu.
The edges selected around the mouth are converted to the UVs selected around the mouth.



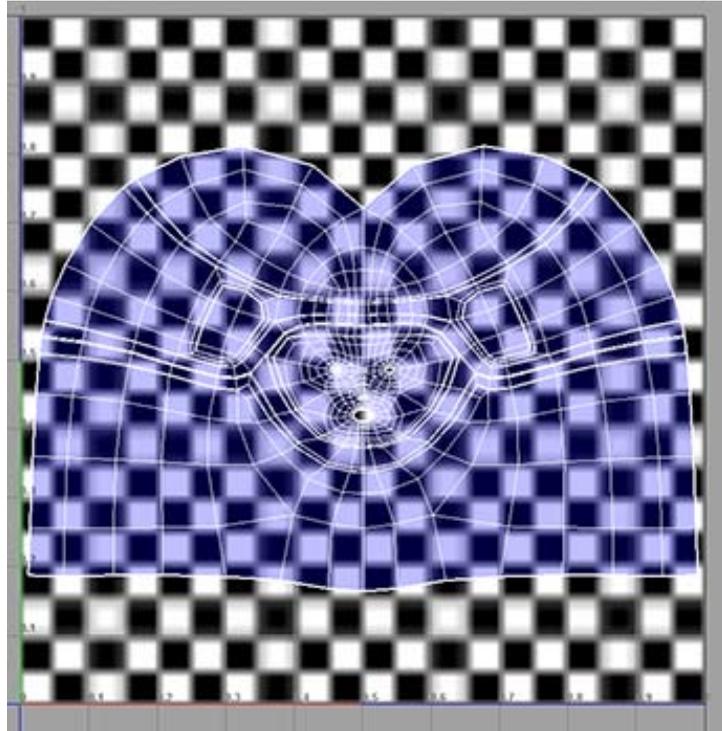
- 4 Drag the Relax control to the right until the UVs stop moving. The mouth UVs even out into a circular shape.



- 5 Select all the UVs on the head.

- 6 Select the Scale Tool and scale the UVs so that they fit in the upper right corner of the UV space.

The UV mesh for the soldier's head now has no UV faces overlapping. This allows you maximum control over what shows up in each portion of the head when you create the texture map for the soldier.



As with the rest of the body, you can now output these UV meshes to an external file to paint a texture on them. For information about how to do this, see [Outputting UVs](#) on page 400.

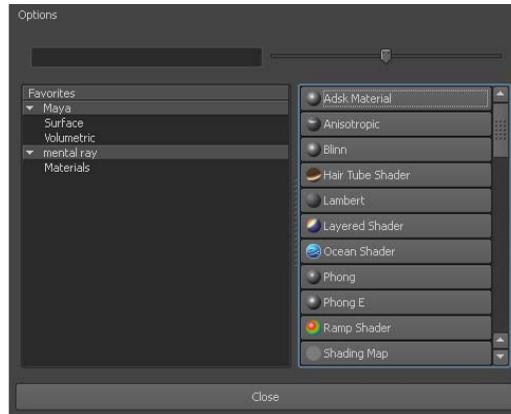
Applying textures

Now that you've outputted all your UV meshes, you can import them into an external image editor to create a texture.

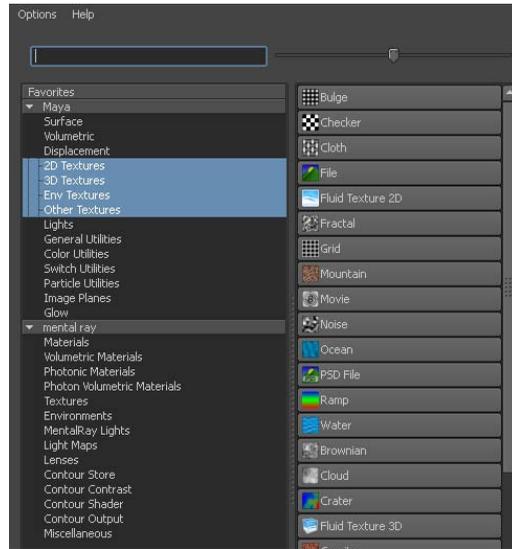
Once you've created associated texture files, you need to apply them to your model. For the purposes of this tutorial, textures have been provided for you in the `sourceimages` folder.

To apply the texture for the torso

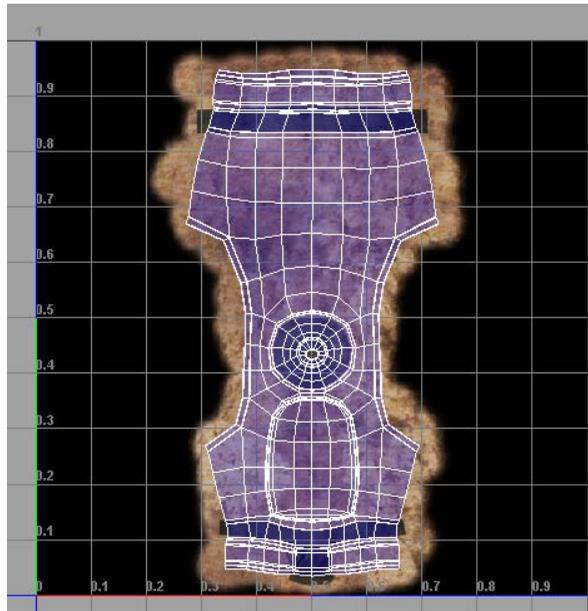
- 1 In the perspective view, right-click the torso and select Assign New Material from the pop-up menu.
- 2 The Assign New Material window appears.



- 3 Select Lambert.
The soldier's torso turns matte grey.
- 4 Open the Attribute Editor and select the lambert tab.
- 5 Change the name of the lambert material to `Mat_Torso`.
- 6 Click the box next to Color.
The Create Render Node window appears.



- 7 Click File.
- 8 In the Attribute Editor, click the file icon next to the Image Name field. A file browser appears.
- 9 Navigate to the Getting Started folder and select UVMapping/sourceimages/soldier_torso.tif. The body armor texture is applied to the soldier's body. If the texture does not appear in the scene, press '6'.
- 10 Select all the UVs and select the Smooth UV Tool icon (). Slowly unfold the UVs until they roughly match the details of the texture as shown below.



If the texture is not aligned or sized correctly you can adjust the UVs in the UV Texture Editor using the Toolbox.

- 11** In the Attribute Editor, in the Mat_torso tab, click the box next to Bump Mapping.
The Create Render Node window appears.
- 12** Click File.
A file tab appears in the Attribute Editor.
- 13** In the file tab, click the browse icon next to Image Name.
A file browser appears.
- 14** Navigate to the `Getting Started` folder and select `UVMapping/sourceimages/soldier_torso_normals.tif`.
Normal maps appear on the soldier's chest. Normal maps give the flat portions of the texture the illusion of 3D bumps.



Now you can repeat the process for the other body parts by applying the following textures:

Appendage	Material node name	Texture filename	Normals filename
Left Arm	Mat_Arm	soldier_arm.tif	soldier_arm_normals.tif
Right Arm	Mat_Arm (reuse from above by using Assign Existing Material).	soldier_arm.tif	soldier_arm_normals.tif
Legs	Mat_Legs	soldier_legs.tif	soldier_legs_normals.tif
Left Foot	Mat_Feet	soldier_foot.tif	soldier_foot_normals.tif
Right Foot	Mat_Feet (reuse from above by using Assign Existing Material).	soldier_foot.tif	soldier_foot_normals.tif
Head	Mat_Head	soldier_head.tif	soldier_head_normals.tif
Eye	Mat_Eye	soldier_eye.tif	none

NOTE When reusing a texture for a mirrored appendage (Right Arm or Right Foot), you must flip the UVs in the UV Texture Editor to fit the texture. You can do this by selecting all the UVs and clicking the Flip UVs in the U Direction button



() in the UV Texture Editor.



Beyond the lesson

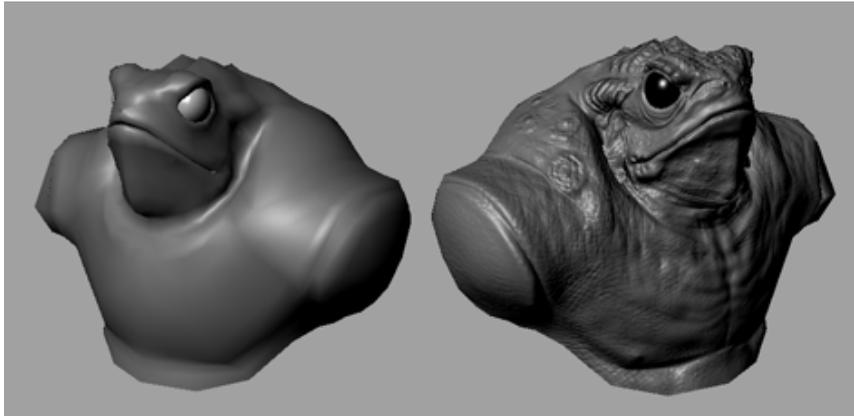
In this lesson you learned how to unfold a UV mesh so that you can create a texture for a complex organic mesh.

- You can divide a mesh into parts for texturing.
- You can create a planar mapping of UVs.

- You can apply a basic checker texture to find problem areas.
- You can cut UV edges in obscure areas to create a seam for unfolding.
- You can pin a UV and unfold a mesh in all directions or constrained to a single direction.
- You can partially unfold a UV mesh with the UV Smooth Tool.
- You can relax UVs to fix problem areas.

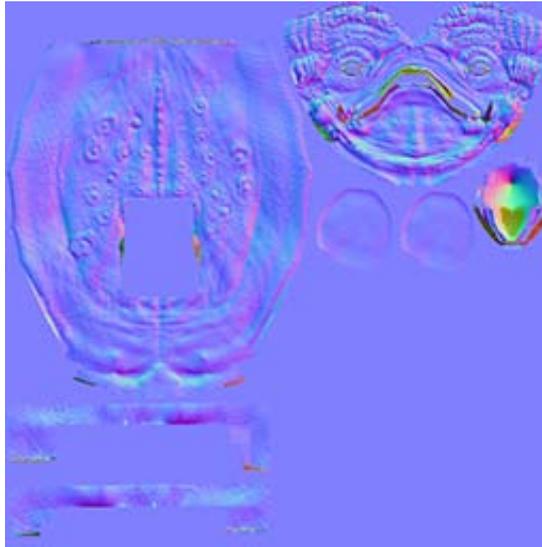
Lesson 3: Normal mapping

Introduction



Normal mapping is a technique in which you use a high resolution mesh to generate a map for a low resolution mesh. High resolution meshes are made of more complicated geometry and can be very taxing in an environment where resources are at a premium (for example, in a large scene with other high poly objects, or on a system with limited processing power). As a result, performance can suffer greatly. Normal mapping is commonly used in these situations to capture the detail of a high resolution mesh with the geometry of a low resolution model.

A normal map differs from a texture map in that it produces a multi-channel image (shown below) based on the normals of the high resolution (source) mesh. This information is used to light the mapped details on a low resolution (target) mesh convincingly, even though the surface itself is relatively flat.



In this lesson you learn how to:

- Create and apply a normal map.
- Display the poly count for a scene.
- Set the Search Envelope for a target mesh.
- Set Default Rendering mode and High Quality Rendering mode.

Lesson setup

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

- 1 Make sure you've done the steps in [Preparing for the lesson](#) on page 360.
- 2 Select the Rendering menu set.
- 3 Select Rendering > Default Quality Rendering from the panel menu.

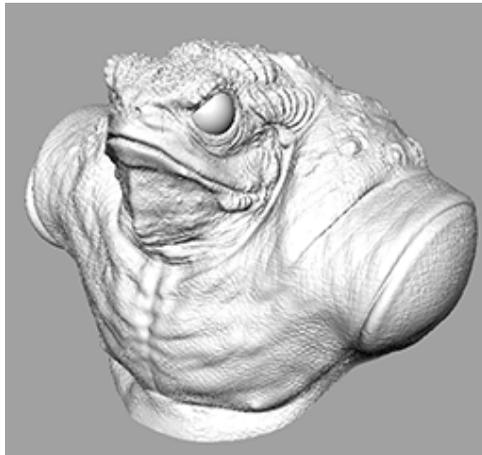
Open the scene for the lesson

To create a normal map you need two files, the source and the target. The source model is generally a higher resolution and more detailed version of the target model. The target has the geometry you actually want for your scene.

To open the high resolution model

- 1 Select File > Open Scene.
- 2 Navigate to `GettingStarted/UVMapping` and select `Toad_hi_res.mb`.
- 3 Click Open.

A high resolution polygonal toad model appears in the scene.



- 4 Tumble the view around the toad model.
Maya may respond slowly because the mesh is highly detailed. You can see exactly how detailed it is by turning on the Poly Count display.

To turn on the Poly Count display

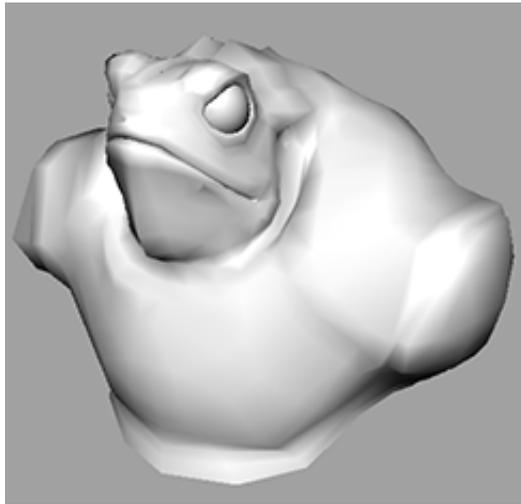
- Select Display > Heads Up Display > Poly Count.

Statistics about the scene appear in the top left corner of the main viewport. This model consists of over a hundred thousand faces just for the head and upper torso. Loading such a high resolution model into the scene is impractical for performance purposes.

Now compare the high resolution mesh with the low resolution alternative.

To import the low resolution model

- 1 Select File > Import.
- 2 Navigate to `GettingStarted/UVMapping` and select `Toad_lo_res.mb`.
- 3 Click Open.
A low resolution polygon model loads on top of the high resolution toad model.
- 4 Select the high resolution model and hide it by pressing `Ctrl + H`.



The low resolution model is much more manageable in size with just over 1500 faces. However, while it maintains the basic shape of the high resolution toad, it lacks much of the detail.

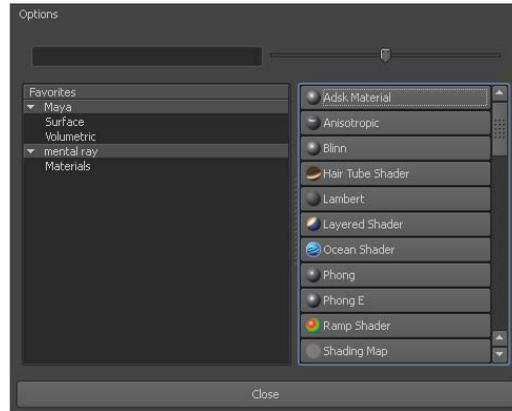
Creating a normal map

A normal map is a type of transfer map, meaning that it is a texture created from a polygon object. Different transfer maps handle the conversion from polygons to textures in different ways. A normal map captures the surface normal information of the source mesh and uses it to light the target mesh.

To set a base material for the target mesh

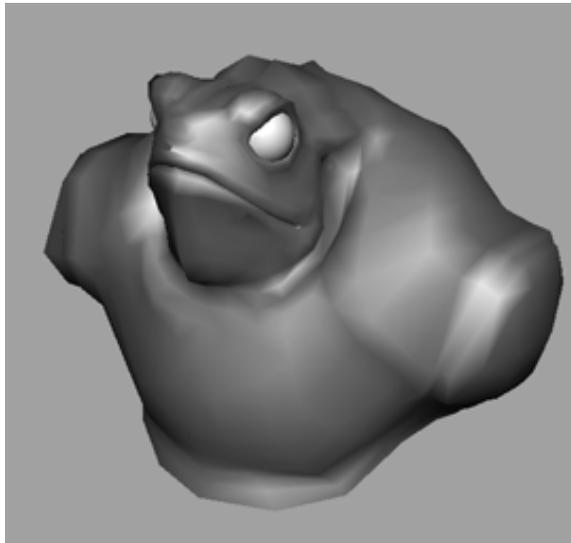
- 1 Right-click the mesh and select Assign New Material from the pop-up menu.

The Assign New Material window appears.



- 2 In the list of materials, select Blinn.

Maya applies a default blinn shader to the low resolution mesh.



- 3 Close the Attribute Editor.

To show the high resolution mesh

- 1 Open the Outliner (Window > Outliner).
- 2 In the Outliner, select hi_res_toad.
- 3 Select Display > Show > Show Selection.
Both meshes appear in the scene overlapping each other.
- 4 Deselect the high resolution mesh.

To create a normal map

- 1 Select Lighting/Shading > Transfer Maps.
The Transfer Maps window appears.
The Transfer Maps window lets you create various textures based on mappings from polygon models. First you need to assign the source and target meshes for the transfer.
- 2 In the Outliner, select lo_res_toad.
- 3 In the Transfer Maps window, in the Target Meshes section, click Add Selected.
- 4 In the Outliner, select hi_res_toad.
- 5 In the Transfer Maps window, in the Source Meshes section, click Add Selected.

Next you need to set the Search Envelope.

The Search Envelope represents the volume of the target geometry to consider for the texture. Whatever part of the source mesh between the Search Envelope and the target mesh is mapped to a texture.

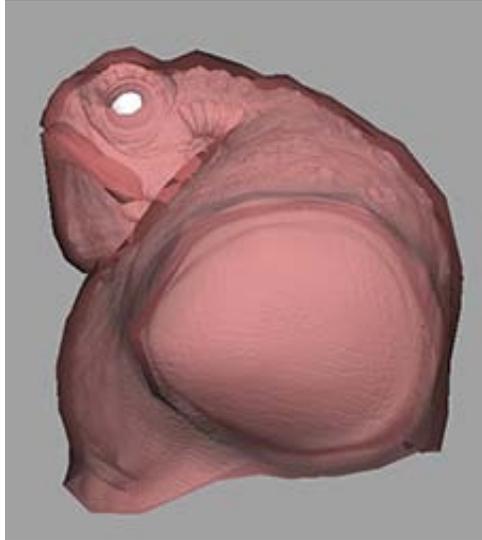
Generally you want the Search Envelope to encompass your entire source mesh so you don't miss any detail.

To set the Search Envelope

- 1 In the Transfer Maps window, in the Target Meshes section, set Display to Envelope.
The low resolution toad is replaced by a translucent red mesh.
This red mesh represents the Search Envelope.
- 2 Drag the slider under Search envelope (%) to the right.

The red mesh grows.

- 3 Drag the slider so that the red mesh barely encompasses all of the high resolution mesh without leaving any white spots (excluding the eyes). You can also enter a value in the box next to the slider. In this case, a value of 4.3 yields just enough coverage around the two meshes.

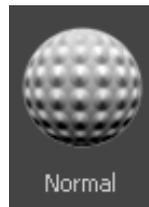


- 4 Set Display back to Mesh.

Now you can set the normal map options.

To set the normal map options

- 1 In the Transfer Maps window, under Output Maps, click Normal.



- 2 In the Normal Map text field, click the browse icon ().
A file browser appears.

- 3 Navigate to `Getting Started/UVMapping` and enter the name `toad_texture`.
- 4 Click Save.
- 5 Set File Format to DDS (dds).
- 6 In the Transfer Maps window, set the following:
 - Under Connect Output Maps, set Connect maps to Assigned shader.
 - Under Maya Common Output, set Map width and Map height to 1024 and Sampling quality to Medium (4x4).
- 7 Click Bake and Close.
Maya generates a normal map.

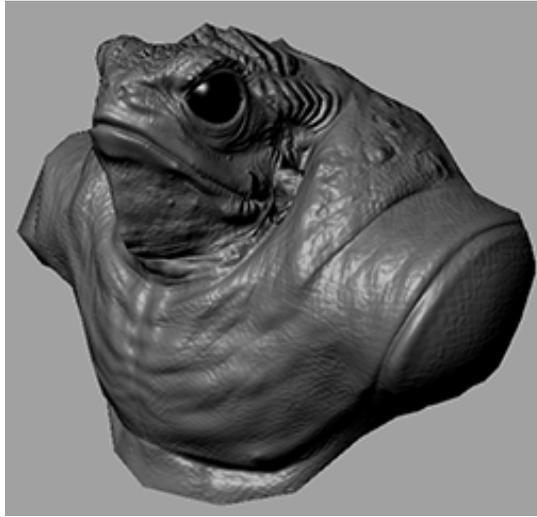
NOTE It may take a few minutes for Maya to generate the normal map.

Viewing a normal map

Now that Maya has generated the normal map, you need to change the render mode to High Quality Rendering mode to see it.

To view the normal mapped texture

- 1 Select the high resolution toad model hide it by pressing `Ctrl + H`.
- 2 In the panel menu, select `Renderer > High Quality Rendering`.
Now you can see the low resolution model with the detail of the high resolution model applied as a texture.
- 3 Tumble the view around the toad.
The light on the surface reacts appropriately to all the details on the texture, without any loss in performance.



Beyond the lesson

In this lesson you learned how to use a high resolution model to create a normal map for a low resolution model. You learned how to:

- Display the poly count for a scene.
- Set the source and target for a normal map.
- Set the Search Envelope.
- Switch between Default Quality Rendering and High Quality Rendering.

You can also try applying other types of transfer maps to a low polygon model. Each mapping feature gives you slightly different looks on your low polygon model. For more information about transfer maps, see [Lighting/Shading > Transfer Maps](#).

Introduction

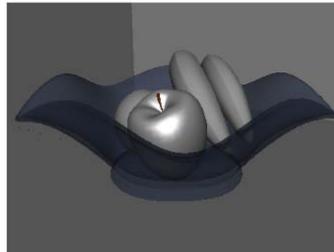
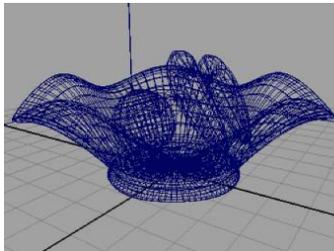


Image created by: Vivien May

In Maya, *rendering* refers to the process of creating bitmap images of your scene based on the various shading, lighting, and camera attributes that you set.

When rendering, Maya takes into account all of the various objects and scene attributes, and performs mathematical calculations to produce the final image or image sequence. Once you render a sequence of images, you can then play them back in sequence, producing an animation.

Rendering involves many components to produce an image:

- Shading materials and textures

- Lighting and shadows
- Cameras and animation
- Rendering method
- Visual effects

In Maya, rendering can be accomplished using software (Maya Software Renderer, mental ray® for Maya® Renderer, or the Maya® Vector Renderer) or hardware rendering methods. Each type has its distinct advantages. Which renderer you decide to use will be determined by your image requirements (that is, the look you require) and time constraints.

Software rendering generally allows you to create more precise results but can take longer to produce each frame or image. The Maya software render type uses a process for scene illumination that directly illuminates objects in the scene or simulates illumination by reflecting neighboring objects in the scene or via texture mapping the effect on the object.

The mental ray® for Maya® software render type additionally provides a feature called Global Illumination: a process that simulates the effect of indirectly illuminating objects that are not receiving direct illumination. It can also simulate light effects known as caustics. Hardware rendering is generally faster but is less capable of producing detailed results. In addition, some visual effects can only be produced via one method; at times, a combination of more than one type of rendering method may be required.

Rendering usually requires several iterations to achieve the final image that meets your requirements. The key is to strike the balance between producing the image that meets your requirements and producing it in the time required. This is particularly important if you are using Maya in a production setting.

This chapter introduces you to some of the fundamental concepts and skills related to rendering. This chapter includes the following lessons:

- Lesson 1 Rendering a scene: [Introduction](#) on page 439
- Lesson 2 Shading surfaces: [Introduction](#) on page 464
- Lesson 3 Lights, shadows, and cameras: [Introduction](#) on page 484
- Lesson 4 Global Illumination: [Introduction](#) on page 501
- Lesson 5 Caustics: [Introduction](#) on page 519

Preparing for the lessons

- 1 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](#) on page 25.
- 2 To ensure the lessons work as described, 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.

Lesson 1: Rendering a scene

Introduction



This lesson provides a brief introduction to rendering in Maya. In this lesson, you learn how to:

- Assign shading materials to surfaces in the scene.
- Edit the color of shading materials.
- Apply simple textures to shading materials.
- Render both a single image and a sequence of images for an animation.
- Use Interactive Photorealistic Rendering (IPR) to perform test renders of the scene.
- Edit the Render Settings attributes to change the quality of the final results of an image.
- View a sequence of rendered frames for an animation.

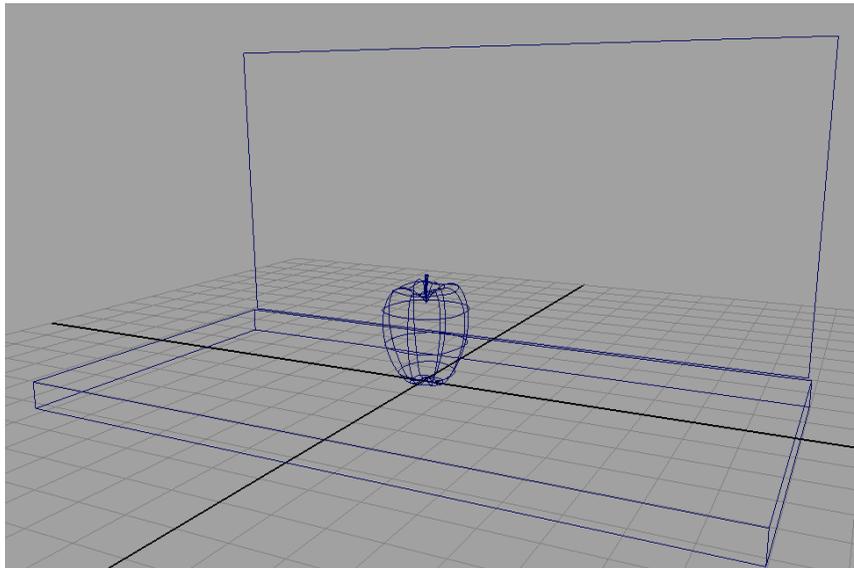
Open the scene for the lesson

In this lesson, you render a scene we've created for your use.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 439.
- 2 Open the scene file named `apple.mb`.

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

```
GettingStarted\Rendering\apple.mb
```



The scene contains an apple sitting on a plane, with another plane as the background. A camera in the scene is animated to revolve around the apple at varying distances. (As an alternative to using the scene `apple.ma`, you can create a similar scene with your own objects.)

- 3 Select `Window > Settings/Preferences > Preferences`.
- 4 Click the `Time Slider` category and make sure the `Playback Speed` option is set to `Real-time (24 fps)` so your animation will playback at the default rate of 24 frames per second.
- 5 Close the `Preferences` window.

You won't use the animated camera until later in the lesson. In the next steps, you color the apple and the planes by creating shading materials.

Creating shading materials for objects

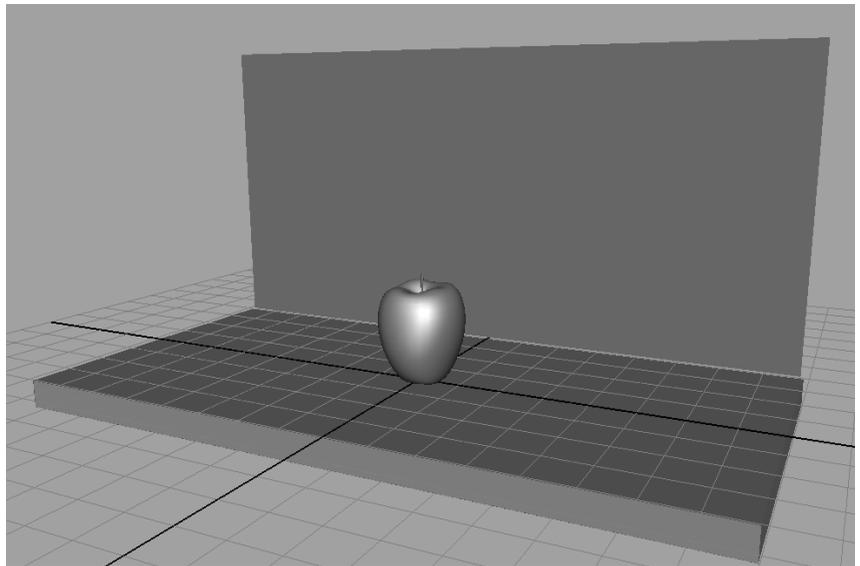
A *shading material* is a collection of attributes that you apply to a surface to control how the surface appears when it is rendered. Shading materials include such attributes as color, shininess, and texture. The following table outlines the materials you'll be working with:

Material	Description
Lambert	Creates a matte surface without specular highlights. Lambert is the default shading material.
Blinn	Creates a shiny surface with specular highlights, often used to simulate metallic surfaces.

When you open the `apple.ma` file, it initially displays the apple and the planes as wireframes (Shading > Wireframe). This is the default display mode for objects in a scene view.

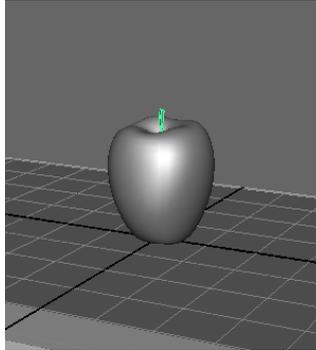
To create materials for the apple stem

- 1 In the perspective view, select Shading > Smooth Shade All (hotkey: 5).



The apple and the other surfaces appear with a default gray, smooth shading material, lit using default lighting. Default lighting occurs when no other lights have been placed in the scene.

- 2 In the perspective view, select the apple stem by clicking its geometry. (Be sure that only the apple stem is selected.)



- 3 From the Rendering menu set, select Lighting/shading > Assign New Material.

The Assign New Material window appears.

- 4 Click Lambert.

The Attribute Editor appears for the Lambert material.

A Lambert material creates a matte surface without shiny highlights. You will learn more about materials and their attributes in a later lesson.

- 5 Click the gray box to the right of Color.



The Color Chooser window appears.

- 6 Click inside the color wheel and drag to a brown color.

The color you select in the Color Chooser is applied to the apple stem.



Move your mouse off the Color Chooser to exit.

- 7 In the Attribute Editor, rename the lambert shading material to `appleStem_lambert`.

To create materials for the apple

- 1 Right-click the apple and select Assign New Material from the marking menu.

The Assign New Material window appears.

- 2 Select Blinn.

The Attribute Editor updates to display the attributes for the Blinn material.

A Blinn material gives the apple a shiny appearance.

- 3 Click the gray box to the right of Color.

The Color Chooser window appears.

- 4 Click inside the color wheel and drag to a red color.

The apple's surface updates to the same color you select in the Color Chooser.

- 5 In the Attribute Editor, rename the Blinn shading material to `appleSkin_blinn`.

To create materials for the counter and wall

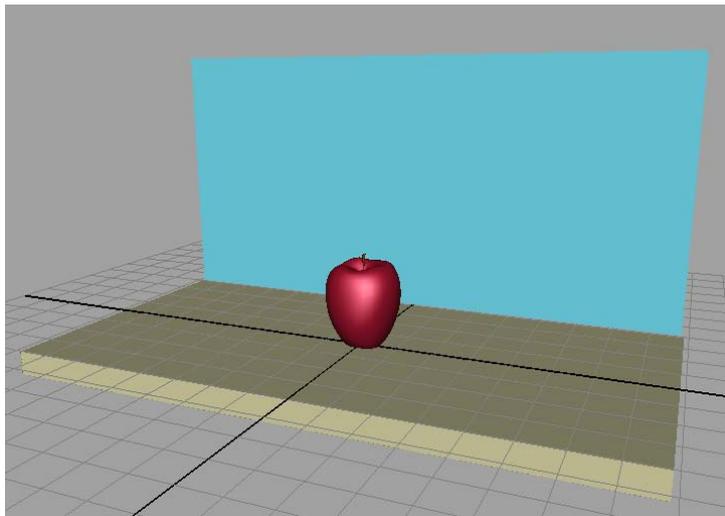
- 1 Right-click the background plane and select Assign New Material from the marking menu.

The Assign New Material window appears.

- 2 Select Lambert.

The Attribute Editor displays the attributes for the Lambert material.

- 3 Click the gray box to the right of Color.
- 4 In the Color Chooser window, click inside the color wheel and drag to a light blue color.
The color you select in the Color Chooser is applied to the plane.
- 5 In the Attribute Editor, rename the Lambert shading material to wall_lambert.
- 6 Right-click the surface that the apple is resting on and select Assign New Material > Lambert.
The Attribute Editor displays the attributes for the Lambert material.
- 7 Click the gray box to the right of Color.
- 8 In the Color Chooser window, click inside the color wheel and drag to a beige color.
The color you select in the Color Chooser is applied to the plane.
- 9 In the Attribute Editor, rename the Lambert shading material to counter_lambert.



The counter, wall, and apple stem all have a matte Lambert surface, while the apple has a shiny Blinn surface. There are many other attributes you can set to change the appearance of these objects, which you'll explore in the next section.

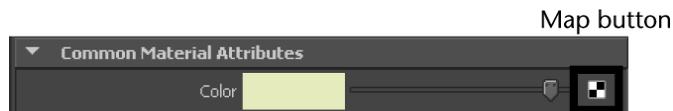
Refining shading materials for objects

You can refine the shading materials in the scene to make the objects appear more realistic. In the next steps you connect (or apply) a *procedural texture* to the counter, and a *ramp texture* to the apple. These textures are connected to the color attribute of the shading material for each surface to provide natural surface variation. When you connect textures, you create *shading networks*. You'll learn more about shading networks in a later lesson.

A procedural texture is a 2D or 3D texture calculated based on an algorithm or mathematical formula. It can be useful for creating random patterning. A ramp texture is a 2D texture or color gradient in which the color or grayscale value changes across the image.

To connect a procedural texture to the counter surface

- 1 In the perspective view, right-click the counter surface and select Material Attributes.
The Attribute Editor for the counter_lambert material appears.
- 2 To the right of the Color slider, click the Map button



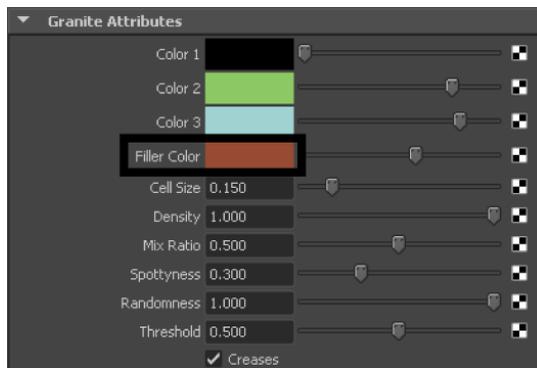
The Create Render Node window appears and lists the various textures you can connect to the Color attribute of the Lambert material assigned to the counter surface.

- 3 In the Create Render Node window, select Maya > 3D Textures > Granite.



The procedural texture gives the counter the look of a stone granite counter top when it is rendered, and provides the context for the scene.

- 4 Press 6 to see a shaded and textured display of the granite texture on the counter top surface.
- 5 In the Attribute Editor, under Granite Attributes change the Filler Color from the default reddish-brown to a beige color as follows:
 - Click the color box to the right of Filler Color.



- In the Color Chooser window, select a beige color.

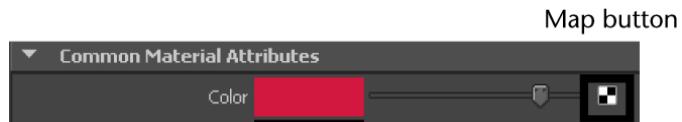
NOTE Materials and textures displayed in the perspective view do not show their true appearance. To see the accurate appearance of the texture you must render the scene.

To connect a ramp texture to the apple surface

- 1 Right-click the apple and select Material Attributes from the marking menu.

The Attribute Editor updates to display the appleSkin_blinn material properties.

- 2 To the right of the Color slider, click the Map button.



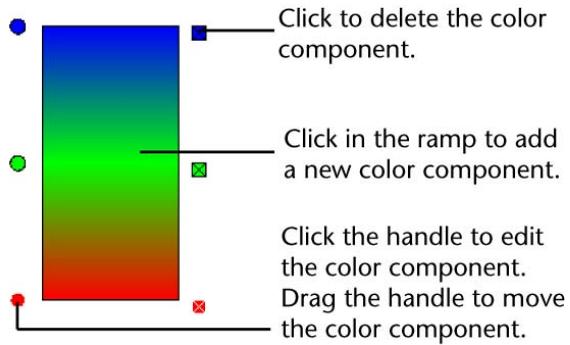
The Create Render Node window appears and lists the various texture patterns you can connect to the Color attribute of the Blinn material assigned to the apple.

- 3 Select Maya > 2D textures > Ramp.

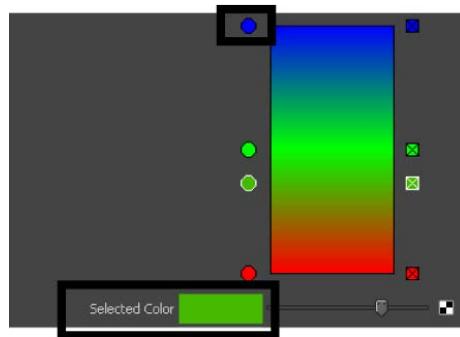


This texture creates a gradation through a series of selected colors. Applying the ramp texture provides the apple with a realistic variation in surface color.

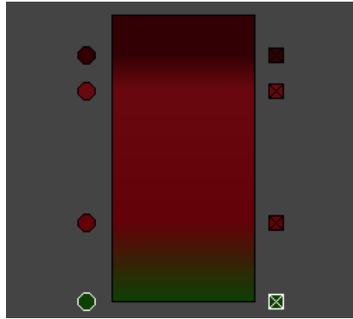
- 4 In the Attribute Editor, under Ramp Attributes, change the Type to U Ramp.
The Ramp Type indicates the direction of the color gradient. A U ramp applies the gradient in a linear, vertical direction across the object.
- 5 Add another color component by clicking within the ramp.
A new set of color handles appears in the ramp, for a total of four color components.



- 6 To change the ramp colors, click the top circular color handle (to indicate which color you want to modify), then click the color box to the right of Selected Color to display the Color Chooser window.



- 7 Click inside the color wheel and drag to a very dark brown for the first ramp color.
- 8 Repeat steps 6 and 7 for the other three ramp colors. For the two middle ramp colors, select a dark red. For the last ramp color, select a deep olive green.
- 9 Use the circular color handles to reposition each component as shown below:



The position of these handles and the corresponding colors affect the location and breadth of the color within the gradient on the apple's surface.

10 Save your work.

In the next section you render your scene to view the textures and materials you've assigned to the objects.

Maya renderers

So far, you've seen the results of shading the apple and other surfaces in the scene view. In this view, Maya uses your computer's graphics hardware to display the shading and textures quickly but with low quality.

To view the shading results of the colors and textures in a more realistic fashion, you must use a renderer. In this lesson you use a process called *mental ray* for Maya rendering.

mental ray rendering can take seconds or minutes to render a single frame of animation from your scene, depending on the complexity of surface geometry, shaders, lighting and other visual elements present in the scene.

The following table outlines the different types of renderers in Maya and what each is used for:

Renderer	Use
mental ray® for Maya® renderer	A general purpose renderer that includes exclusive, advanced rendering functionality, such as host and network parallel rendering, area light sources for soft shadows,

Renderer	Use
Maya's Software renderer	global illumination, and caustics (light patterns).
Interactive Photorealistic Rendering (IPR)	A feature of Maya's software renderer and mental ray for Maya renderer, used to make interactive adjustments to the final rendered image. You can adjust shading and lighting attributes in real-time, and IPR automatically updates the rendered image to show the effects of your changes. IPR is useful for tweaking an image before rendering to disk.
Maya Vector renderer	A specialized renderer used to produce stylized renderings (for example, cartoon, tonal art, line art, hidden line, wireframe) in various bitmap image formats (IFF, TIFF) or in 2D vector formats (SWF, AI, EPS, SVG). The Maya Vector renderer is often used to render web-ready images.
Maya's Hardware renderer	A general purpose renderer that uses your machine's graphics card for computation. You can produce broadcast resolution images in less time than with software rendering, and in some cases, the quality may be good enough for final delivery.

The Render View window

When you render your scene the rendered image appears in its own window called the *Render View*. By default, the Render View uses the same camera as the Scene View (persp), but includes particular rendering capabilities.

The following table shows the differences between the Render View and the Scene View:

	Scene View	Render View
Display	3D object scene	2D rendered image
Surfaces	modeled surfaces, the grid, vertices, curves, and object manipulators	shaded surfaces only
Background	default gray background	black background by default because only objects with materials that are lit can be seen
Quality	low-quality, colors and textures do not appear in their final display form	high-quality, colors and textures appear in their final rendered form

Rendering a single frame using IPR

In this section, you work with IPR (Interactive Photorealistic Rendering) for mental ray rendering.

The advantage of rendering with IPR is that, after you render an image, you can adjust the shading and lighting attributes of the scene and then quickly update part or all of the rendered image to review the effect of your changes. IPR is useful for iterations when you need to fine-tune specific attributes. Updates to the image are not always instantaneous, but they are full quality.

Using the mental ray for Maya renderer

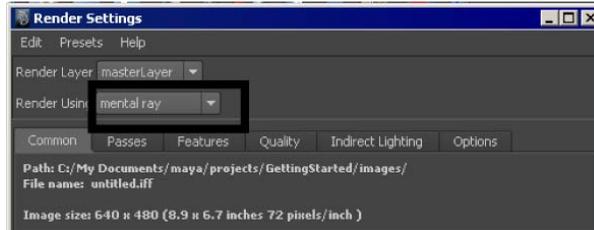
To use IPR for mental ray rendering, you must first select the mental ray for Maya renderer.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.



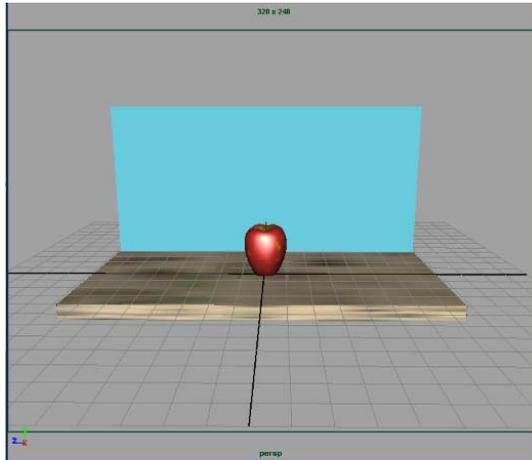
To render the scene using IPR

- 1 In the perspective view, select View > Camera Attribute Editor to display the attributes for the camera.

You need to specify the exact region of the perspective view you want to render.

- 2 In the Attribute Editor, open the Display Options section and turn on Display Resolution.

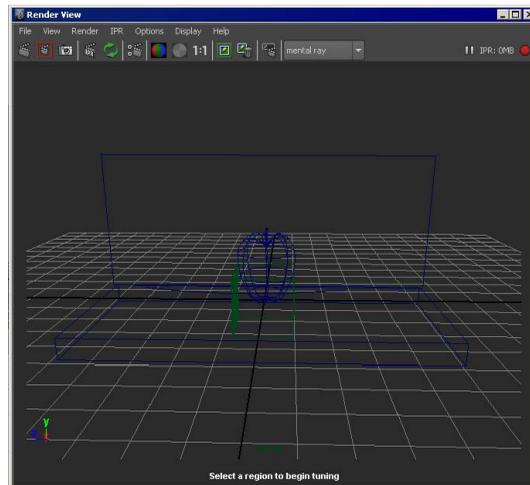
A dark green rectangular border shows the region to be rendered. This region defines the area of the final image, and allows you to decide what appears or does not appear in the final image. (The left and right borders of the rectangle may be positioned at the edge of the window.)



Dolly the view for a closer view

- 3 Select Render > IPR Render Current Frame (or select the IPR render current frame button on the Status Line).

The Render View appears as follows.

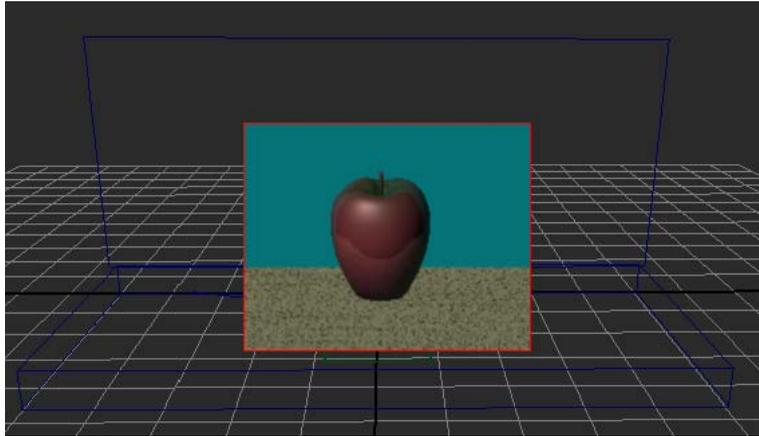


The message displayed at the bottom of the IPR Render View window states "Select a region to begin tuning". In the next step, you create a subregion of the image to render using IPR.

- 4 In the Render View, select View > Frame Image to resize the Render View window so it fits closely to the rendered image.

To select a subregion to render using IPR

- 1 In the Render View, drag a selection box around the small area surrounding the apple, counter, and wall as shown:

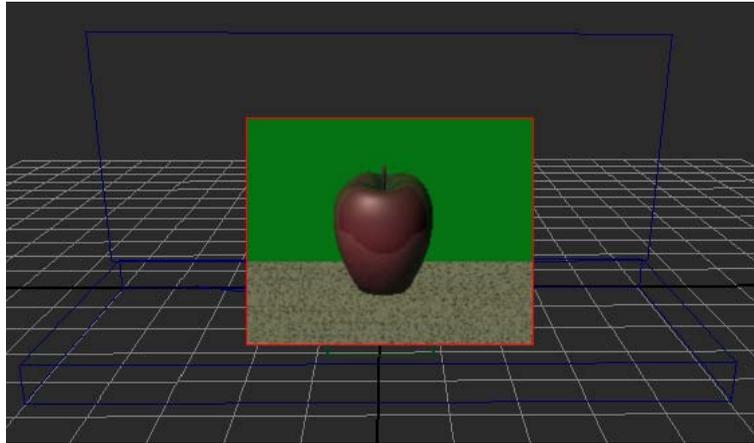


When using IPR, this selected region updates as you alter any shading, texture, or lighting attributes. Selecting a smaller region quickens the update. You can select regions throughout the image to see those areas update.

- 2 In the scene view, right-click the wall and select Material Attributes to display the Attribute Editor for the Lambert shader you applied to the wall.
- 3 Click the Color box and change the color to green.

The area within the selected region of the Render View updates immediately as you alter the color; the wall turns green.

When you drag within the color wheel of the Color Chooser window, the selected region updates occur immediately.



NOTE If the region does not update, in the Render Settings window, turn the Raytracing option off.

(You won't alter any other attributes in this lesson. The purpose of the lesson is to learn the process of rendering, not to perfect the results at this point.)

So far, you've used IPR to see changes you made to a shader. If you add lights to the scene rather than using default lighting, you can also use IPR to help adjust the lighting.

Rendering using the mental ray for Maya renderer

You are nearly ready to render the entire animation to disk. You must first switch camera views so you can render the scene from the point of view of the camera that was animated for this lesson. When animating a camera it's a good practice to create a second camera to animate, rather than animating the default persp camera. In that way you can return to the default perspective view should it become necessary.

Rendering the entire animation can take a lot of time, so it is useful to first use the mental ray for Maya renderer to test render a few frames to your monitor and examine the results.

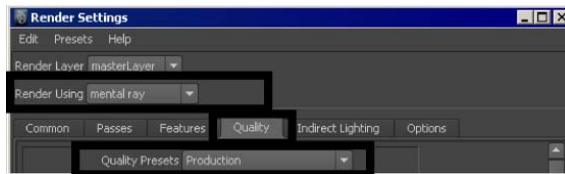
Once you are satisfied with the quality of a few rendered frames, you can render the entire animation to disk.

To switch camera views and playback the animation

- 1 In the perspective view, select Panels > Perspective > apple_camera.
The view updates to display the scene as seen through the lens of the apple_camera.
- 2 Click the Play button in the Timeslider playback controls.
During playback, the animated camera tracks and tumbles about the apple.
- 3 Stop the animation at a frame you want to test render.

To test render the current frame of animation

- 1 In the Render View window, select Options > Render Settings.
- 2 In the Render Settings window, ensure that mental ray is selected under the Render Using drop-down list.
- 3 In the Render Settings window, select the Quality tab.
- 4 Select Production from the Quality Presets drop-down list.



This setting provides high quality results by smoothing any jagged surface edges in the rendered image.

- 5 Close the Render Settings window.
- 6 In the Render View window, select Render > Render > apple_camera.
This menu item uses the mental ray renderer to render a single frame of the scene that you rendered with IPR previously but from the point of view of the animated apple_camera.
- 7 Go to frame 120 of the animation, where the apple is viewed in a different position.
- 8 In the Render View, select Render > Render > Current (apple_camera).
Ensure that the rendered image looks good at this frame also.

Now that you've checked a pair of rendered frames that represent the entire animation, you are ready to render the entire animation sequence. In your own projects, you might want to test render several frames, particularly where new shading and lighting elements occur in the animation.

Batch rendering a sequence of animation frames

After you model, animate, and color your scene, you set several Render Settings options and then use the mental ray renderer to *batch render* part or all of the animation's range of frames to files on disk. Each file represents a single frame (image) of the animation.

To set the Render Settings for batch rendering

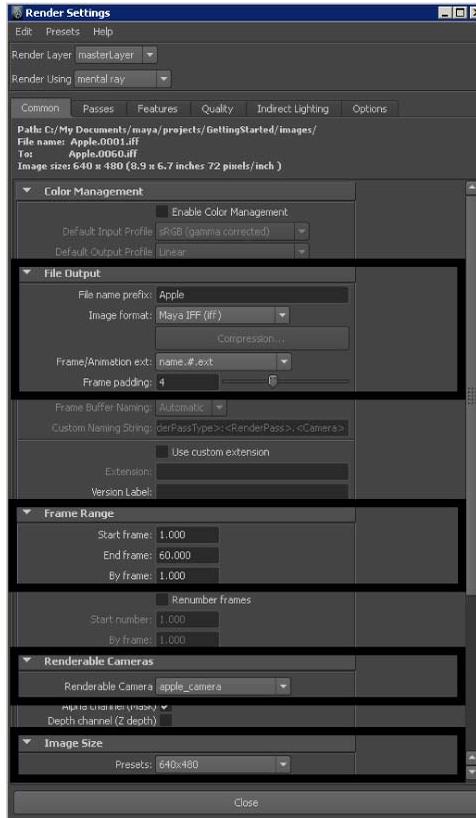
- 1 In the Render View window, select Options > Render Settings to display the Render Settings window.
- 2 In the Render Settings window, select the Common tab.
- 3 In the File Output section, set the following options:
 - **File Name Prefix:** Type the name `Apple`. This name will be the base of the filenames created by batch rendering.
 - **Image Format:** Select Maya IFF (.iff), Maya's standard image file format. You can use the `.iff` format for any further work you need to do, including previewing and compositing the animation. If you require a different format, you can specify it instead of `.iff` in the Render Settings.
 - **Frame/Animation ext:** Select `name.#.ext`. This specifies that the filenames will have the format *prefix.frameNumber.fileFormat*. For example, batch rendering the entire 200-frame animation will create `Apple.0001.iff`, `Apple.0002.iff`, and so on through `Apple.00200.iff`.
 - **Frame padding:** Enter 4. This causes the *frameNumber* part of the filenames to be four digits prefixed with 0s. For example, the filenames will be `Apple.0001.iff` through `Apple.0050.iff`.

The four-digit padded filename is compatible with many image playback programs, for example, Maya's FCheck utility. Image playback programs let you view rendered animation sequences on your monitor at real-time speed.

- 4 In the Frame Range section, set the following options:
 - Start frame: Enter 1, the first frame of the animation sequence to be batch rendered.
 - End frame: Enter 60, the last frame to be batch rendered. (Rendering all 200 frames may be time-consuming.)
- 5 In the Renderable Cameras section, set the following option:
 - Renderable Camera: Select `apple_camera` from the drop-down list to indicate which camera view to render.
- 6 In the Image Size section, select 640x480 from the Presets drop-down list.

For the remaining options in the Render Settings window, you'll use the default settings. Maya will render using the quality preset (Production) that you specified earlier in the lesson.

After you set the Render Settings, the top portion of the Common tab shows the correct path and filenames for the files to be created during batch rendering. Check that this information is correct.



7 Close the Render Settings window.

To batch render animation frames

1 Save the scene.

It's a good practice to save the scene before batch rendering. This is useful if, after batch rendering, you need to change any display settings and render again. By saving the scene prior to batch rendering, you can examine the scene to learn which option settings were in effect at the time you batch rendered.

2 From the Rendering menu set, select Render > Batch Render.

The Batch Render Frame window appears.

Batch rendering 60 frames of a simple scene takes a few minutes. A complex scene may take hours per frame, depending on the speed of your computer.

To check the status of the batch render

- 1 While Maya is rendering, select Window > General Editors > Script Editor. Expand the size of the Script Editor window. The window shows a completion log for the frames being rendered.

Maya puts the resulting files in a default *images* directory. The files have the following names:

```
Apple.0001.iff  
Apple.0002.iff  
Apple.0003.iff  
...  
...  
Apple.0050.iff
```

The images directory is located in the same path as the scenes directory. You can have Maya save to a different path by changing the project setting. See the Maya Help for further information on projects.

- 2 Close the Script Editor when the following message appears:

```
// Result: Rendering Completed. See mayaRenderLog.txt for  
information. //
```

The `mayaRenderLog.txt` file contains rendering statistics for advanced users.

Viewing a sequence of rendered frames

You can use Maya's *FCheck* image viewing utility to view the batch-rendered images at a speed that approximates video or film.

To use FCheck to view a sequence of rendered frames

- 1 Select File > View Sequence.
The File Browser window appears.
- 2 Using the File Browser, navigate to the images directory for your current project (or wherever you saved the rendered sequence of images for the Apple).

- 3 Click the image file `Apple.0001.iff` to select the first image in the sequence and then click **Open**.
The FCheck image viewing utility appears and the rendered sequence of 60 frames plays back as an animated loop.
- 4 Close the window after viewing the animation.

NOTE The FCheck rendering utility has playback, display, and other options you can select from your keyboard. Details are available from the FCheck Help menu.

Beyond the lesson

In this lesson, you were introduced to some of the basic concepts for rendering an image. In this lesson you learned that:

- Surfaces are assigned shading materials to provide basic attributes such as color, shininess, reflectivity, and so on. Each shading material has unique characteristics that can be modified to suit your requirements.
- You can additionally assign a texture map to some of the shading material attributes so that your objects appear more realistic.
- Maya provides several methods for rendering an image. You can render using the mental ray® for Maya® renderer, the Maya Software renderer, the Maya Hardware renderer, or the Maya Vector renderer. Each type of renderer has specific advantages and uses.
- You can choose the size and quality for a rendered image prior to rendering based on your specific needs and requirements.
- You can render a single image or a sequence of images from an animation.
- You can view one image, or a sequence of rendered frames using the FCheck utility.

As you render scenes, consider the following issues:

Batch rendering

In this lesson, you used the mental ray for Maya renderer to check the image quality of two frames before you started to batch render. When you create scenes with sophisticated animation, it's useful to batch render with low-quality resolution to check the animation accuracy before batch rendering

with production-quality resolution. For instance, you might preview your animation with the frames resulting from batch rendering at a small image size (320 by 240) with Preview as your Quality Preset.

You don't need to use the batch renderer to render single frames from your scene to disk. From the Render View window, select File > Save Image.

Hardware rendering

Hardware rendering leverages the power provided by hardware graphics cards to render your images. The benefits of hardware rendering include the ability to batch render frames more quickly than with software rendering, and rendering specific particle effects not possible through software rendering. For more information about hardware rendering particles, see [Rendering particles](#). In some cases, the image quality may be good enough for final use. You access the hardware renderer by selecting Render > Render Using > Maya Hardware.

Hardware rendering has its own limitations when compared to software rendering. For more information on hardware rendering, see the [Maya Help](#).

Using render passes

It's often useful to render attributes of your scene in different passes and combine them using compositing software. For example, you can render each of diffuse, shadow, reflection, specular, and ambient occlusion in a separate pass and then composite the passes afterwards. You can select a subset of the objects or lights in your scene to contribute to each render pass. You can render an unlimited number of render passes, and then group them into logical groups called render pass sets. For more information about render passes, see [Multi-render passes](#) in the [Maya Help](#).

Lesson 2: Shading surfaces

Introduction



Images courtesy The Art of Maya

In the real world, objects are seen in specific ways based on the following:

- The materials they are made of.
- Their surface textures.
- How they are lit and reflect light.
- The environment surrounding them.

In Maya, you provide your surfaces and objects with these characteristics by assigning shading materials to them.

Shading Materials provide instructions to the renderer so it can simulate how the surfaces in your scene react to light and appear in the final image. Shading involves such attributes as color, transparency, shininess, and many others to create a realistic look.

Editing the material attributes associated with shading materials affects how they appear in the rendered image. In this way, Maya provides the option of

allowing you to create images as they would appear in the real world or in your imagination.

In this lesson, you learn to use basic shading features to render a sphere so it appears like an orange in the final image.

In this lesson you learn how to:

- Assign a shading material with specific color and surface attributes.
- Differentiate between shading material types (Lambert, Blinn, Phong, and so on) to achieve a specific material appearance.
- Assign a label texture map to a surface.
- Use the Hypershade editor to view and work with shading materials.
- Create a bump texture map to simulate the orange's pebbled skin.
- Control the placement of a 3D texture using a 3D manipulator.

Open the scene for the lesson

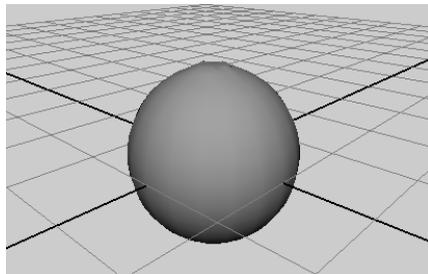
In this lesson, you work with a scene we created for your use.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 439.
- 2 Open the scene file named `ShadeOrange.mb`.

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

```
GettingStarted\Rendering\ShadeOrange.mb
```

This scene contains a sphere modeled to resemble an orange.



The scene's perspective view also has two settings turned on—Shading > Smooth Shade All and Shading > Hardware Texturing (Hotkey: 6)—which let you see the shading changes you are about to make.

Assigning a shading material

In the next steps, you create a material and assign it to the surface. A material is a collection of attributes that define color, shininess, and other surface characteristics.

To assign a material to an object

- 1 Before you create a new material, right-click the sphere and select Material Attributes.

The default material attributes appear in the Attribute Editor.

The default material assigned to the sphere (and all other surfaces) is a type of material called Lambert. The material displays the surface with gray shading. You should not edit the default shading material.

- 2 Right-click the sphere and select Assign New Material.

The Assign New Material window opens.

- 3 Click Blinn to create a new material for the sphere.

The new material's attributes appear in the Attribute Editor.

A Blinn material gives a surface a shinier look than a Lambert material. Blinn is a good choice for creating a variety of surface appearances.

Knowing which material type to use for a particular surface comes with practice. Fortunately, you can change the material type later on, so you do not need to make a decision now. You will learn more about material types later in the lesson.

- 4 In the Attribute Editor, click the gray box to the right of Color to open the Color Chooser.



- 5 In the Color Chooser, click inside the color wheel and drag the pointer to an orange-colored area, as illustrated.

The sphere becomes the same color you select in the Color Chooser.



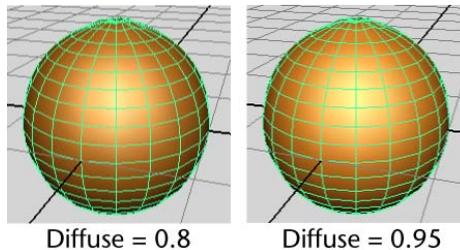
Move your mouse off the Color Chooser to exit.

If you compare the color in the Color attribute box with the sphere's color in the scene view, you'll notice a difference.

The color you assigned is only one aspect of surface appearance. The appearance also depends on how the surface reacts with light. The main attributes that control the reaction to light are under the Common Material Attributes heading. As an example of these attributes, you will change the Diffuse setting.

- 6 In the Attribute Editor, drag the Diffuse slider to its right to increase the color brightness as viewed in the scene view.

This will help create the illusion of a bright orange.



In practical usage, the Diffuse setting makes a surface appear brighter. The name diffuse comes from a property in physics that describes how light spreads after it strikes a surface. A surface with a high diffuse property spreads the light that strikes it, making the object appear brighter.

Like the Diffuse attribute, the other Common Material Attributes also refer to general characteristics of light and the way a surface reacts to light.

The next section in the Attribute Editor, Specular Shading, contains attributes that control the appearance of shiny highlights on the surface.

Next, you'll alter the surface specularity—the amount of light that reflects off a surface, resulting in shiny highlights.

Using the mental ray for Maya renderer

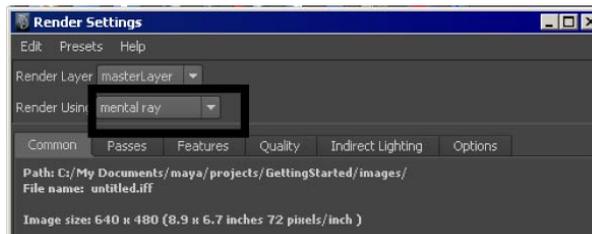
In this tutorial, you will use IPR for mental ray rendering to preview your render results. You must first select the mental ray for Maya renderer.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.



Modifying surface specularity

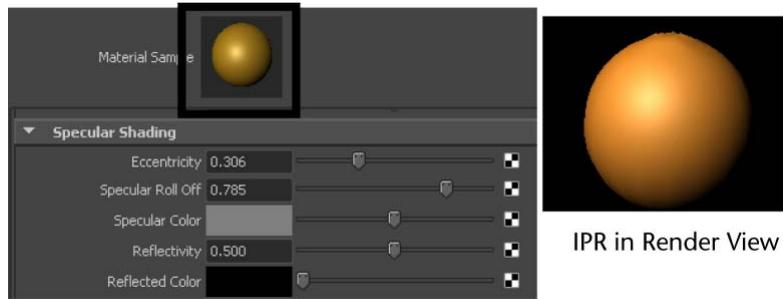
When light reflects in a concentrated manner off of a surface, a specular highlight results. Specular highlights can be thought of as “pseudo” reflections of bright areas in the scene (sunlight, bright lights, and so on). Modifying the specular attributes affects the perceived reflectivity for a surface. When you reduce the specularity, you in turn reduce the perceived reflectivity.

To modify the specularity of a material

- 1 In the Specular Shading section of the Attribute Editor, move the Eccentricity slider back and forth to change the size of the shiny highlights.

- 2 Move the Specular Roll Off slider to change the blend between highlighted and other areas of the surface.

Changes to this attribute do not alter the surface in the scene view, so you need to check the Material Sample in the Attribute Editor (see the following illustration). You can also render a test image (Render > IPR Render Current Frame).



Attribute Editor

As you work with shading, you'll find many attributes that have no effect in the scene view, for instance, Ambient Color and Translucence. Generally, the scene view is a rough approximation of how the scene will render.

The IPR renderer is ideal for testing shading. It shows the results of all changes you make to shading and shows them immediately. See the next step for details.

Some attribute changes such as reflectivity and reflections are not displayed by the Material Sample.

Material types

A material type (also referred to as a shading model) defines how a particular shading material on a surface simulates a natural reaction to light. Blinn, Phong, and Lambert are a few examples of material types available in Maya. Each of these material types provide distinct shading characteristics based on the mathematical algorithms that define them.

From the settings you've made so far, you may decide you want to give the orange a different look than possible with the Blinn material. You will now change to a different material for comparison.

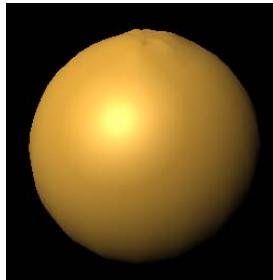
To set the material type

- 1 Next to the Type attribute near the top of the Attribute Editor, pull down the menu and select Phong E.

Phong and Phong E are variations of the Lambert material type, with specular highlights added. The main difference between materials is the way each handles specularity. By switching between the material types, you can compare which material gives you the specularity you need.

Your settings for the Color and Diffuse attributes are maintained because they are common to all material types. The Specular Shading and all other attribute settings do not convert when you change the material, so it's best to choose your material before you make many changes.

- 2 In the Specular Shading section, you can leave Roughness and Highlight Size at the default settings. The default highlight settings give the orange a waxy look.



Rendered image

These attributes are similar to the Blinn attributes, Eccentricity and Specular Roll Off. Unlike the Blinn counterparts, these highlights do not appear in the scene view. You can see the highlights only in a rendered image and in the Material Sample.

- 3 The Phong E material type has a Whiteness attribute not available with Blinn. Change Whiteness to a light gray to accentuate the highlights. This change is subtle, so drag the slider back and forth while watching the effect in the Material Sample or in an IPR region. Look for the entire highlighted area to become whiter. Again, a strong highlight will create a waxy look.

Assigning textures

For an orange, one color may be all that you need. In this section, you will go a step further by making a logo stamp appear on the orange skin. For this and other color effects, you need to apply a texture to your material.

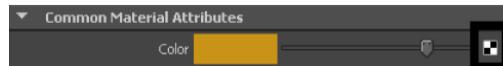
In visual arts, a texture is any kind of surface detail, both visual and tactile. In Maya, a texture is a collection of attributes that creates surface detail. Textures have a more specialized display purpose than materials. For example, you can use textures to create the appearance of a marbled pattern, bumps, or a logo image on the side of a can.

In the Maya Help, a texture is also referred to as a texture node. A *node* is a collection of attributes (or actions) with a common purpose. A shader is sometimes called a shader node or material node.

To display a texture on a surface, you apply a texture node to an attribute of the surface material. In the next steps, you apply a texture to the Color attribute of the Phong E material.

To apply a texture to a material

- 1 Click the Map button next to the Color slider. This is a quick way of applying a texture to an attribute. This is also called mapping a texture.



The *Create Render Node* window displays a list of the textures you can apply. The texture names and their icons help you to decide which texture is appropriate for the effect you want.



To create the logo stamp effect, you will use the Maya > 2D Textures > File texture. A File texture uses an image file that you or someone else created separately, such as a drawing from a paint program. For this lesson, you'll use an image we created for you.

- 2 Before you create the texture, in the 2D Textures list, right-click the File texture and select Create texture.

This option means Maya will apply the texture normally and stretch the texture evenly around the surface. The other settings—Create as projection and Create as stencil — are other ways you can apply a texture. You will learn about these settings at the end of the lesson.

- 3 The file1 node appears in the Attribute Editor.

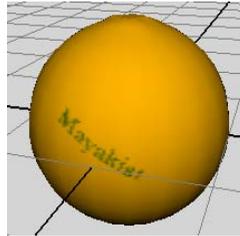
The file texture also causes the sphere to turn white in the scene view and black in a rendered view. This is a temporary color until you specify the actual file name of the file texture.

- 4 Click the folder icon to the right of the Image Name attribute. A browser window opens.
- 5 Navigate to the `GettingStarted\Rendering\sourceimages` directory and open `Render_Mayakist.iff`.

Maya looks for source images in a default sourceimages directory. However, Maya can use file texture images anywhere on your workstation

or local network. For example, you could have them on a central disk drive that is shared among your coworkers.

The file texture with a Mayakist logo now appears on the sphere. The logo is somewhat jagged, which will simulate the look of an ink stamp on rough orange skin.



- 6 In the Attribute Editor, click the place2dTexture1 tab.



The place2dTexture1 tab is a node with attributes that control the texture's position on the surface.

If you select another object and the Attribute Editor no longer displays the file texture attributes, it's easy to restore the display. Right-click the sphere, select Material Attributes, and click the texture map icon next to the Color attribute.

- 7 As an example of the place2dTexture1 placement controls, move the Rotate UV slider slightly to the right, to a value about 5.9. This rotates the Mayakist logo on the surface.

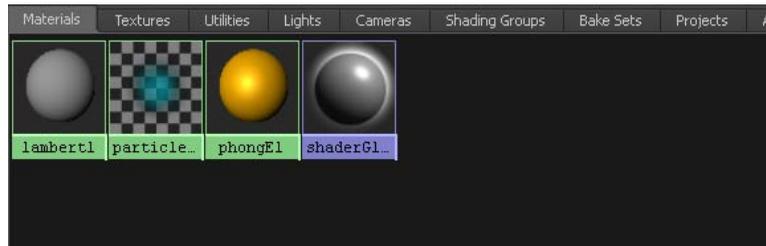
In this case, the texture placement needs almost no adjustment. For the ease of this lesson, we created a file texture that conforms to the sphere's shape by default. In practice, placing file textures requires more adjustment. For details, see [Preparing for the lessons](#) on page 439.

Using the Hypershade editor

After you create a texture node, you will often want to return to the associated material node to make changes. Because material and texture nodes do not have icons that represent them in your scene view, you need a special editor, called Hypershade, to select them and edit their attributes. In this section, you learn how to work with Hypershade.

To view and edit material attributes using Hypershade

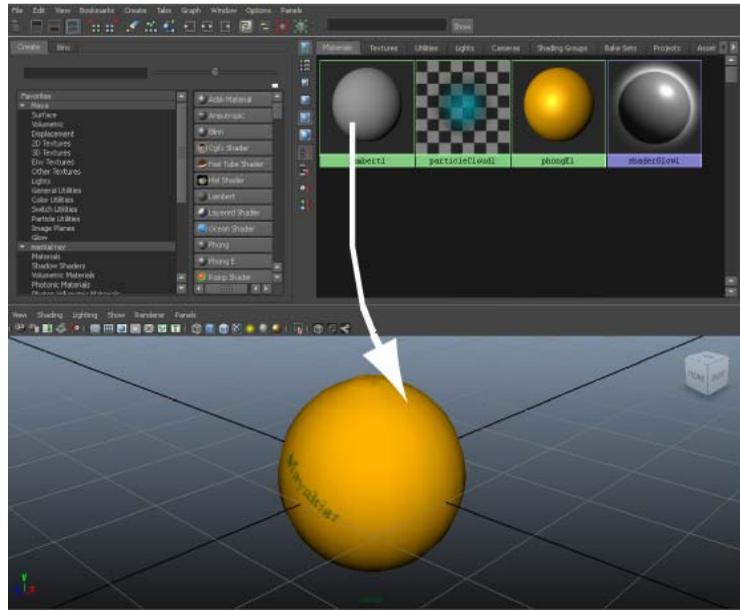
- 1 In the perspective view, select Panels > Saved Layouts > Hypershade/Persp. This layout is convenient because it's common practice to drag and drop items from Hypershade (using the middle mouse button) onto objects in the perspective view. You can also open Hypershade as a separate window (Window > Rendering Editors > Hypershade).
- 2 At the top of Hypershade, click the Materials tab to make sure the tab is displayed. If the material swatches are too small to see clearly, dolly into the view using the same keyboard and mouse shortcuts you would use in the scene view.



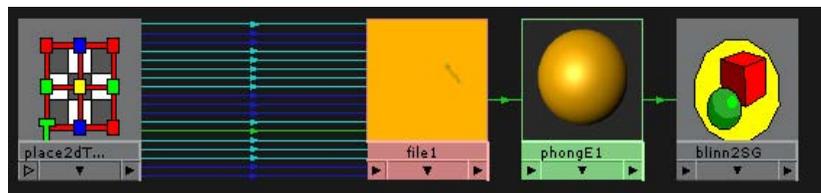
This tab shows all the materials currently in your scene. It is especially useful when you have a complex scene and you need to locate a material to edit it. The materials you will recognize are the default Lambert material and the Phong E material. You can ignore the others. They exist in all scenes, by default, for use with other Maya features.

In addition to the Materials tab, Hypershade includes other tabs to help you keep track of textures, lights, and other nodes related to rendering.

- 3 With the middle mouse button, drag the lambert1 swatch into the scene view and over the sphere. When you release the mouse, the Lambert material appears on the sphere instead of the Phong E material you previously created.



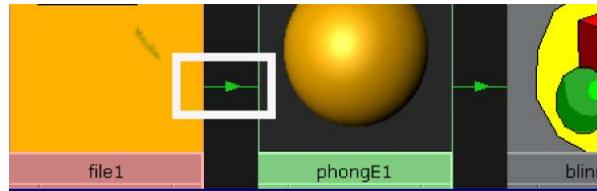
- 4 Drag the phongE1 swatch onto the sphere to reapply that material. Hypershade is designed for quick, drag-and-drop operations such as this.
- 5 Select the sphere.
- 6 From the menus in Hypershade, select Graph > Graph Materials on Selected Objects.
A graph appears in the Work Area tab in the bottom section.



This graph shows all material and texture nodes applied to this object. While the Materials tab shows a catalog of the materials in the scene, the Work Area tab shows whatever material, texture, or other node you are currently working on. The Graph menu has several menu items that change the contents of the Work Area display.

The Work Area shows connection lines between the nodes. The lines have arrows that all point in one direction. For a closer look at these

arrows, dolly into the view using the same keyboard and mouse shortcuts you would use in the scene view.



The arrows depict Maya's internal information input and output, called a stream. The next step shows the importance of this connection stream.

- 7 Click the line between the file1 and phongE1 nodes (it turns yellow) and press the Delete key.

The sphere reverts to the default gray color because you have broken the connection between these nodes and the texture is no longer assigned to it. This is a simple illustration of how the connection stream works. The file1 texture node is upstream from the connection you deleted, so it no longer inputs data into the material; Maya ignores it.

In practical terms, this is an example of how you can disconnect a texture from a material without actually deleting the texture. For example, you may want to remove the file texture from this material and connect it to another material instead. In a later step, you will learn about using Hypershade to make connections between textures and materials.

- 8 Select Edit > Undo to restore the file texture connection you deleted.
- 9 In Hypershade, right-click the center of the phongE1 swatch and select Rename.
- 10 Type: Orange.

Renaming is not as important in this scene as it would be in more complex scenes with several materials.

The Rename operation is one of several useful menu items that appear when you right-click a material swatch. Another handy menu item is Assign Material to Selection. You can use this operation to assign a material to multiple selected objects rather than assign it one object at a time.

Creating a texture within the Hypershade editor

To mimic the skin of an orange, you will add a *bump map* texture. A bump map creates the illusion that the surface has bumps or other types of surface relief. You will map the texture using the Create Bar in Hypershade.



The Create Bar is similar to the Create Render Node window you have used already. It is built into Hypershade for greater convenience.

The Mayakist logo file texture is a 2D texture, which wraps evenly around the surface. Next, you create a 3D texture. A 3D texture is for making objects appear to have been carved out of a solid substance, such as marble or wood.

To create a bump texture within the Hypershade Editor

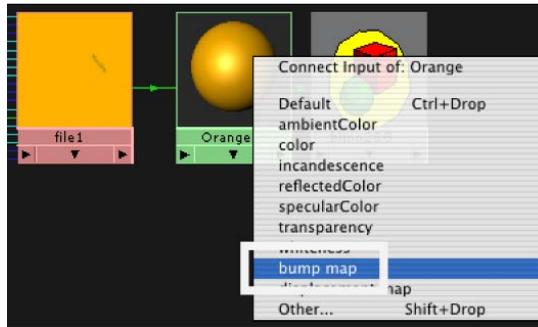
- 1 In the Create bar, select Maya > 3D Textures.

For this bump effect, you use a Brownian 3D texture. Experience has shown that a Brownian texture is the best texture for approximating the bumps on an orange skin. Becoming skillful with textures is a matter of observation, experimentation, and experience.

- 2 With the middle mouse button, drag the Brownian texture (under 3D Textures) onto the Orange material swatch in the Work Area tab and release the mouse button.

- 3 From the menu that appears, choose bump map.

Maya connects or applies Brownian to the Orange material's bump map attribute.

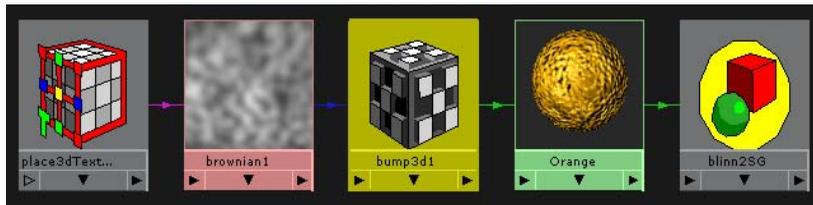


The pop-up menu shows a list of common material attributes. Although you can map textures to any material attribute, these are the most commonly mapped attributes.

By default, you can assign only one texture per attribute. Attributes that already have an assigned texture appear in italics. (For a discussion about assigning multiple textures to an attribute, see [Layered textures](#) on page 483.)

This drag-and-drop method of applying a texture is essentially the same as clicking the map icon next to a material's attribute slider. Using Hypershade offers a faster, more graphical alternative to using the Attribute Editor.

- 4 Select Graph > Input and Output Connections to view the node connections for the new texture you created.



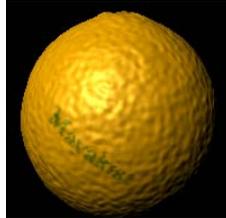
Also notice that bumps from the bump map appear in the Orange material's swatch. The Orange material swatch is the same as the Material Sample that appears in the Attribute Editor for the material node. (The bumps also appear in a rendered image, but not in the scene view.)

- 5 Double-click the bump3d1 swatch (the checkered cube) to display its attributes in the Attribute Editor.

This node is the link between the texture and the material. It converts visual information, such as the mottled Brownian pattern, into the appearance of surface relief.

The most important attribute in the bump3d1 node is Bump Depth, which controls the intensity of the bumps.

- 6 Change Bump Depth to 0.15. In the Orange swatch or a rendered image, you'll see that this diminishes the bumps as compared to the default value 1.0.



NOTE If you select Render > IPR Render Current Frame while Hypershade is selected, a message appears: *Please select the view you want to render.* Either select the perspective view before choosing IPR Render Current Frame, or, in the Render View window, select IPR > Redo Previous IPR Renderer.

- 7 In the Hypershade panel, click the brownian1 texture swatch to reveal its attributes. Set the following:
 - Increment: 0.3
 - Octaves: 4.0

These values produce a rendered image closer to real orange skin.

Our recommendation for these values is based on experimentation. For Brownian and some other textures, it's usually faster to get desirable results by experimenting with the settings rather than trying to understand the definition of each attribute.



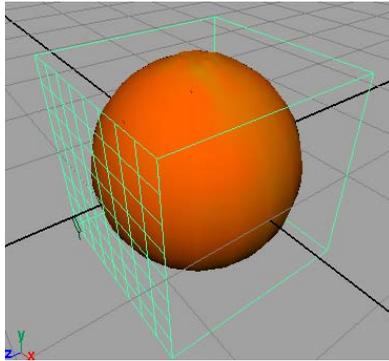
Modifying a bump texture

The bump texture is too crater-like to be a convincing orange. The bumps are too numerous and too deep. A quick way to fix this problem is to scale up the entire texture.

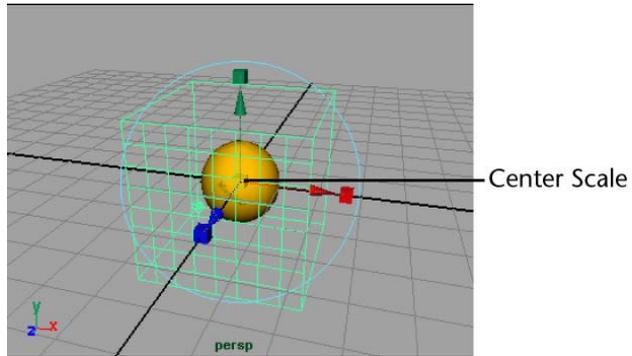
To interactively position a 3D texture

- 1 Double click the `place3dTexture1` swatch, to the left of the Brownian color swatch.

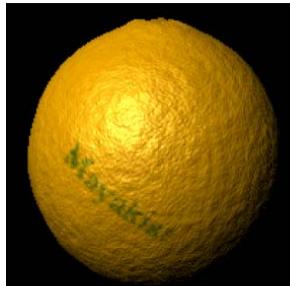
This node controls the 3D texture's position, scale, and rotation. The visual display of this node is a large cube that surrounds the sphere in the scene view. This is the texture placement cube.



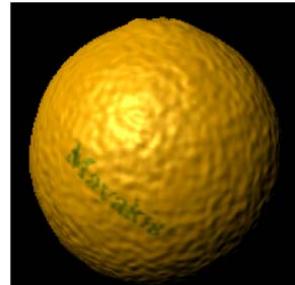
- 2 Click the Interactive Placement button in the Attribute Editor. A manipulator appears for the texture placement cube.
- 3 In the scene view, click on the center scale box to activate the center scale option. Scale the whole texture placement cube to about twice the original size (until the Scale attribute values are about 2).



Now when you render, the bumps on the surface appear wider and smoother, like the bumps on an orange skin. For example, the bumps in the area surrounding the highlight do not appear to be as deep as they were before you scaled the texture.



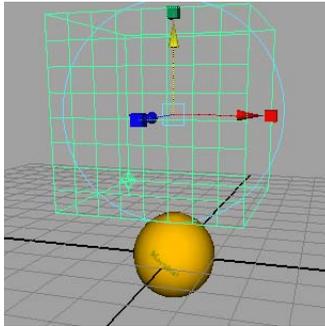
Before scaling



After scaling

Scaling a texture placement cube is a common way to tune the texture's display. Scaling does not actually change the pattern, but enlarges or shrinks it relative to the surface.

- 4 Now that you've changed the surface texture through scaling, move the texture placement cube above the sphere (drag the green arrow up), to see how that affects the texture.



NOTE If you have tumbled the camera for a different view of the orange or the texture placement cube, ensure you reposition the camera so that the orange's Mayakist logo is visible before rendering again. Otherwise, the rendered image will appear incorrect.

In the rendered image, notice that the bumps still remain on the sphere. Objects do not need to be within the texture placement cube, because it represents an imaginary texture volume, which is infinite.

However, movement of the texture placement cube does shift the surface texture pattern. The difference is hard to notice on a uniform texture like Brownian. Transforming the cube would be more noticeable with a texture such as Marble, where the veins within the pattern would be repositioned.

Movement of the object itself also shifts the surface texture. For example, if you were to animate the orange rolling and then render an image sequence, the texture would shift each frame rather than roll with the surface. To prevent this undesirable effect, you need to make sure that the texture moves in the same way the object moves. One way to do this is to parent the texture placement cube (also called the `place3dTexture` node) to the object. For more information on parenting, refer to the Maya Help.

Beyond the lesson

In this lesson, you were introduced to some of the features available for shading surfaces. You learned that:

- Shading materials can employ a variety of material types that can be used to achieve specific surface characteristics (Shininess, Matte, and so on).

- Textures can be applied to a shading material to enhance the realism of the final image. For example, logo graphics, bumps, and so on, can be added to enhance the detail on the surface.
- Textures can be 2D or 3D in nature. Each has its unique techniques for application and placement.

Placement of textures



When you put the Mayakist logo on the surface, the letters appeared straight. The letters are actually skewed in the texture. We skewed the logo for your use in this lesson so that it looks straight when stretched onto the surface.

In your own work, skewing texture images is usually too difficult and impractical. A more practical Maya technique for assigning and positioning file textures is to use texture projection. With this feature, you project the texture onto the surface as if you are projecting a slide image onto a screen. The texture is positioned where the projection strikes the surface. Instead of a skewed Mayakist logo, you could use a straight logo and project it onto the sphere. You could tune the angle of projection so that the curvature of the surface skews the logo just as in the lesson. To assign textures as projections, right-click the 2D texture in the Hypershade or Create Render Node window and select the Create as projection option.

The placement of texture projections is similar to the placement you performed for the bump map (3D) texture. Like the bump map texture, you need to parent the texture projection (the place3dTexture node) to the object in order to avoid the texture placement changing when the object moves.

Layered textures

Suppose you want to add a scarred pattern to the orange skin. Because you can only have one texture assigned to the Color attribute you can use a Layered Texture to combine two or more texture maps.

The Layered Texture is a texture node that lets you composite or layer several textures on top of each other. You assign it to a material's attribute just like other textures. Within the Layered Texture's attribute controls, you can connect

other textures and control how they blend together. For example, the bottom layer could be the logo image you used in the lesson, and a top layer could be an image of streaking lines that resemble scars. The combined logo and scars would give the orange the final aged look that you wanted.

Shader Library

As an option, you can install a Shader Library with Maya to help you quickly create common materials, such as bricks, fabrics, foods, and many other surface types. When installed, a Shader Library tab appears in Hypershade. To use a material, find the type of shading you want, drag and drop it into Hypershade, then drag and drop it onto a surface in your scene using the middle mouse button.

Lesson 3: Lights, shadows, and cameras

Introduction



Image courtesy The Art of Maya

Directors of live-action film use camera and lighting techniques to enhance the visual impact of a scene. Wide, medium, and close up views of the subject, depth of field, high or low key lighting all communicate something about the subject and the scene to the viewer.

Lighting and camera techniques are one of the most crucial aspects to consider when working with artificial characters and objects. The more realistic the lighting and shading appear, the more convincing the scene will appear to the viewer.

A prerequisite to creating effective 3D rendered animation is to study the lighting and camera effects used in live-action film. Your goal is to create the desired scene ambience while keeping the lights and camera view as unobtrusive as possible.

In Maya, it is possible to simulate both realistic and unrealistic lighting and shadow effects for your final images and animations. In this lesson, you learn basic techniques for working with lights, shadows, and cameras. In this lesson, you learn how to:

- Create and position various types of lights in the scene.
- Edit the attributes of lights and experiment with their effect in the scene.
- Cast shadows on your surfaces using the lights and objects in the scene.
- Create and animate an additional camera in the scene.

Open the scene for the lesson

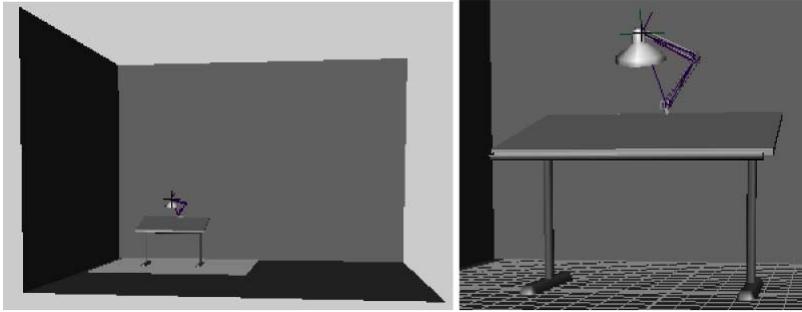
In this lesson, you work with a scene we created for your use. In the first steps, you open the scene, create a directional light, and compare the directional light's effect to the scene's default lighting.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 439.
- 2 Open the scene file named `Lights.mb`.

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

```
GettingStarted\Rendering\Lights.mb
```

The scene contains a room with two walls and a floor. Sitting in the room is a drafting table and lamp, grouped under the name `tableGroup`. All objects have a default gray shading.



When you open the scene, Maya lights the scene with default lighting provided by an imaginary light fixture. It emanates light infinitely from a point above and behind the active scene view camera. There is no icon that represents the default light in the scene view.

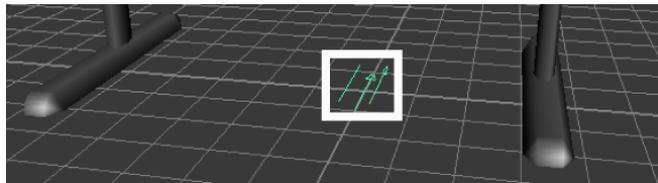
Directional lights

Maya has many types of lights that simulate natural and artificial lighting. In the next steps, you create a light for the scene using a *directional light*. A directional light uses parallel rays of light, as if illuminating from a very far distance, to illuminate the scene. A directional light is often used to simulate sunlight.

To create a directional light

- 1 Select Create > Lights > Directional light > .
- 2 Turn off Interactive Placement, then click Create.

A directional light is created at the center of the scene.



Because a directional light is similar to sunlight. Its parallel rays strike all objects in the scene from a single direction as indicated by the arrow icon representing the light. The position of the light is not so much important as the direction that the arrow icon points.

When you create a light, the scene view does not display its effect, by default. The scene view instead uses default lighting.

- 3 Select **Lighting > Use All Lights** (Hotkey: 7). This lights up the scene view only with lights you've created, not with default lighting. If you later want to see the scene view with default lighting again, select **Lighting > Use Default Lighting** (Hotkey: 6).

When you render the scene, by default, Maya uses all lights you've created. If you don't create any lights, Maya creates a temporary default directional light for you and then deletes it when the render is complete.

Next, you aim the directional light and edit its attributes.

To edit the directional light

- 1 With the directional light selected, rotate the light in various directions. The shading of surfaces changes as you rotate the light. The more directly the light points at a surface, the brighter the shading. A directional light is affected by its rotation, not its position. As you'll see later, the position of other lights affects the lighting.
- 2 Rotate the light as follows:
 - Rotate X: -40
 - Rotate Y: 25
 - Rotate Z: -20

With this orientation, all object surfaces in the scene show the effect of the light in the current camera view.

- 3 With the light still selected, open the Attribute Editor (under the Window menu). Drag the Intensity slider to various values to see the effects of intensity.

Higher values brighten the surfaces. For example, an Intensity of 1.6 brightens the lighting so much that the gray default shading of some surfaces are bleached to white.

NOTE Several of the following illustrations in this lesson are snapshots of the scene after rendering. To render the scene, select **Render > Render Current Frame**. Do not use IPR rendering for this lesson because it doesn't automatically update the image for some of the changes you make to the scene.



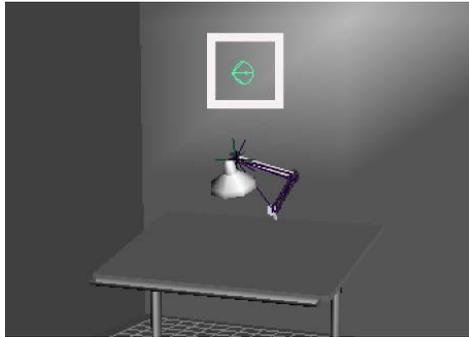
- 4 In the Attribute Editor, click the white Color box above the Intensity box. This displays the Color Chooser.
- 5 Click inside the color wheel (hexagon) and drag the pointer to a red color. The lighting imparts a red hue to the surfaces in the scene. Move your mouse off the Color Chooser to exit.
- 6 Change the color back to white and set the Intensity to 1.2 or so. You'll use these settings for the basic lighting of the scene.

Spotlights

In the next steps, you light part of the desk by creating a spotlight and positioning it within the desk lamp. A *spotlight* emits light from a single point within a limited cone angle. You can aim a spot light in the direction you want the light to illuminate.

To create a spotlight

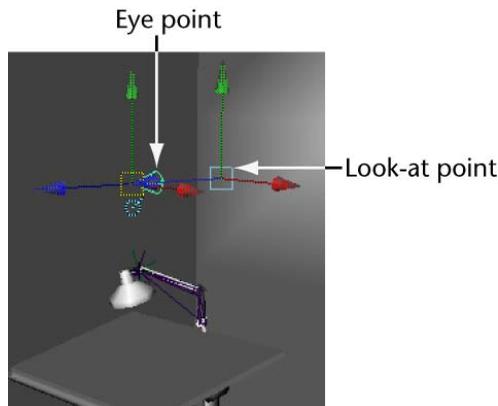
- 1 Select Create > Lights > Spot Light >  .
- 2 Turn off Interactive Placement, then click Create. This creates the spotlight's icon at the center of the grid. The light points toward the back wall.
- 3 Move the spotlight above the lamp. (Drag the light up along its local Y-axis.)



If you look at the spotlight icon from several angles, you'll notice that it is shaped like a cone with an arrow pointing out of it.

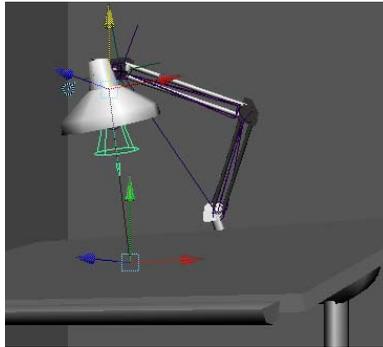
The cone symbolizes that a spotlight emits a beam of light that gradually widens with distance.

- 4 Select Modify > Transformation Tools > Show Manipulator Tool. See Modify > Transformation Tools > Move Tool, Rotate Tool, Scale Tool, Show Manipulator Tool for more information. With the spotlight selected, this tool provides two manipulators that you can move to position and aim the light precisely.



The look-at point specifies where the light focuses. The eye point defines the position of the light source. All types of lights have an eye point, but not necessarily a look-at point.

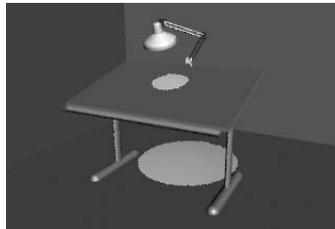
- 5 Move the look-at point to the top of the table, and move the eye point roughly to the center of the lamp housing.



You can also use the Rotate tool to center the eye-point more accurately in the lamp housing.

An alternative way to position a spotlight is to select the light and then select Panels > Look Through Selected. You can then dolly and pan the view to focus on the desired surface. The area of focus is where the light strikes the surface. To return to the perspective view select Panels > Perspective > persp.

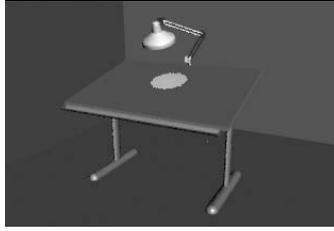
- 6 If you render the scene, you'll notice that the spotlight lights up a circular region on both the desk and the floor.



By default, light passes through and lights up each surface in its path. The circular region of light is wider on the floor than the desk because a spotlight emits a beam of light that widens with distance.

Only the floor, not the desk, will be lit by the spotlight in the scene view. You'll see the desk lit up only if you render the scene.

- 7 You can prevent the light from passing through the table. With the light selected, display the Attribute Editor. Open its Shadows section and then the Depth Map Shadow Attributes section, then turn on Use Depth Map Shadows. Rendering creates this result:



By turning on Use Depth Map Shadows, the spotlight's lighting is blocked by the first surface it hits (the desk). The floor is in the shadow of the desk so it receives no light. You'll learn more about Depth Map Shadows later in the lesson.

Editing light attributes

Spot lights, directional lights, and other light types have several unique attributes (for example, Intensity, Dropoff, Color, etc.). In the next steps, you modify some of the attributes for the spotlight.

To modify the attributes of a spotlight

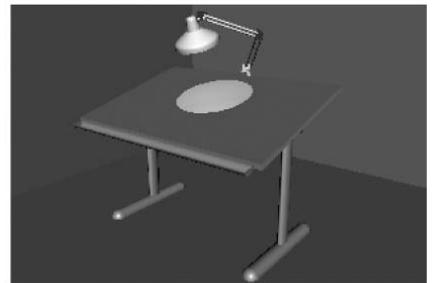
- 1 The spotlight's default Cone Angle is 40 degrees. In the Attribute Editor, open the Spotlight Attributes section and set Cone Angle to 20. The circle of light narrows.

- 2 Set Cone Angle to 60.

The circle of light widens. Rendering creates this result:

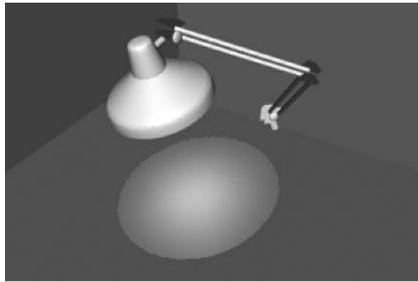


Cone Angle = 20

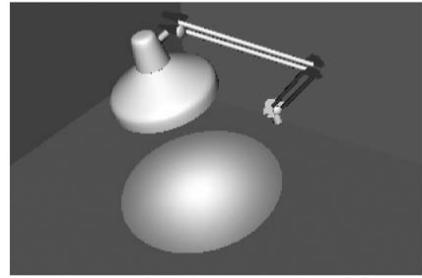


Cone Angle = 60

- 3 Set Dropoff to 10.



Dropoff = 10 Intensity = 1



Dropoff = 10 Intensity = 1.6

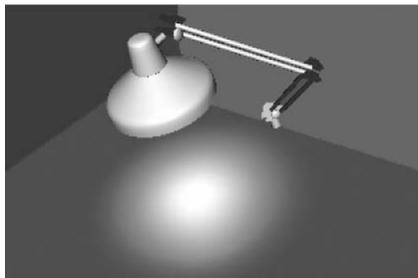
This attribute sets how much the light intensity diminishes from the center of the circular region out to the edge. Finding the right value is a matter of experimentation. When you increase the Dropoff, the light's intensity diminishes, so you need to increase the Intensity. For example, increase the Intensity to 1.6. Rendering creates this result:

To mimic the way a light source's intensity naturally decreases with distance, you might want to select a Decay Rate to diminish the light's intensity. The Linear setting diminishes light with the least abruptness, Quadratic with the most abruptness. Cubic is between the two other settings.

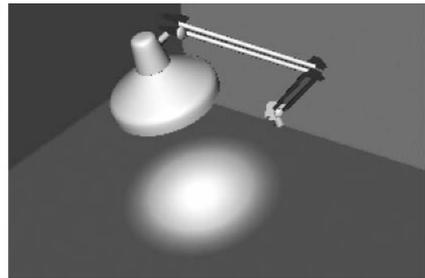
The default setting is No Decay, which means the light remains at full intensity regardless of distance. If you use a setting other than No Decay, you likely will need to increase the Intensity. For this lesson, leave the Decay Rate set to No Decay.

4 Set the Penumbra Angle to 10.

This attribute controls the softness along the edge of the area of illumination provided by the light. Again, finding the right value is a matter of experimentation. You can also use negative numbers to fade inward from the circle's edge. The viewing angle of the camera influences the appearance of the fading at the edge. Rendering creates this result:



Penumbra Angle = 10



Penumbra Angle = -10

- 5 In the Light Effects section of the Attribute Editor, click the map button to the right of the Light Fog box.



This gives the appearance of the light beams illuminating fog or dust in the air. Rendering creates this result:

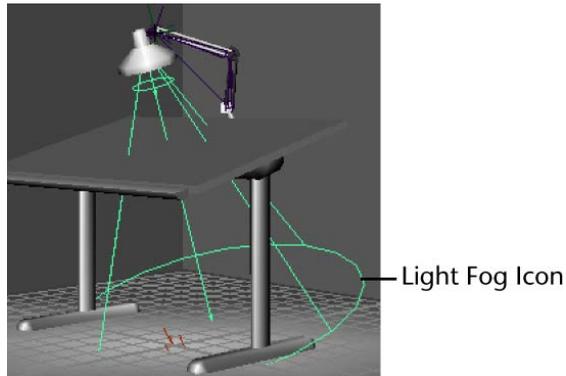


Fog is more noticeable when you view (render) the light beam from a side angle rather than from above the light fixture.

When you click the Light Fog map button, Maya creates a light fog node and displays its attributes in the Attribute Editor.

The Color and Density attributes are useful for altering the fog's hue and transparency. These attributes affect only the light on its way to the surface, not the light on the surface.

In the scene view, you'll also notice a Light Fog icon extending from the spotlight icon. When highlighted, it looks like this:



The icon displays the region of the potential illumination. You won't see light fog under the table. The table blocks the light fog because you previously turned on spotlight's Depth Map Shadows attribute. You can

use the Scale tool on the icon to expand or contract the region of the illumination effect.

Shadows

Shadows are the darkened areas that appear on a surface when an object gets in the path of a light source. Shadows are cast onto the area of a surface that doesn't directly receive light. Shadows ground the objects in your scene and aid in defining their three-dimensional appearance. Shadows can add drama to your scene.

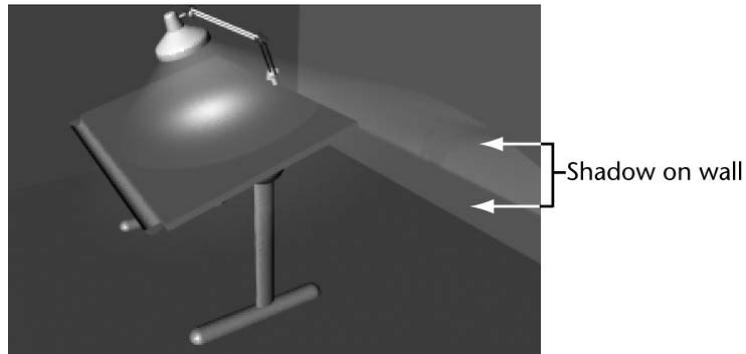
Next, you create a shadow on the floor from the light cast by the spotlight. To do this, you must edit a few of the spotlight's attributes.

To create shadows using a spotlight

- 1 Select the spotlight and click the `spotLightShape1` tab in the Attribute Editor.
- 2 Set the Cone Angle to 105.

When you render the scene, you'll see the spotlight has widened its beam to light up a small part of the back wall. The table and light fixture block light and create shadows.

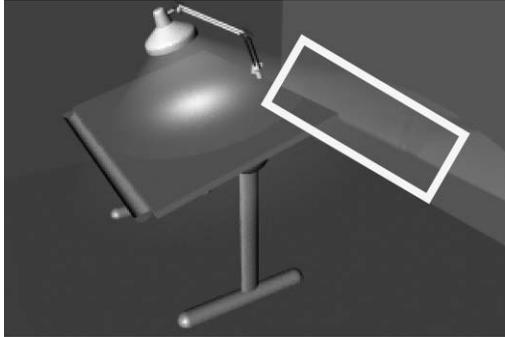
The shadows occur because you turned on Use Depth Map Shadows for the spotlight earlier in the lesson.



Depth Map Shadows refers to the algorithm Maya uses to produce shadows. For basic shadowing, you don't need to know about the algorithm, but you do need to make sure you turn on Use Depth Map Shadows.

- 3 To sharpen the shadow's edges, open the Attribute Editor's Shadows section and then the Depth Map Shadow Attributes section. Set the Resolution to 1024.

Rendering creates this result:



Depth Map Resolution =1024

Increasing the shadow's resolution increases rendering time. Use a higher value only if it makes a noticeable difference on the shadows. In this case, the difference is so subtle that you might prefer to use the default value 512. In your future projects, keep in mind that higher shadow resolution values can make a big difference in the quality, but also affect rendering time.

A related attribute, Filter Size, is useful for blurring or softening shadow edges that seem too sharp. Increase the value to increase the blur. Again, higher values increase rendering time.

If you are intent on perfecting shadows in your future projects, try various combinations of values for Resolution and Filter Size. For this lesson, the default values are satisfactory because the shadow edge has minimal contrast. If you create shadows with sharp contrast at the edges, altering the default values will be more desirable.

Creating additional cameras in a scene

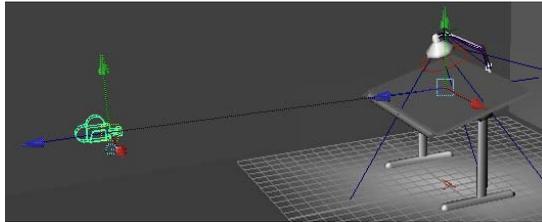
Elsewhere in this book, you've used the default perspective camera to examine and render your modeling and animation work in progress. In some situations, it is convenient to create and use additional cameras.

For example, you might want to render the scene from two different camera views to compare the results. Alternatively, you might want to use the default camera for interactive modeling and animation and the second camera strictly

for rendering the scene. In the next steps, you create an additional camera and manipulate its view of the scene.

To create a second camera for the scene

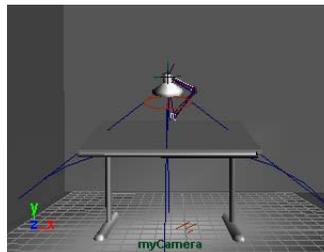
- 1 Select **Create > Cameras > Camera**. This creates a perspective camera with an icon representing it at the center of the grid.
- 2 In the Channel Box, change the name of the camera to **myCamera**.
Even though you've created a new camera, you are still looking at the scene through the default perspective camera named *persp*. You will view the scene through **myCamera** after you position it in the next step. If you were to view the scene through **myCamera** now, you would see a disorienting view of the back wall and floor. The camera's icon points in the direction of the camera's view.
The scene's default *persp* camera has no icon. An icon is displayed only for a camera you create.
- 3 Use the **Show Manipulator Tool (Modify > Transformation Tools > Show Manipulator Tool)** to position **myCamera** roughly in this position:



With a camera selected, the **Show Manipulator Tool** provides the same two manipulators that you used to position the spotlight: **eye point** and **look-at point**. The **look-at point** sets where the camera aims. The **eye point** sets the position of the camera.

You can select the current camera by selecting **View > Select Camera**.

- 4 Select **Panels > Perspective > myCamera** to view the scene from **myCamera**.



If your camera view is not similar to the above illustration, select `tableGroup` in the Outliner (`Window > Outliner`) and select `View > Frame Selection` from the panel menu. This centers the view of `myCamera` on the table.

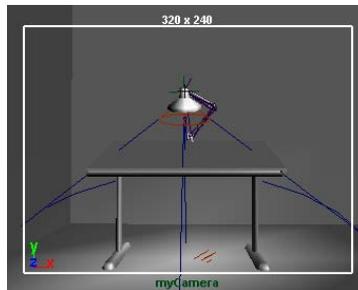
The dolly, track, and tumble movements work the same for `myCamera` as they do for the default `persp` camera.

You can return to the view of the `persp` camera by selecting `Panels > Perspective > persp`.

- 5 While working with a camera, it's useful to know the exact region of the `myCamera` view that is displayed in rendered images. To see the border of the render region, select `View > Camera Attribute Editor`. In the Attribute Editor, open the `Display Options` section and turn on `Display Resolution`. A rectangular border shows the region that will be rendered.
- 6 In the `Film Back` section of the Attribute Editor, set the `Fit Resolution Gate` setting to `Overscan`.

By enabling the `Overscan`, the view displays more of the region outside the area that will be rendered.

By displaying a small part of the scene that lies outside the rendered area, you can plan future camera movement more easily, especially if the scene has objects that move in and out of view.

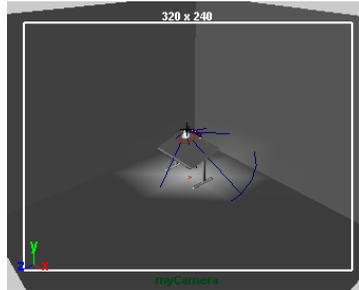


Animating camera moves

In Maya, the camera can be animated. You can set keyframes for the camera moves the same as for other objects in your scene. In the next steps, you set keys to do a simple animation of the camera's movement.

To animate the camera

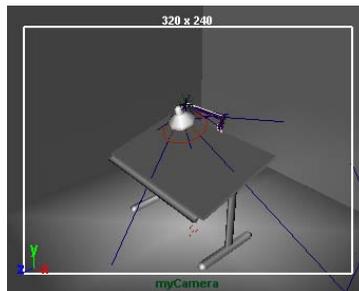
- 1 Go to the start of the playback range.
- 2 Dolly, tumble, and track myCamera to create a view similar to this:



- 3 With myCamera selected, set a key for the camera's current view at the first frame. To set a key, select the Animation menu set, and then select Animate > Set Key. (For details on setting keys and animation technique, see [Lesson 1: Keyframes and the Graph Editor](#) on page 204.)

TIP You can undo and redo view changes such as dolly, track, and tumble by selecting either the [or] bracket respectively. You can press these keys repeatedly to undo or return to a particular view orientation.

- 4 Go to frame 150.
- 5 Dolly myCamera to create a view similar to this:



- 6 Set another key.
- 7 Play the animation to see the camera dolly toward the table for the first 150 frames.

At this point, feel free to experiment with keying changes in the view resulting from tumbling and tracking the camera. Excessive animation of tumbling and tracking creates a dizzying effect that is a common mistake of inexperienced artists. In general, it's best to keep camera animation simple except when you are striving for an unusual effect.

Beyond the lesson

In this lesson, you were introduced to some basic techniques related to lights and cameras in Maya. You learned how to:

- Place two types of lights in your scene and edit their attributes.

In addition to the directional light and spotlight you created in this lesson, Maya has several other types of lights for creating nearly any type of natural or artificial lighting. You might use as few as two lights or as many as 10 to get the look you want. Generally, the more lights you create, the longer the rendering will take.

In addition to the light fog effect you created in this lesson, you can also create Glow, Halo, and Lens Flare light effects with a similar workflow. To see these effects, the light must point toward the camera view.

In some situations, you might want to prevent a light from striking a surface in its path. For example, your scene shows a person leaning against a wall in an outdoor night setting. You might want to light the wall with dark, shadowy spotlights, and the person's face only with a brighter, soft spotlight. Because the face and the wall are near each other, all the spotlights currently strike both objects. You can use Maya's light-linking capability to isolate the lighting.

- Create shadows in the scene by having the lights cast shadows.

The mental ray for Maya renderer has three types of shadows: Depth Map Shadows, Ray Traced Shadows, and shadow maps. In this lesson, you created Depth Map Shadows, which are less realistic than Ray Trace shadows but much faster to render.

Ray Traced Shadows are mainly useful in scenes where the audience is focusing on the shadow region, for instance, a close-up of a still setting. Ray Traced Shadows are also useful if you want to cast shadows more accurately for transparent objects. To reduce rendering time, use Ray Traced Shadows only when necessary.

You can create shadow maps if you are using a spot light, directional light, point light or area light. There are two types of shadow maps: Regular Shadow Map, or Detail Shadow Map. The regular shadow map takes fewer

samples than the detail shadow map but requires more resolution. The detail shadow map, on the other hand, takes more samples per pixel and therefore requires less resolution. Detail shadow maps also take extra attributes into account such as refraction, reflection and transparency settings on objects. You can therefore use detail shadow maps to render colored shadows in your scene.

- Create and animate an additional camera from which to view the scene. If you have previous experience with photography, you'll recognize the names of various camera attributes displayed in the Attribute Editor while a camera is selected. You can modify the attribute values to suit your requirements.

The values for certain camera attributes (for example, Film Aspect Ratio) might cause a rendered image to not match the region bounded by the Resolution Gate. In such cases, you can display the Film Gate (View > Camera Settings > Film Gate) to see the region that will be rendered.

You can animate a camera along a path; for example, to create the illusion of flying through mountainous terrain. This type of animation involves the use of Animate > Motion Paths > Attach to Motion Path (from the Animation menu set). If you additionally want the motion path-animated camera to focus on a stationary or moving object, you need to use Create > Cameras > Camera and Aim. If you want the camera to move in a looping path, (for example, a roller coaster) you need to use Create > Cameras > Camera, Aim, and Up.

Before you batch render a scene, you must select the camera you want to use for rendering. Select Window > Rendering Editors > Render Settings to display the Render Settings window. In the File Output section, select the camera from the Camera option.

Lesson 4: Global Illumination

Introduction

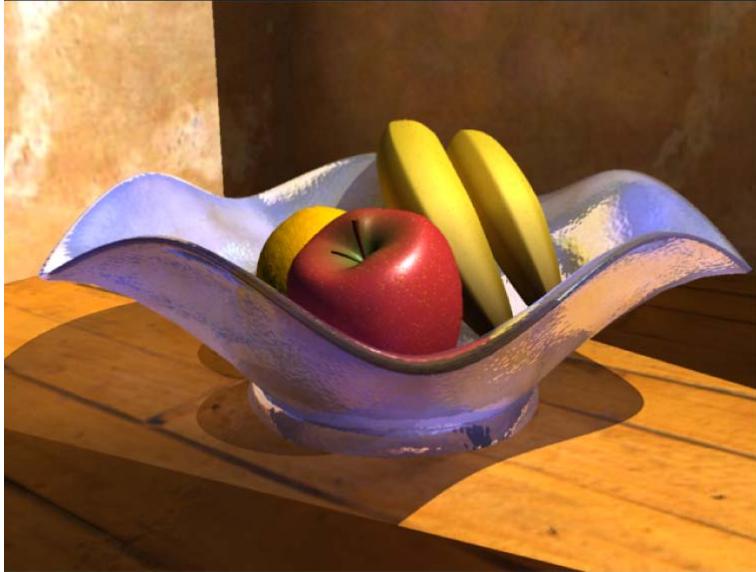


Image created by: Vivien May

The mental ray® for Maya® renderer can render using a feature called *Global Illumination*. Global Illumination simulates the effect of all lighting and inter-reflection in the scene, whether the items are lit directly by a light source or indirectly illuminated by other objects (and diffuse sources of illumination in the scene).

For example, when a photographer lights an object in a studio, some of the lighting for the object in the final image comes directly from the lights that are used, while additional illumination comes from objects that are placed near the subject to act as light reflectors. These are referred to as fill cards. Light bounces off the fill cards, indirectly lighting nearby objects with a secondary source of diffuse light. The color of the fill card colors the light that gets reflected onto the object. The final illumination of the object comes from this combination of multiple sources (direct light, specular reflections, and diffuse light).

The Global Illumination feature in the mental ray® for Maya® renderer simulates this effect and provides for a realistic illumination simulation in your Maya scenes.

In this lesson you learn how to:

- Differentiate between images made with Global Illumination compared to other lighting types.
- Render an image using Global Illumination by performing repeated test renders to achieve a final result which includes:
 - Lighting a scene using the mental ray Global Illumination attributes.
 - Adjusting the Global Illumination attributes and quality settings in a systematic manner to achieve a high quality image.
- Visualize photon maps for the scene.

Open the scene for the lesson

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

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 439.
- 2 Open the scene file named `GlobalIllum.ma`.

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

```
GettingStarted\Rendering\GlobalIllum.ma
```



Shaded view

The scene contains a still life with a fruit bowl. Other surfaces act as the floor and walls for the scene. The primary light source is a spotlight. The spotlight casts light through a vertical opening in the wall. Some areas in the scene are directly lit and other areas are not.

Shading materials have already been assigned to the bowl, fruit, and other surfaces. The apple and orange models are from the previous lessons. If you want to see what other shaders and textures are in the scene, you can view them by opening the Hypershade editor.

When you render this scene using Global Illumination, the effect you want to achieve is to have the upper and lower regions of the scene receive indirect light: you want to be able to view more detail in the shadow areas as a result of the Global Illumination.

Render the scene using raytracing

To better understand the differences in a rendering performed using direct lighting methods vs. global illumination, you begin by rendering the scene using the mental ray for Maya renderer without using Global Illumination.

You start by performing some preparatory setup and then render the scene. In this section you:

- Select the mental ray for Maya renderer.
- Set the quality setting for your rendering.
- Set the image size for your rendering using the Image Size settings.
When performing repeated test renders, a good practice is to render small images so the tests render quickly.
- Turn on shadows for the spotlight in the scene.
- Render an image using the mental ray for Maya renderer using raytracing.

Using the mental ray for Maya renderer

To use the Global Illumination feature in Maya you must render using the mental ray for Maya renderer.

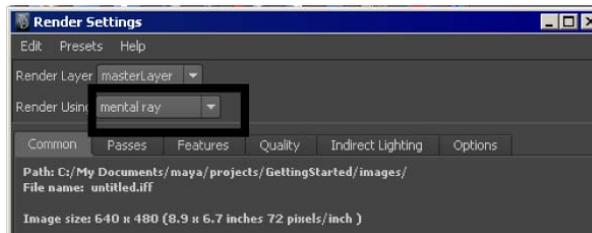
You initially render the scene without using Global Illumination.

To change the renderer to mental ray for Maya

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon on the Status Line) to display the Render Settings window.



- 2 In the Render Settings window, set the Render Using setting to mental ray.



When you render an image it will use the mental ray for Maya renderer.

NOTE When you switch between renderers in Maya (for example, Maya Software, Maya Hardware, and mental ray for Maya), the common Render Settings do not require re-setting.

When you perform repeated test renderings for a particular scene, consider the time it takes for each render to complete. If you reduce the time it takes to render each test image, the testing process proceeds much faster.

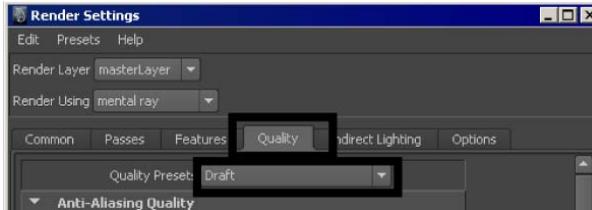
Setting render settings for raytracing

For the first rendering test, you render the scene to see how the scene appears without Global Illumination. The testing process proceeds faster if the quality level for the image is initially set low.

To set presets for rendering

- At the top of the Render Settings window, click on the Quality tab and set the Quality Presets setting to Draft.
The Draft setting ensures low anti-aliasing settings are used and reduces the rendering time compared to other preset settings. This is a good

practice when evaluating the lighting and basic material properties in your test renderings.

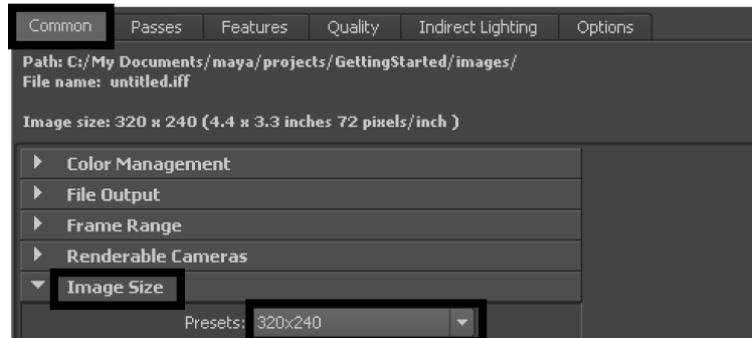


Setting the image size for rendering

You set the size that an image is rendered in the Render Settings window. For this lesson, you render the image at a size that allows you to evaluate the Global Illumination effect.

To set the image size for rendering

- 1 At the top of the Render Settings window, select the Common tab and open the Image Size settings.



- 2 Set the Preset to 320 X 240.
This sets a small image size for your rendering tests. The smaller image size renders in less time when compared to a larger image size.
- 3 Close the Render Settings window.

Turning on shadows

Shadows add realism to a scene, and in this lesson, help to demonstrate how a scene rendered using Global Illumination differs from a scene rendered using direct illumination.

To turn on shadows for the spotlight

- 1 Open the Hypershade window (Window > Rendering Editors > Hypershade) and click on the Lights tab to display the lights in the scene.



The scene is lit with one spotlight named spotLightShape1.

- 2 In the Hypershade window, double-click the icon for spotLightShape1 to display the attributes for the spotlight.
- 3 In the Attribute Editor, click the spotLightShape1 tab to display the lighting attributes for the spotlight. (You may need to minimize the Hypershade window to see all of the Attribute Editor.)
- 4 In the Shadows section, open the Raytrace Shadow Attributes turn on Use Ray Trace Shadows. This turns on shadows for the spotlight.



- 5 Close the Attribute Editor and Hypershade window.

Rendering the image

When you render an image, all of the objects, lighting, shading materials, and image quality settings are used to calculate the image, based on the particular camera view that is set.

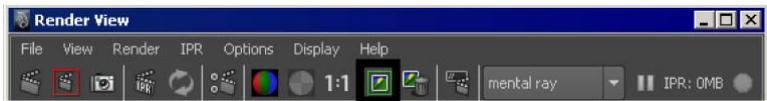
To render an image using mental ray for Maya

- 1 Click in the perspective window. This indicates which camera view you want Maya to render.
- 2 On the Status Line, click the Render Current Frame icon to launch the renderer to produce an image.



The Render View window appears and the fruit bowl scene renders using the mental ray for Maya renderer.

- 3 Once the render is complete, click the Keep Image icon in the Render View window to save this image.



Keeping the image allows you to compare it to any subsequent test renders you produce. In this lesson you compare each rendered image to the previous ones to understand how the changes you make in various settings affect the rendered image.



mental ray raytrace rendering

The rendered image shows the fruit bowl lit from the right of the frame. The upper and lower portions of the image are in shadow where the edges of the wall prevented the spotlight from lighting those portions of the scene. The shadow areas appear dark and do not show any detail of the floor or walls. This illumination effect is typical of an image produced with a single light source and direct illumination.

Render the scene using Global Illumination

To render using Global Illumination you need to turn on specific settings or *attributes*. There are two areas that you must initially set.

- Turn on the Global Illumination rendering attributes using the Render Settings Window.
- Turn on the Global Illumination lighting attributes using the Attribute Editor.

You initially render the scene using the attributes at their default settings to see how they affect the rendered image and then compare it to the previous raytrace rendering. As you gain more experience rendering with Global Illumination you will better understand what settings to initially use.

To turn on Global Illumination rendering attributes

- 1 Open the Render Settings window.
- 2 In the Render Settings window, click on the Quality tab and set the Quality Presets setting to Preview: Global Illumination.



This turns on preset attribute settings that enable and control the quality of the Global Illumination. You initially render the scene using the default settings. Once you've rendered an image and observed their effect, you return to the Render Settings window and modify the settings accordingly.

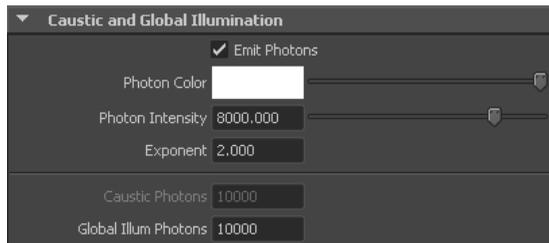
- 3 Close the Render Settings window.

Global Illumination uses *photons* to simulate the light inter-reflection effects in a scene. Photons are small packets of energy emitted from a light source into the scene that bounce off surfaces to determine the intensity of light for those areas not receiving direct illumination. The renderer keeps track of the illumination that results from this photon emission in a file called a *photon map*.

In addition to having the spotlight emit light you need to set the spotlight to also emit photons for the Global Illumination simulation to occur. The overall illumination for the scene will be determined from a combination of the light's intensity and photon illumination from this single spotlight.

To turn on Global Illumination lighting attributes

- 1 Open the Hypershade window (Window > Rendering Editors > Hypershade) and click on the Lights tab to display the lights in the scene.
- 2 In the Hypershade, double-click on the icon for spotLightShape1 to display the attributes for the spotlight.
- 3 In the Attribute Editor, click the spotLightShape1 tab to display the lighting attributes for the spotlight.
- 4 Open the mental ray section, and in the Caustic and Global Illumination attributes set the following settings:



- Emit Photons: On
- Photon Intensity: 8000
- Global Illum Photons: 10000

The spotlight emits photons into the scene when you next render an image. Photon Intensity sets the brightness of the Global Illumination. Global Illum Photons sets the number of photons that get emitted into the scene.

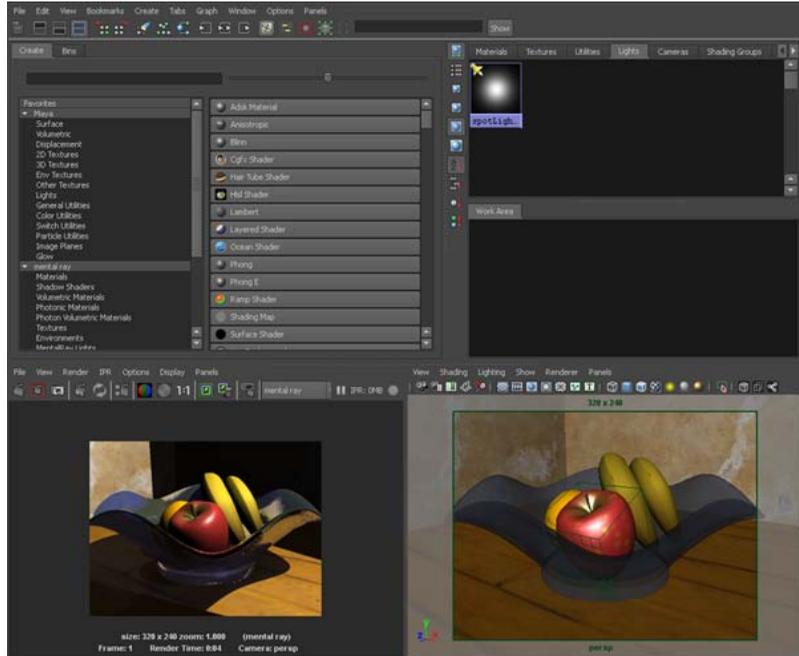
Setting a panel layout for rendering

The process of producing a rendered image usually involves rendering and evaluating successive test images to achieve the desired final image. This process involves adjusting many render quality, shading, and lighting settings in specific windows and panels. Opening and closing the various windows can become tedious.

Displaying the most frequently used windows and panels for rendering allows you to work more efficiently without having to open and close them repeatedly.

To set a panel layout for rendering

- 1 Close the Render View window.
- 2 From the perspective view's panel menu, select Panels > Saved Layouts > Hypershade/Render/Persp.



The panel layouts update to display the Hypershade window, Render View window, and Perspective view simultaneously. This allows you to easily work with these windows without having to open and close them repeatedly.

If the image in the Render View window appears pixelated as a result of the new panel layout, do the following:

- In the Render View window menu, select View > Real Size. This resizes the rendered image to its actual size.

TIP You can resize the panels in a preset layout by dragging the border between neighboring panels using your left mouse button.

Rendering the scene with Global Illumination

Now that the lights are set to emit photons and the Global Illumination preset is selected, you're ready to render an a test image using Global Illumination.

To render the scene using Global Illumination

- 1 In the Render View window, click the Redo Previous Render icon to render the image with the default Global Illumination settings.



- 2 When the image has completed rendering, click the Keep Image icon to save the image.
- 3 Visually compare this rendered image with the previously rendered images.



Global Illumination = Off



Global Illumination = On

In this rendering, the shadow areas of the scene appear less dark as a result of Global Illumination. The bowl also appears more transparent than in the previous rendering.

The effect you want to achieve is to have the shadow regions receive indirect light; that is, you want to be able to view more detail in the shadow areas as a result of the Global Illumination. To do this, you must adjust the Global Illumination photon attributes. After you change an attribute, test render the scene to see the effect of this change in the rendered image. These repeated test renderings or *iterations* allow you to work systematically towards the goal of creating the desired final image.

In the next steps you use the Attribute Editor to increase the Photon Intensity to increase the overall Global Illumination for the scene.

Increasing the brightness of Global Illumination

Increasing the Photon Intensity setting increases the brightness of Global Illumination. The goal is to provide more visible detail and brightness in the shadow areas of the scene in the next test image.

To increase the Global Illumination lighting

- 1 In the Hypershade window, double-click on the icon for `spotLightShape1` to display the attributes for the spotlight.
- 2 In the Attribute Editor, click the `spotLightShape1` tab to display the lighting attributes for the spotlight.
- 3 In the Caustic and Global Illumination attributes adjust the settings to the following:
 - Photon Intensity: 120000

Increasing the Photon Intensity increases the illumination in the areas affected by Global Illumination.

To render the scene

- 1 In the Render View window, click the Redo Previous Render icon to render the image.
- 2 When the image has completed rendering, click the Keep Image icon to save the image.
- 3 Compare this rendered image with the previously rendered images.



Photon Intensity = 8000



Photon Intensity = 120000

The brightness of the illumination in this image has increased as a result of increasing the Photon Intensity. This is indicated by the increased brightness and details seen in the shadow areas of the scene. However,

some circular spots have also appeared in the image along the dark wall behind the bowl. These spots are the result of low photon quality settings.

Adjusting the quality of Global Illumination

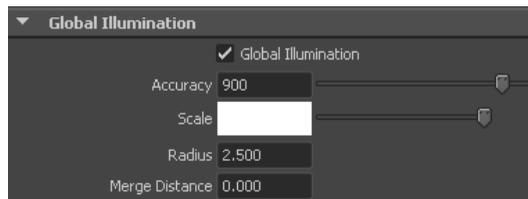
The quality of Global Illumination is primarily affected by three attributes:

- The accuracy of sampling (Accuracy)
- The radius size that determines the size of the region surrounding a particular sampling point (Radius).
- The number of photons (Global Illum Photons).

In the next steps, you return to the Render Settings window and the Attribute Editor to increase the quality attributes that affect the Global Illumination effect.

To increase the Global Illumination quality settings

- 1 In the Render Settings window, click on the Indirect Lighting tab and expand the Global Illumination section. Change the following Global Illumination attributes to:
 - Accuracy: 900
 - Radius: 2.5



The *Accuracy* setting determines how many photons will be evaluated to determine the Global Illumination intensity at a given sampling point. *Radius* determines the size of the region surrounding the particular sampling point. The more photons that get sampled, the more even the Global Illumination will appear in the final image.

NOTE As you increase the Global Illumination related quality and photon settings, the time to render the image also increases. Your goal should be to increase these settings just enough to achieve the desired image quality, without increasing them beyond what is required so that your rendering times are not adversely affected.

- 2 In the Render View window, click the Redo Previous Render icon to render the image.
- 3 When the image has completed rendering, click the Keep Image icon to save the image and then compare this rendered image with the previously rendered images.



Accuracy = 64
Radius = 0



Accuracy = 900
Radius = 2.5

When the Accuracy and Radius values are increased, the resulting rendered image shows the circular spots have almost fully disappeared and the Global Illumination is much more uniform when compared to the previous rendering. Increasing the number of Global Illumination photons should aid in making the Global Illumination more uniform.

In the next steps, you increase the number of photons to provide a more uniform Global Illumination in the scene.

To increase the Global Illumination photons

- 1 In the Hypershade, double-click on the icon for `spotLightShape1` to display the attributes for the spotlight.
- 2 In the Attribute Editor, click the `spotLightShape1` tab to display the lighting attributes for the spotlight.

- 3 In the Caustic and Global Illumination attributes adjust the settings to the following:
 - Global Illum Photons: 80000

When you increase the number of Global Illumination Photons you are increasing the resulting density of the photon map that gets produced at the beginning of the render process. The more photons, the smoother the resulting Global Illumination. Increasing the number of photons also increases the rendering time.

To render the image

- 1 In the Render View window, click the Redo Previous Render icon to render the image.
- 2 When the image has completed rendering, click the Keep Image icon to save the image and compare this rendered image with the previously rendered images.



Global Illum Photons = 10000



Global Illum Photons = 80000

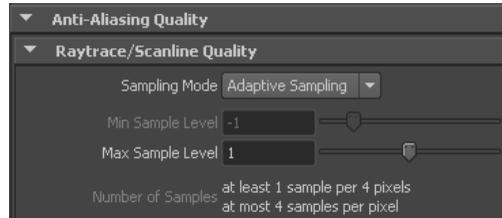
In this image, the circular spots that were seen in the previous renderings have disappeared. You can observe other Global Illumination effects compared to the original raytrace rendering:

- Many of the objects have received some additional color from neighboring objects in the scene as a result of the Global Illumination, an effect known as *color bleeding*. When color bleeding occurs, objects take on some of the color from neighboring objects as a result of the reflected and transmitted light.

Now that you've tested and refined the Global Illumination for a small image you can render a final image for the scene at a larger image size and increase the rendering quality settings.

To increase the image size and antialiasing

- 1 In the Render Settings window, select the Common tab and open the Image Size settings.
- 2 Set the preset to 640 X 480 to render a larger sized image.
- 3 In the Render Settings window, select the Quality tab.
- 4 Open the Anti-Aliasing Quality attributes. Leave the Sampling Mode as the default (Adaptive Sampling) and set the following:



- Number of Samples - Max sample level: 1

Sampling is the process for determining the amount of anti-aliasing that occurs. The Sample Level attribute sets minimum and maximum limits for the amount of sampling that will occur for a given pixel area in the image. Modifying the anti-aliasing *Sample Level* settings will ensure that the surfaces in the rendered image appear smooth.

A Max value of 1 should provide the correct amount of anti-aliasing sampling for this particular image at this image size.

To render a final image of the scene

- ▶ In the Render View window, click the Redo Previous Render icon to render the image.



Final rendering using Global Illumination

Beyond the Lesson

In this lesson you were introduced to the Global Illumination features provided by the mental ray® for Maya® renderer. You learned that:

- The mental ray for Maya renderer provides a feature that simulates indirect illumination in the Maya scene called Global Illumination.
- Global Illumination allows you to achieve realistic lighting effects for objects and areas of the scene that do not receive direct lighting.
- Photons are used to measure the Global Illumination of the scene via the creation of a photon map.
- The brightness of the Global Illumination in a scene is controlled by the Photon Intensity.
- The quality of the Global Illumination effect is dependent on the number of photons emitted into the scene, the Accuracy, and the Radius attribute settings.
- Multiple test renders are often required to achieve the desired final image.

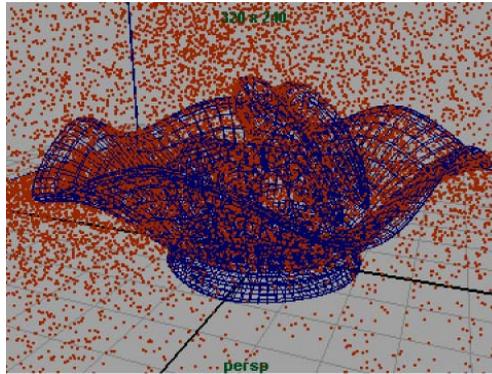
Light sources for Global Illumination

Point, spot, and mental ray area lights can be used as light types for Global Illumination with the mental ray for Maya renderer. A Directional light does

not work as a source for Global Illumination because this light type does not have a defined point of origin (that is, the rays emit in a parallel manner).

Photon maps

You can visualize the photon mapping that occurs at the beginning of the rendering process provided you turn on the Enable Map Visualizer option in the Caustics and Global Illumination > Photon Tracing section of the Render Settings window. To view a visualization of the photon mapping, select Window > Rendering Editors > mental ray > Map Visualizer.



Visualization of the photon map

Visualizing the photon mapping can aid in determining whether the coverage and density of photons for the scene is appropriate for the resulting final image.

By default, the mental ray for Maya renderer rebuilds the photon map each time you render a scene using Global Illumination. If you are rendering a scene containing camera animation in which the objects and lights remain static, or rendering several iterations of a single image, you may consider turning off the Rebuild Photon Map setting in order to reduce the rendering times for each frame.

Lesson 5: Caustics

Introduction



image created by: Vivien May

Caustics are the light effects and specular patterns that are cast on surfaces as a result of focused light reflecting off highly reflective surfaces or refracting through translucent surfaces.

The light patterns that occur on the bottom of a swimming pool on a sunny day, or the bright areas that occur in the shadow of a glass object as light shines through it are examples of caustics.

The Caustic feature in the mental ray® for Maya® renderer simulates caustic effects in your Maya scenes. Caustics are similar to Global Illumination in that they are an indirect illumination effect.

In this lesson you learn how to:

- Create the shadow effect that results from the refraction and concentration of light through a transparent object, using the Caustic feature provided in the mental ray for Maya renderer.
- Adjust refraction level limits in order to achieve desired refractive effects with transparent objects.

- Systematically perform test renders and adjust the render and caustic quality settings in order to achieve a high quality image.

Open the scene for the lesson

In this lesson, you work with a scene we created for your use. In the first steps of this lesson, you open the scene and view the various elements that have been preset for your use.

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 439.
- 2 Open the scene file named `CausticStart.ma`.

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

```
GettingStarted\Rendering\CausticStart.ma
```



The scene contains a still life with a translucent bottle and apples. Other surfaces act as the floor and walls for the scene. (The bottle appears opaque when viewed in shaded and textured display mode.)

Two lights illuminate the scene: a spotlight and a point light.

Shading materials have already been assigned to the models and surfaces. You can view the shaders and textures by opening the Hypershade editor.

When you initially render this scene, the objects cast shadows onto the floor surface. Caustics are added to the lighting of the scene to create a more realistic shadow for the translucent bottle.

Render the scene using raytracing

To better understand the differences between caustics versus other lighting, you begin by rendering the scene using the mental ray for Maya renderer without using Caustics.

You start by preparing the scene and then rendering. In this section you:

- Set a panel layout for rendering
- Turn on shadows for the spotlight
- Turn on the mental ray for Maya renderer
- Set the image size for your rendering
- Set the image quality preset
- Render the image using raytracing

Setting a panel layout for rendering

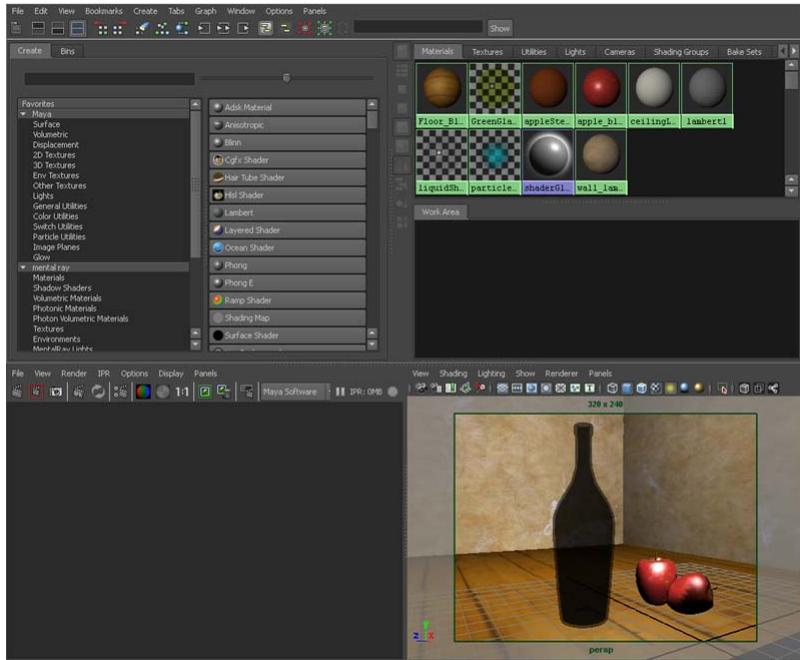
Producing a rendered image usually requires that you render test images to achieve the final result. This may involve adjusting many render quality, shading, and lighting settings. Opening and closing the various windows can become tedious.

Keeping the most frequently used windows and panels open allows you to work more efficiently.

In the following steps, you select a panel layout to work more efficiently with the various panels and windows.

To set a panel layout for rendering

- From the perspective view's panel menu, select Panels > Saved Layouts > Hypershade/Render/Persp.



The panel layouts update to display the Hypershade window, Render View window, and Perspective view simultaneously.

When you render an image, it appears in the Render View window.

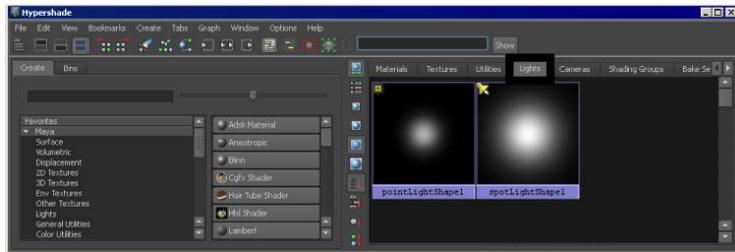
TIP You can resize the panels in a preset layout by dragging the border between neighboring panels using your left mouse button.

Turning on shadows

In this lesson, you learn how the bottle's shadow can be made to appear more realistic as a result of the caustics.

To turn on shadows for the spotlight

- 1 In the Hypershade window, click the Lights tab.



The Hypershade displays icons for the two lights in the scene.

- 2 In the Hypershade window, double-click the icon for spotLightShape1. The Attribute Editor appears.
- 3 In the Attribute Editor, click the spotLightShape1 tab to display the lighting attributes for the spotlight.
- 4 In the Shadows section, open the Raytrace Shadow Attributes and turn on Use Ray Trace Shadows.



When you render the scene, shadows are cast by the spotlight.

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



Using the mental ray for Maya renderer

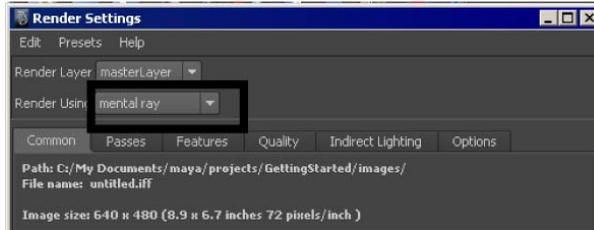
To use the Caustics feature in Maya you must render using the mental ray for Maya renderer.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.

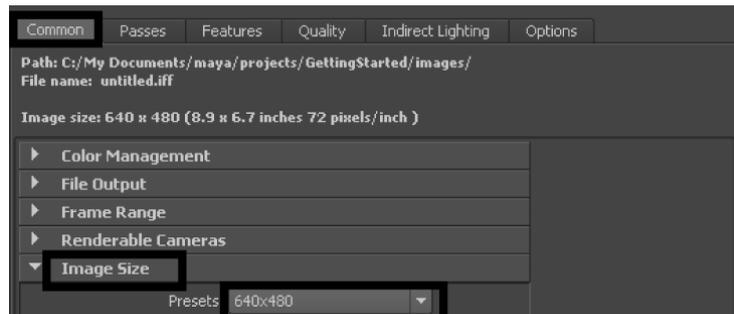


Setting the image size for rendering

You set the size that an image will be rendered in the Render Settings window. For this lesson, you render the image at a size that allows you to evaluate the caustic effect.

To set the image size for rendering

- 1 At the top of the Render Settings window, select the Common tab and open the Image Size settings.



- 2 Choose the 640 X 480 preset from the Presets drop-down list.

Setting render presets for raytracing

For the first rendering, you render the scene to see how the shadows appear without using caustics.

To set Render options for raytracing

- 1 In the Render Settings window, select the Quality tab and set the Quality Presets setting to Production.



The Production preset ensures that the refraction level settings (found in Raytracing Quality section) are at a level high enough to render objects with translucency (for example, the bottle). The Production preset also sets the anti-aliasing level for the image.

- 2 Close the Render Settings Window.

Rendering the image

When you render an image, all of the objects, lighting, shading materials, and image quality settings are used to calculate the image, from the camera's view.

To render an image using mental ray for Maya

- 1 Click in the perspective window to indicate which view you want to render.
- 2 In the Render View window, click the Render Current Frame icon.



The bottle scene renders using the mental ray for Maya renderer and the image appears in the Render View window.

TIP If the rendered image appears either too small or too large in the Render View window, select View > Frame Image from the Render View menu to fit the image to the Render View.

- 3 Once the render is complete, click the Keep Image icon in the Render View window.



Keeping the image allows you to compare it to any subsequent test renders. In this lesson you compare each successive rendered image to the previous one to observe how the changes you make to render settings affect the rendered image.



Raytrace rendering

The rendered image shows the following:

- A translucent bottle with apples beside it casting shadows towards the rear of the scene. Notice how the shadows appear when raytrace rendered.
- The distortion of the table surface in the translucent glass effect based on the refractions and reflections.
- A dark region on the surface of the bottle. This results from the bottle reflecting the empty black area of the scene in front of it.

In the next section, you turn on the caustics rendering attributes and render the scene.

Render the scene using caustics

To render using caustics you need to turn on specific settings or attributes. There are two sets of caustic-related attributes that you must initially set.

- Turn on the caustics-related rendering attributes using the Render Settings window.

- Turn on the caustics-related lighting attributes using the Attribute Editor.

You initially render the scene using these attributes at their default settings to see how they affect the rendered image and then compare it to the previous raytrace rendering. As you gain more experience rendering with Caustics you will better understand what settings to initially use.

To turn on the caustics preset settings

- 1 In the Render Settings window, select the Quality tab and set the Quality presets setting to Preview: Caustics.



This turns on the presets that control the quality of the caustics.

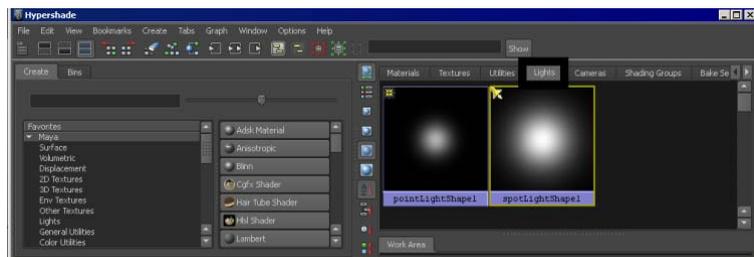
- 2 Close the Render Settings window.

Mental ray for Maya uses *photons* to simulate the caustic effects in a scene. Photons are small packets of energy emitted from a light source into the scene. The photons travel throughout the scene, refracting through transparent objects or reflecting off reflective surfaces. The renderer keeps track of the illumination that results from this photon emission in a file called a *photon map*. The overall caustic simulation is determined from the photon calculations that result.

In addition to having the spotlight emit light into the scene you set the light to also emit caustic photons. The overall caustic effect is determined from the photon calculations that result.

To turn on the caustics lighting settings

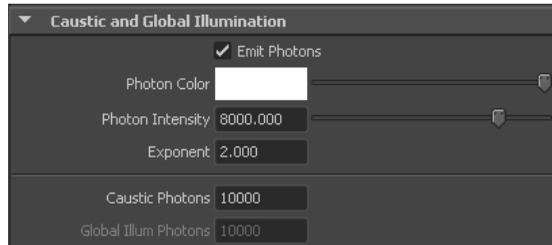
- 1 In the Hypershade, double-click on the icon for spotLightShape1.



- 2 In the Attribute Editor, click the spotLightShape1 tab to display the lighting attributes for the spotlight.

3 Open the mental ray section, and in the Caustic and Global Illumination attributes set:

- Emit Photons: On
- Photon Intensity: 8000
- Caustic Photons: 10000



4 Hide the Attribute Editor.

Rendering the scene with caustics

Now that the lights are set to emit caustic photons and the caustics preset is turned on, you're ready to render an image using caustics.

To render the scene using caustics

1 In the Render View window, click the Redo Previous Render icon.



2 When the image has completed rendering, click the Keep Image icon.

3 Compare this rendered image with the previously rendered image by dragging the scroll bar at the bottom of the Render View window left and right.



Raytrace rendering



Default caustic settings

This image differs from the previous one in the following ways:

- The bottle appears more opaque. This indicates that the correct number of refractions may not be occurring. The Raytracing Refraction Level setting may need to be increased. When a different preset was selected (Production to Preview), the refraction levels changed.
- There are spotted areas on the table surface directly in front of the bottle. This can be corrected by adjusting the photon quality settings.
- The desired caustic effect in the shadow has appeared but is not very pronounced. You want to achieve an effect where the shadow has brighter regions as a result of light being refracted through the surfaces of the bottle.

Adjusting the refraction levels

When the refraction levels are set too low, light rays do not refract through all of the surfaces enough to provide a realistic indication of a translucent surface when raytraced. This results in the object appearing opaque when it should appear translucent.

Increasing the *Refractions* setting allows the renderer to calculate what is seen through the surfaces of the glass by allowing the light rays to refract the correct number of times.

In the next step, you change the Refraction settings so that the bottle will appear translucent when rendered.

To increase the refraction levels

- 1 In the Render Settings window, select the Quality tab and then open the Raytracing attributes.

2 In the Raytracing section set the following:

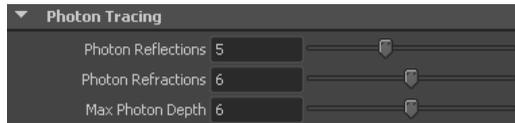
- Refractions: 6
- Max trace depth: 8

The *Refractions* setting sets the number of times a light ray can be refracted. When the setting is too low the ray does not penetrate through all of the surfaces. The *Max Trace Depth* setting limits the total Refraction and Reflection settings. By setting it to 8, you ensure that two reflection and six refraction calculations can occur.

Photons refract in a manner similar to light rays when rendering. Increasing the Refractions setting for the Caustic Photons ensures that the renderer can correctly calculate the caustics that should occur with the bottle. In this case, matching the Caustic Photon refraction settings with the Raytracing refractions is a good practice.

To increase the photon refraction settings

- 1 In the Render Settings window, select the Indirect Lighting tab.
- 2 In the Photon Tracing section set the following:
 - Max Photon Depth: 6
 - Photon Refractions: 6



The *Max Photon Depth* setting limits the Refraction and Reflection calculations for the Caustic Photons to the limit set. The *Photon Refractions* setting is similar to the Refractions setting for raytracing but sets the refractions for the Caustic photons. Close the Render Settings window.

Increasing the brightness of caustics

Increasing the Photon Intensity setting increases the brightness of the refracted caustic that appears in the shadow of the bottle.

To increase the Photon Intensity

- 1 In the Hypershade window, double-click the icon for `spotLightShape1` to display the attributes for the spotlight.

- 2 In the Caustic and Global Illumination attributes for the light adjust the settings to the following:
 - Photon Intensity: 36000

To render the scene

- 1 In the Render View window, click the Redo Previous Render icon to render the image.
- 2 Once the render is complete, click the Keep Image icon in the Render View window to save this image.
- 3 Compare this rendered image with the previously rendered images.



Refractions = 2
Photon Intensity = 8000



Refractions = 6
Photon Intensity = 36000

The translucency of the glass now appears more realistic. The caustic effect is more evident in the shadow of the bottle. However, the spotted effect, that appeared on the surface in front of the bottle, is more pronounced. These spots are the result of a low caustics quality setting.

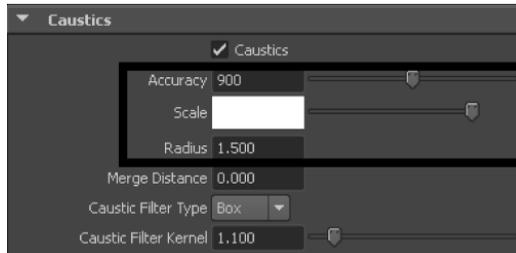
Adjusting the quality of caustics

The quality of the caustic effects in a scene is primarily affected by the number of photons, the accuracy of sampling, and the radius surrounding the sampled area of illumination. (If you have previously completed the Global Illumination tutorial, you may notice similarities in this area.)

In the next steps, you return to the Render Settings window and the Attribute editor to increase the quality attributes that affect the caustics effect.

To increase the caustics quality settings

- 1 In the Render Settings window, click on the Indirect Lighting tab and expand the Caustics section. Change the following Caustics attributes to:
 - Accuracy: 900
 - Radius: 1.5



The *Accuracy* determines how many photons will be evaluated to determine the Caustic intensity at a given sampling point. *Radius* determines the size of the region surrounding the particular sampling point. The more photons that get sampled, the more realistic the caustic effect appears in the final image.

- 2 Close the Render Settings window.

Increasing the number of caustic photons that get emitted into the scene also increases the quality of the caustic effect. When you increase the number of caustic photons you increase the density of the resulting photon map that gets produced at the beginning of the render process. The more photons, the smoother the resulting caustics effect.

NOTE As you increase the caustic quality and photon settings so will the render time. Your goal should be to increase these settings just enough to achieve the desired image quality, without increasing them beyond what is required so that your rendering times are not adversely affected. Performing test renderings is the best method to determine these settings.

To increase the number of caustic photons

- 1 In the Hypershade window, double-click the icon for spotLightShape1.

- 2 In the Caustic and Global Illumination attributes, adjust the following settings:
 - Caustic Photons: 50000
- 3 In the Render View window, click the Redo Previous Render icon.
- 4 When the image has completed rendering, click the Keep Image icon and compare this rendered image with the previously rendered images.



In this image the bright spots in front of the bottle should be reduced when compared to the previous images. The caustic in the bottle's shadow should also appear more pronounced and more evenly illuminated.

Beyond the Lesson

In this lesson you were introduced to the caustic features provided by the mental ray® for Maya® renderer. You learned that:

- Caustics provide additional realism in a rendered image by simulating the focusing of light on diffuse surfaces as a result of the light reflecting off specular surfaces or via the refraction of light through translucent surfaces.
- Photons are used to determine the caustic effect in a scene via the creation of a photon map.
- The brightness of the caustic is controlled by the Photon Intensity attribute.

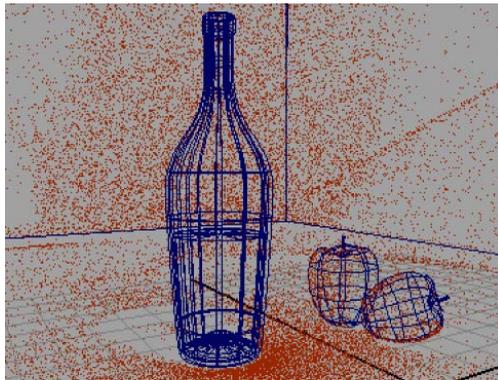
- The quality of the caustic effect is controlled by the number of photons emitted into the scene, the Accuracy, and the Radius attributes.
- Multiple test renders are often required to achieve the desired final image.

Light sources for caustics

Point, spot, and mental ray area lights can be used as light types for caustics with the mental ray for Maya renderer. A Directional light does not work as a source for Caustics because this light type does not have a defined point of origin (that is, the rays emit in a parallel manner).

Photon maps

You can visualize the photon mapping that occurs at the beginning of the rendering process provided you turn on the Enable Map Visualizer option in the Caustics and Global Illumination section of the Render Settings window. To view a visualization of the photon mapping, select Window > Rendering Editors > mental ray > Map Visualizer.



Visualization of the photon map

Visualizing the photon mapping can aid in determining whether the coverage and density of photons for the scene is appropriate for the resulting final image.

By default, the mental ray for Maya renderer rebuilds the photon map each time you render a scene using Caustics. If you are rendering a scene containing camera animation in which the objects and lights remain static, or rendering several iterations of a single image, you may consider turning off the Rebuild Photon Map setting in order to reduce the rendering times for each frame.

Introduction

Dynamics is a branch of physics that describes how objects move using physical rules to simulate the natural forces that act upon them. Dynamic simulations are difficult to achieve with traditional keyframe animation techniques.

Maya provides a means to do this type of computer animation where you set up the conditions that you want to occur, then let the software solve how to animate the objects in the scene. Maya® Dynamics™ provides the tools for creating effects that will enhance your final images and animations.

With Particle Effects, you can create the illusions of smoke, fireworks, rain, fire, and explosions.

With Rigid Body Dynamics, you can simulate real-world physical interactions between objects, such as collisions between surfaces. For example, you can simulate a bowling ball crashing through pins or simulate the effects of gravity when a ball falls to the ground. You can also simulate natural forces, such as wind.

This chapter contains the following lessons:

- Lesson 1 Particles, emitters, and fields: [Introduction](#) on page 536
- Lesson 2 Rigid bodies and constraints: [Introduction](#) on page 554

Preparing for the lessons

Do these steps to ensure the lessons in this chapter work as described:

- 1 Select File > New Scene to create a new scene before you start each lesson.

- 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](#) on page 25.
- 3 Select Window > Settings/Preferences > Preferences. Click Time Slider under Categories and make sure the Playback Speed is set to Play every frame. Dynamic animation plays more accurately with this setting.
- 4 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.

NOTE Before you perform the lessons in this book, ensure that the Interactive Creation option for primitives is turned off by selecting Create > NURBS Primitives > Interactive Creation. That is, ensure a check mark does not appear beside the menu item.

Lesson 1: Particles, emitters, and fields

Introduction

Particles are points that display as dots, streaks, spheres, or other shapes. Particles can have attributes applied to them so they animate and render to simulate natural phenomena.

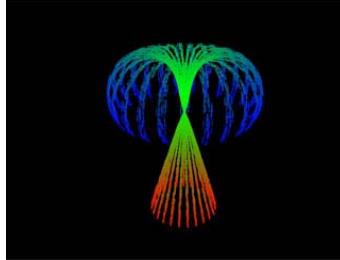
You can create particles by clicking positions in the scene view, or you can create them with an emitter that shoots particles into view.

To animate particles, you typically apply fields such as gravity or wind. By combining emitters, particles, and fields, you can create natural phenomena such as smoke, fireworks, or rain.

In this lesson, you will be introduced to some of the basic concepts of particles and use particles, emitters, and volume fields to create a fountain-like effect. In this lesson, you learn how to:

- Create a particle emitter in your scene.
- Create a field to direct the movement of particles.
- Use a volume shape to control the movement of particles.

- Work with particle attributes.
- Color particles.
- Use a color ramp to add colors to the particles.
- Render the particles using mental ray for Maya rendering.



Creating an emitter

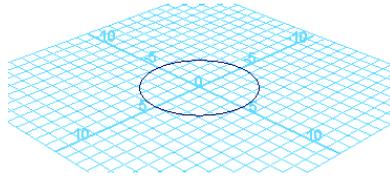
Emitters create particles as an animation plays. The emitter controls the position, direction, quantity, and initial velocity of the emitted particles when they are born. After particles are born, the attributes of the particle object control their appearance.

You can make a standalone emitter, or you can make a surface, curve, and even another particle an emitter. You move an emitter the same way you move any geometry, such as setting keys or parenting it to a moving object.

In the next steps, you create a circle and emit particles from it.

To create a particle emitter

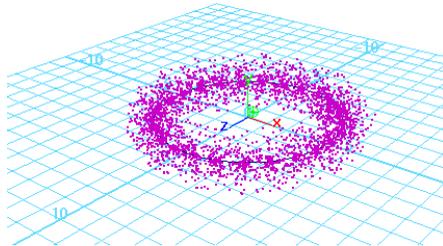
- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 535.
- 2 Set the start frame to 1 and end frame to 120.
- 3 Select Create > NURBS Primitives > Circle > . In the options window, set the following options and click Create:
 - Radius: 4
 - Number of Sections: 25



With Radius set to 4, the circle will be wide enough to see the particle emission clearly. With Number of Sections set to 25, the emission will be distributed evenly from points around the circle.

Note that the illustrations in this chapter show the grid with a custom blue color and numbered lines so that the shape and size of the objects you create are easier to distinguish. (If you want to change the grid color in your scene, select Display > Grid > .)

- 4 With the circle selected, select Particles > Emit from Object.
By default, this creates an Omni emitter. If you dolly toward the center of the grid, you can see a tiny icon that represents the emitter. When you create an Omni emitter on a NURBS object, for instance, a circle, each control vertex (CV) of the object emits particles in all directions.
- 5 Play the animation for a few seconds to see the emission.



The particles begin to emit from the object. When you create any type of emitter, a particle object is automatically created and connected to it. The connected particle object originally has no particles and is therefore invisible. As the animation plays and the emitter generates particles, the particle count of the connected particle object increases. When you go to the start of the animation, the particle count returns to 0.

Note that if you had created the circle with a smaller Number of Sections, there would be fewer CVs for the circle and therefore bigger gaps between the emission points on the circles.

6 Display the Outliner (Window > Outliner).

7 In the Outliner, select particle1.

This is the particle object. A particle object is a collection of particles that share the same attributes. You can create particle objects containing a single particle or millions of particles. Each particle in a scene belongs to some particle object.

The particle object's attributes define the appearance and other characteristics of the emitted particles. The emitter's attributes control the initial position, direction, quantity, and velocity of the emitted particles.

Creating volume axis fields

Fields are forces that you use to animate the motion of particles. With a Volume Axis field, you can move particles in various directions within the volume of a cube, sphere, cylinder, cone, or torus. With clever use of this type of field, you can create effects such as particles flowing around obstacles, solar flares, explosions, tornadoes, and rocket exhaust.

In the next steps, you'll use a volume axis field to move the particles up a conical path. When the particles exit the top of a conical region, you'll use another volume axis field to give them a downward arcing motion within a torus. The result resembles a fountain of light.

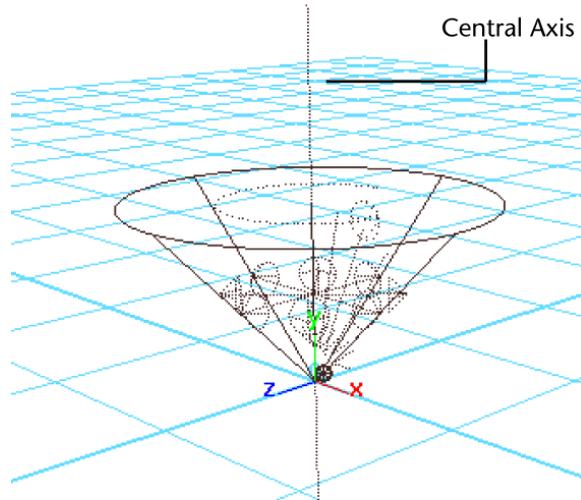
To create a cone volume axis field

- 1 With the particle object still selected, select Fields > Volume Axis > .
- 2 Set the following Volume Axis Options and click Create:
 - Magnitude: 50
 - Volume Shape: Cone
 - Away From Axis: 0
 - Along Axis: -1

NOTE If any of these options are not visible, enlarge the Volume Axis Options window.

This creates a conical Volume Axis field that affects the motion of particles that enter or exist within its volume. Particles move in relation to a central

axis within the volume. For a Cone Volume Shape, the central axis is the positive Y axis:



Using the axis attributes, you can specify that the particles:

- move in a direction roughly parallel to the axis toward or away from the point of the cone (Along Axis)
- radiate outward or inward relative to the axis (Away From Axis)
- swirl around the axis (Around Axis)

- move in a direction based on a combination of these directions
By setting Along Axis to -1, the particles move toward the point of the cone. (A positive value would move particles in the opposite direction.) In the next step, you'll flip the cone 180 degrees so the aim of the cone and particles is upward.

By using 0 for the Away From Axis value rather than the default value of 1, the particles will not radiate away from the axis.

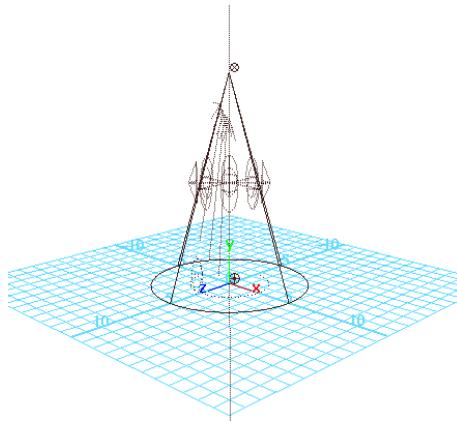
Note that you can use larger or smaller values for the Axis attributes to intensify or diminish movement.

Magnitude specifies the strength of the volume axis field. The greater the value, the more force is exerted on the particles.

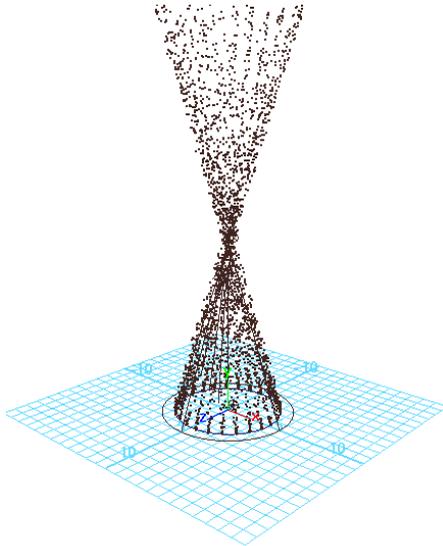
You'll typically need to experiment with various Volume Axis field attribute values to achieve a desirable motion for particles. Be aware that small arrow-shaped icons within the Axis Field icon show which direction the particles will go.

- 1 With the Volume Axis field still selected, set the following values in the Channel Box:
 - TranslateY: 13
 - RotateZ: 180
 - ScaleX: 5
 - ScaleY: 13
 - ScaleZ: 5

This positions, rotates, and scales the volume axis field so that its base lies on the same plane as the circle emitter. Even though the circle emits particles in random directions, the volume axis field will redirect their motion up the cone.



- 2 Go to the start of the playback range and play the animation for a few seconds.

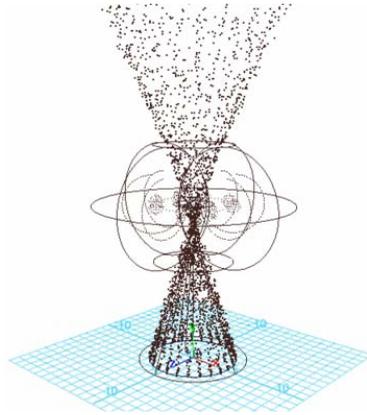


When the particles move out of the conical region, they continue in the same direction, mirroring the pattern of movement within the conical region. The volume axis field no longer controls the motion, but the particles continue with the same speed and direction they had at the moment they exit.

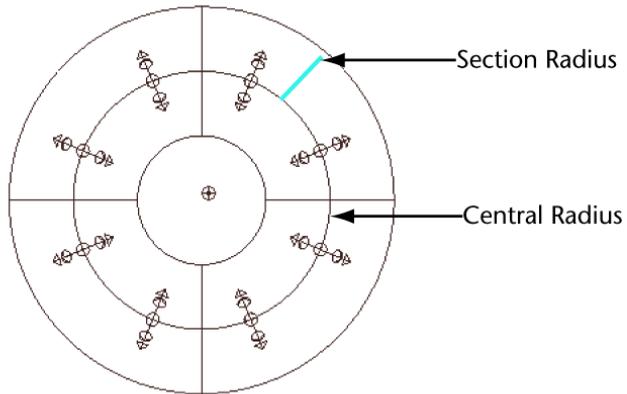
To create a torus volume axis field

- 1 In the Outliner, select the particle object
- 2 From the main menu, select Fields > Volume Axis >  .
- 3 Set the following Volume Axis Options and click Create:
 - Magnitude: 25
 - Volume Shape: Torus
 - Section Radius: 1.5
 - Away From Axis: - 0.7
 - Along Axis : 0.0
 - Around Axis: 2

The Torus Volume Shape creates a volume shaped like a doughnut.



Section Radius defines the thickness of the torus.



A value of 2 for Around Axis causes the particles to swirl around the central axis. For a torus, the central axis is the ring in the center of the solid part of the torus.

A value of -0.7 for Away From Axis causes the particles to move radially toward the central axis.

When you later play the animation, the combination of these values cause the particles to loop up then down as if guided along the inner walls of a torus.

Again, achieving the desired motion for particles within a volume axis field requires experimentation with attribute values. Examine the arrows

and other small icons within the axis volume field to help get the desired results.

Note that the field's effect is pervasive enough that all particles passing through the hole of the torus are influenced by the field. You can cause particles to pass through the hole by altering the Max Distance and Attenuation.

- 4 With the torus volume field still selected, enter the following values in the Channel Box:

- TranslateY: 15

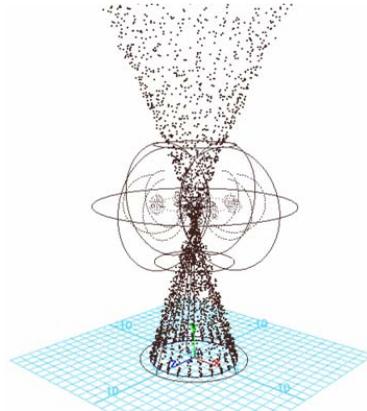
- ScaleX: 3.6

- ScaleY: 3.6

- ScaleZ: 3.6

This moves the torus volume field up 15 units and scales it larger. Finding the appropriate position for the volume axis field is a matter of experimentation.

- 5 Go to the start of the playback range and play the animation.



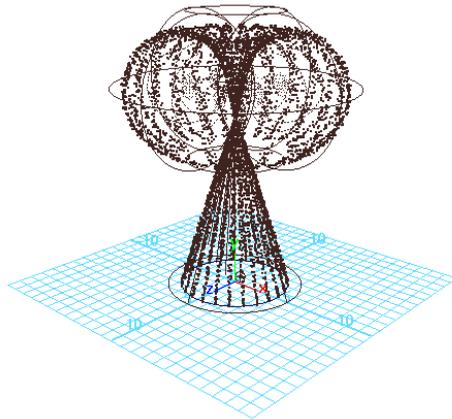
When the particles move through the top of the cone region, they spread outward somewhat but do not follow the shape of the torus volume field as specified in the prior steps. You'll correct this situation in the next steps.

Adjusting the velocity of moving particles

The torus volume axis field doesn't guide the motion of the particles as expected because the initial emission velocity of the particles dominates the motion of the particles rather than the field. You can use a particle object's Conserve attribute to scale down the emission velocity's influence and increase the field's influence.

To adjust the velocity of moving particles

- 1 Select the particle object and set the Conserve attribute to 0.8 in the Channel Box.
By changing the Conserve value from the default value, 1, to 0.8, the emission velocity's influence on the particle motion decreases and the field's influence increases.
- 2 Go to the start of the playback range and play the animation. Notice how the particle movement now conforms to the shape of the torus volume field.



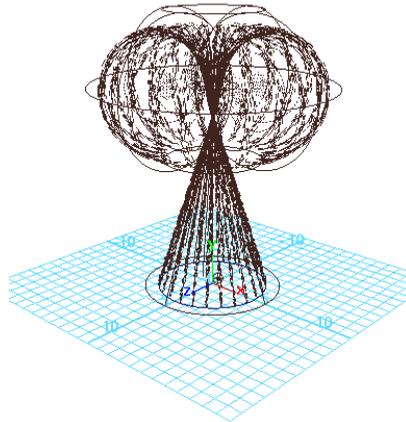
Setting the particle render type

The *particle render type* of a particle object determines how the particles look when rendered. For example, you can display particles as small spheres, streaks, or 2D images (known as sprites) of your favorite snapshot. Once you select

the particle render type, you can add attributes specific to the render type to adjust the appearance.

To set the particle render type

- 1 With the particle object still selected, display the Attribute Editor, and select the particleShape1 tab.
- 2 Scroll down to the Render Attributes section, and set the Particle Render Type to Streak.
- 3 Go to the start of the playback range and play the animation.



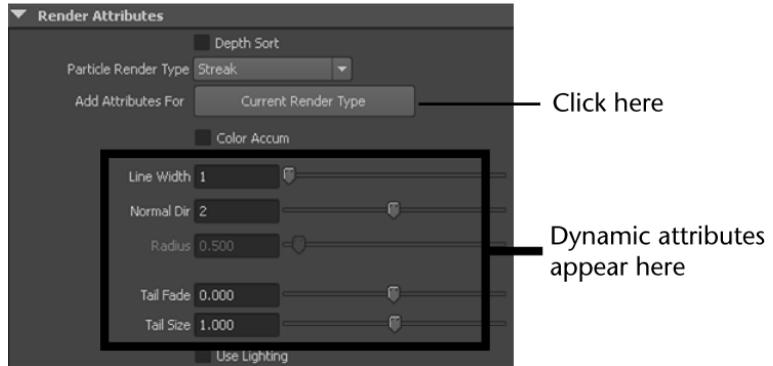
The Streak render type displays moving particles with an elongated tail. This render type enhances the display of objects such as meteors, and it's often used to create rain. The streak length is based on the velocity of the particle, so if the particles are stationary or moving slowly and the render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

Adding dynamic attributes

To maximize the display versatility of particle render types, you need to add built-in dynamic attributes to the particle object. When you add a dynamic attribute to an object, the attribute appears in the Attribute Editor for the selected object.

To add dynamic attributes

- 1 In the Attribute Editor, under the Render Attributes section, click the Current Render Type button next to Add Attributes.
- 2 Observe the attributes that are added.



- 3 Set the following attributes:

Line Width	2	This sets the width of each streak.
Tail Fade	0.5	This sets the opacity of the tail fade; it can range from 0 to 1. A value of 1 makes the tail completely opaque; a value of 0 makes the tail transparent.
Tail Size	2	This scales the length of the tail. A value of 1 is the default length. Values less than 1 shorten the tail; values greater than 1 lengthen the tail.

You can experiment with these settings to change the look of the particles.

Adding per particle attributes

Particle objects have two different kinds of attributes: per object and per particle.

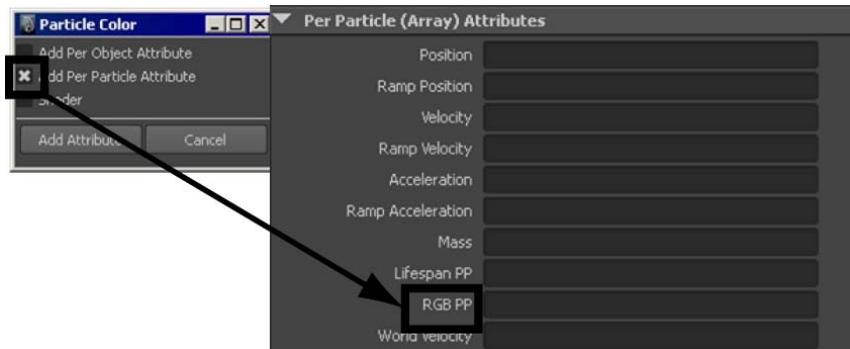
A per object attribute lets you set the attribute value for all particles of the object collectively with a single value. For instance, the per object color attribute lets you set a single color for all the particles in the object.

A per particle attribute lets you set the value of the attribute individually for each particle of the object. For example, the per particle rgbPP attribute lets you set a unique color for each particle. Though there is only one rgbPP attribute in a particle object, the attribute holds the value for each particle's color value. The attribute holds the values in an *array*. An array is similar to a list. Arrays are appropriate for storing data which must be accessed in an unpredictable order, in contrast to lists which are best when accessed sequentially.

Although per particle attributes are best for creating complex effects, you can't keyframe them. You can keyframe per object attributes. In this lesson, you add a per particle color attribute so you can give different particles different colors.

To add a per particle color attribute

- 1 In the Attribute Editor, expand the Add Dynamic Attributes section, if necessary.
- 2 Click the Color button. The Particle Color box is displayed,
- 3 Turn on the Add Per Particle Attribute and click Add Attribute. Notice the rgbPP attribute that is added to the Per Particle (Array) Attributes section of the Attribute Editor.



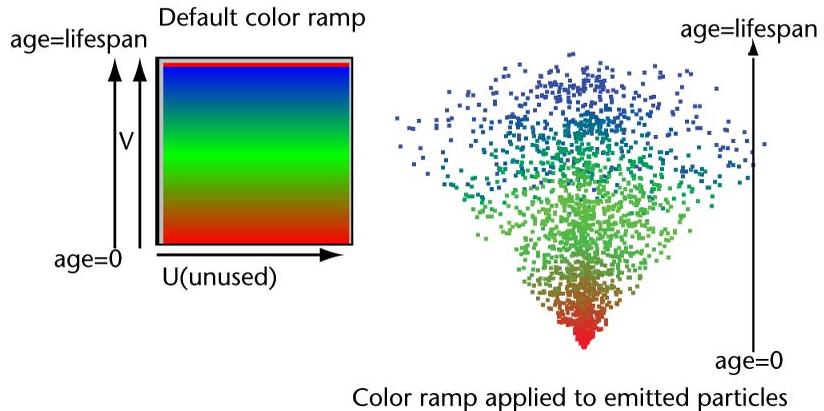
Adding color to particles with a color ramp

A ramp texture is a patterned 2D image in which the color blends from one shade to another. In the next steps, you use a color ramp to color the particles as they age.

To add color to particles using a color ramp

- 1 In the Attribute Editor, right-click the rgbPP data box and select Create Ramp from the drop-down menu.

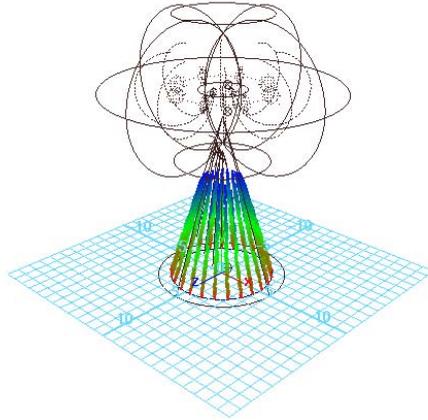
This causes a default color ramp to color the particles as follows:



As each particle ages from its time of birth (emission), its color changes according to the ramp's color in an upward vertical (V) direction. The horizontal (U) direction of the ramp's color is unused in this lesson.

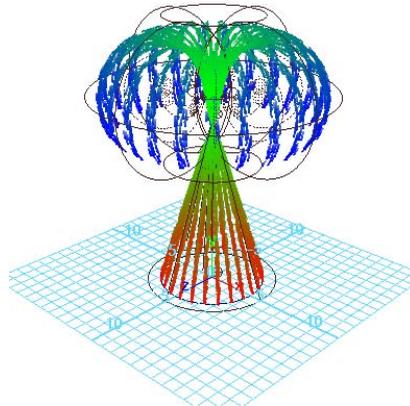
When you create the ramp, Maya adds a lifespan attribute and sets it to 1 second, by default. Each particle changes from red to green to blue over its lifespan of 1 second. The particle then disappears (*dies*).

- 2 From the perspective window menu, select Shading > Smooth Shade All. This enables you to see the colors of the particles the next time you play the animation.
- 3 Go to the start of the playback range and play the animation. Notice the particles never reach the end of the cone volume.



This is because the Lifespan is set to the default of 1, which is the number of seconds the particles exist in the animation.

- 4 In the Channel Box, set Lifespan to 3. Go to the start of the playback range and play the animation.



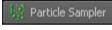
Because the particles live longer, they pass all the way through the cone and part of the way through the torus. The particles change from red to green to blue and disappear as they are born, age, and die.

mental ray rendering particles

In 3D animation, rendering typically refers to the act of creating a sequence of high-quality image snapshots for each frame of an animation sequence. After rendering the images, you play them in sequence to create a film or video clip. If the concept of 3D animation rendering is new to you, consider doing the lesson [Lesson 1: Rendering a scene](#) on page 439 before completing this section.

Before you can render the particles using mental ray, you must add the particleSamplerInfo node to your shading network so that mental ray can access the attributes of your particle object such as color information and Lifespan.

To add the particleSamplerInfo node to your shading network

- 1 Select Window > Rendering Editors > Hypershade to open the Hypershade window.
- 2 With the particle object selected, click the  button to graph its shading network.
- 3 In the shading network, select the Lambert shader.
- 4 Click the Map  button beside the Color attribute in the Lambert shader Attribute Editor. The Create Render Node window appears.
- 5 In the Create Render Node window, under Maya, click Utilities.
- 6 In the right panel, click Particle Sampler, .

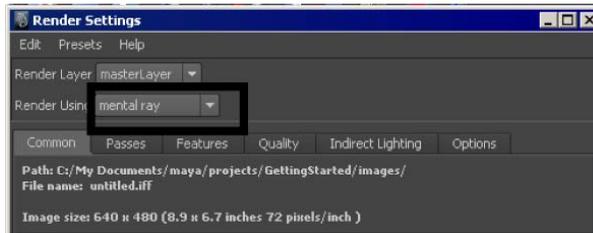
A particleSamplerInfo node now appears in the shading network and connected to the Lambert shader node.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.



To batch render the sequence

- 1 In the Render Settings window, select the Common tab.
- 2 In the File Output section, set the following options:
 - File Name Prefix: Type the name Emit. This name will be the base of the filenames created by batch rendering.
 - Image Format: Select Maya IFF (.iff), Maya's standard image file format. You can use the .iff format for any further work you need to do, including previewing and compositing the animation. If you require a different format, you can specify it instead of .iff in the Render Settings.
 - Frame/Animation ext: Select name.#.ext. This specifies that the filenames will have the format `prefix.frameNumber.fileFormat`. For example, batch rendering the entire 120-frame animation will create `Emit.0001.iff`, `Emit.0002.iff`, and so on through `Emit.0120.iff`.
 - Frame padding: Enter 4. This causes the frameNumber part of the filenames to be four digits prefixed with 0s. For example, the filenames will be `Emit.0001.iff` and `Emit.0002.iff` and so forth.
- 3 In the Frame Range section, set the following options:
 - Start frame: Enter 1, the first frame of the animation sequence to be batch rendered.
 - End frame: Enter 75, the last frame to be batch rendered. (Rendering all 120 frames may be time-consuming.)
- 4 Close the Render Settings window.
- 5 Select Render > Batch Render to render the sequence.
Rendering is complete when the following message appears:

```
// Result: Rendering Completed. See mayaRenderLog.txt for  
information. //
```

To use FCheck to view the sequence of rendered frames

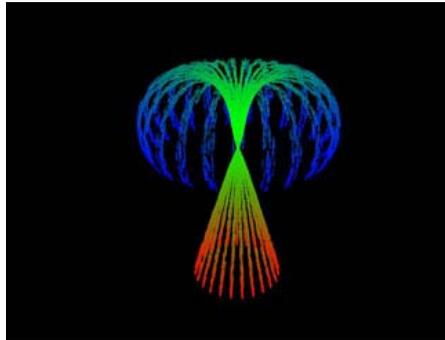
- 1 Select File > View Sequence.

The File Browser window appears.

- 2 Using the File Browser, navigate to the images directory for your current project (or wherever you saved the rendered sequence of images for the Apple).

- 3 Click the image file Emit.0001.iff to select the first image in the sequence and then click Open.

The FCheck image viewing utility appears and the rendered sequence of 75 frames plays back as an animated loop.



Beyond the lesson

In this lesson you were introduced to a few of the basic concepts when using particles. You learned how to:

- Create particles from a source object called an emitter.
When you create any type of emitter, a particle object is also automatically created and connected to it. Emitters can be points (CVs, vertices), surfaces (NURBS, polygons), curves (NURBS curves) or volumes (spheres, cylinders).
- Control particles using fields, and volume objects.
With volume axis fields, you can funnel or swirl particles within the boundaries of common volumetric shapes. Although this lesson showed how to use a pair of volume axis fields to control the motion of particles,

Maya has several other types of fields such as Gravity and Turbulence for simulating natural phenomena. You can use the fields to animate the motion of curves and surfaces in addition to particles.

- Change the color of particles using a ramp texture.
You can also change ramp colors by editing other ramp attributes, such as Noise and Noise Frequency. To do so, right-click the attribute box for rgbPP, then select `arrayMapper1.outColorPP > Edit Ramp`. See the Maya Help for details on ramp attributes. You can also control other particle attributes with ramps; for example, opacity. See the Maya Help for more information.
- Render particles using the hardware renderer.
The majority of particle rendering types are visible only when using the hardware render. Software rendering does not display them. However, if you use a particle render type of Bloppy Surface, Cloud, or Tube, you must render the particles using the software renderer as hardware rendering does not display those types.
If you create a scene that includes both particles and geometric surfaces, you may need to render the scene twice, once with hardware rendering and once with software rendering. In such cases, you'll need to composite the two sets of rendered images using a compositing software package.

Lesson 2: Rigid bodies and constraints

Introduction

In Maya, a *rigid body* is a surface which has the attributes of an unyielding shape. Unlike conventional computer surfaces, rigid bodies collide rather than pass through each other during animation and are used for creating dynamic simulations. As well, rigid surfaces do not deform when they collide with other objects in Maya.

Rigid bodies can be created from polygonal or NURBS surfaces. Rigid body attributes can be assigned to the surfaces so they act and react in a simulation in specific ways. Examples of rigid body attributes are velocity, mass, and bounce.

To animate rigid body motion, you can use fields, keyframes, expressions, rigid body constraints, or collisions with particles or other rigid bodies.

Rigid body *constraints* restrict the motion of rigid bodies. The constraints simulate the behavior of real-world items you're familiar with, such as pins, nails, barriers, hinges, and springs.

In this lesson, you create a rigid body dynamic simulation in which you drop a ball on some planks and watch the simulation in action. In this lesson you learn how to:

- Create objects with physical properties.
- Apply hinge constraints to objects so they react in specific ways.
- Set attributes for the rigid bodies to be either active or passive for the simulation.
- Play back a rigid body dynamic simulation.

Lesson setup

You can create rigid bodies from NURBS surfaces and polygonal surfaces. In the following steps, you create some polygonal surfaces to be used for the rigid bodies.

To create polygonal objects for the rigid body simulation

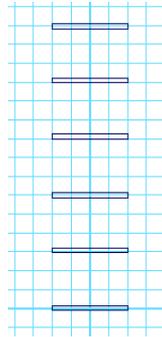
- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 535.
- 2 In the Time and Range Slider, set the start frame to 1 and end frame to 200.
- 3 Select Create > Polygon Primitives > Cube > .
- 4 In the Polygon Cube Options window, select Edit > Reset Settings (to make sure the tool is set to its default, and then set the following options and click Create:
 - Width : 2
 - Height: 4
 - Depth: 0.25
 - Axis: Z
- 5 With the polygon still selected, select Edit > Duplicate Special > .

6 In the Duplicate Special Options window, select Edit Reset Settings. Set the following options and then click Duplicate Special:

- Translate: 0 3 0

- Number of copies: 5

This creates five copies of the polygon, each one placed 3 units above the lower one.



To position the objects for the rigid body simulation

► Position the objects as shown in the side view in the following illustration. To position the planks precisely, select each plank individually and use the Channel Box to enter the appropriate value from this list:

- pCube1 - Translate Z: 0

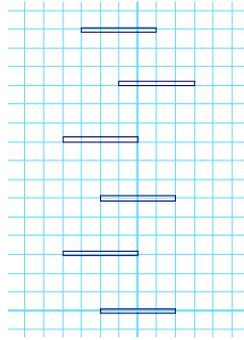
- pCube2 - Translate Z: 2

- pCube3 - Translate Z: 0

- pCube4 - Translate Z: 2

- pCube5 - Translate Z: -1

- pCube6 - Translate Z: 1



Creating hinge constraints

A Hinge constraint constrains rigid bodies along a specified axis. You can use a Hinge constraint to create effects such as a door on a hinge, a link connecting train cars, or a pendulum of a clock. In this example, the Hinge constraint constrains the plank to a position in the scene view.

In the following steps, you'll add Hinge constraints to the center of the planks, which automatically turns the plank into a rigid body. You will do this for each plank. By default, Maya automatically creates a rigid body when you use a field or rigid body constraint to control its motion.

To create a hinge constraint

- 1 Select **Soft/Rigid Bodies > Create Hinge Constraint** > .
- 2 In the **Constraint Options** window, set the following options:
 - Initial Orientation X, Y, Z: 0 90 0
 - Leave the options window open.

By default, the initial orientation is set to 0, 0, 0, which orients the hinge constraint to lie parallel to the Z-axis in world space. By setting the orientation to 90 in the Y-axis, the hinge constraint is rotated 90 degrees around the Y-axis, which orients the constraint parallel to the X-axis.



By leaving the Set Initial Position at its default setting, the hinge constraint is created at the rigid body's center of mass.

- 3 Select pCube1 and click Apply in the Constraint Options.
- 4 Select pCube2 and click Apply in the Constraint Options.
- 5 Repeat this until each of the six planks has a constraint applied to it.
- 6 Close the Constraint options window.

Running a dynamics simulation

In the next steps, you create a ball and add gravity to make the ball drop and hit the planks and then playback the simulation.

To create a ball for the simulation

- 1 Select Create > Polygon Primitives > Sphere > .
- 2 Set Radius to 0.2 and click Create.
- 3 In the Channel Box, set the Sphere's Translate Y value to 18.

To add gravity attributes to the ball

- With the sphere still selected, select Fields > Gravity.
When you connect Gravity to the sphere, it automatically makes the sphere an active rigid body. Gravity affects only the object that was selected when you add the Gravity field.

To run the dynamics simulation

- In the Time and Range slider, play the animation.
The ball falls and collides with the planks, which rotate about the hinge constraint.

Changing an active rigid body to passive

Maya has two kinds of rigid bodies—active and passive. An active rigid body is animated by dynamics—fields, collisions, and springs—not by keys.

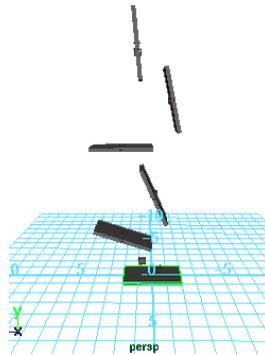
Because the ball and planks are all active rigid bodies, gravity pushes it down and it rebounds after colliding with the planks.

A passive rigid body can have active rigid bodies collide with it. You can key its Translate and Rotate attributes, but dynamics have no effect on it.

In the previous steps, all the planks were active rigid bodies and reacted to the collision of the ball. In the following steps, you'll change the bottom plank to be a passive rigid body so the ball bounces off the plank but the plank doesn't react to the collision.

To change an active rigid body to a passive rigid body

- 1 Select pCube1 and display the Attribute Editor.
- 2 Click the rigidBody1 tab to display the rigid body attributes.
- 3 In the Rigid Body Attributes section, turn off Active to make pCube1 a passive rigid body. Hide the Attribute Editor.



- 4 Go to the start of the playback range and play the animation.
Observe how pCube1 no longer reacts to the impact of the ball, but the ball continues to bounce off it.

Beyond the lesson

In this lesson you learned some fundamental concepts with respect to rigid body dynamics.

- There are differences between active and passive rigid bodies. When a rigid body is passive, active rigid bodies can collide with it but will not move it. You can control its movement by setting keys.

When a rigid body is active, you control its movement using dynamic forces, specifically fields and collisions. You can change the active and passive state of the rigid body to achieve various effects. For example, you can roll a ball off a table using keys (passive rigid body) then cause the ball to fall using gravity (active rigid body).

- A hinge constraint can be used between an active rigid body and a position in the scene view.

You can also create a hinge constraint between a passive rigid body and a position in the scene view, two rigid bodies, and an active and passive rigid body.

Other types of constraints include pin, nail, spring and barrier constraints. For additional information on the types of constraints and their usage, please refer to the Maya Help.

Introduction



Image created by: Martin Crawford

Maya provides painting tools that go above and beyond what many 3D creative artists might normally envision. These tools allow you to accomplish a wide range of modeling, animation, texturing, and effects work.

The painting tools are separated into three categories: Maya® Artisan™ brush tools, Maya® Paint Effects™ tools, and 3D Paint. If you completed the previous lessons in Getting Started with Maya, you already have experience with a few of these tools.

The Artisan brush tools have a variety of applications and allow you to directly modify attribute values interactively:

- In the NURBS modeling lesson, you used the brush tools to sculpt a simple face from a sphere.

- In the Character Setup chapter, you used Artisan brush tools to modify the skin weights on a surface mesh to ensure the character's joint deformed properly.
- In the Character Setup chapter, you used the Artisan brush tools to modify the cluster weights for Control Vertices to ensure a blend shape deformer worked predictably.

As you have seen, Artisan brush tools can also be used to select surface components. This selection method can dramatically improve the speed of your workflow.

Maya Paint Effects allows the user to create natural and fantastic effects either on a 2D canvas or as actual 3D objects in the scene.

- Paint Effects enables you to create a 2D backdrop for your scene, or a dynamic 3D paint effect such as a field of grass, with flowers and a wind-blown tree—all at the drag of your mouse or tablet pen.
- Paint Effects comes with a suite of preset brushes to enable you to create many types of 2D and 3D effects.

With 3D paint, you can paint and modify textures directly onto 3D surfaces using both the Artisan and Paint Effects tools. You can paint texture attributes such as color, transparency, and bump, to name but a few.

As you have already gained some experience with the Artisan brush tools, this chapter focuses on Paint Effects and 3D Paint. It includes the following lessons:

- Lesson 1 Painting in 2D using Paint Effects: [Introduction](#) on page 563
- Lesson 2 Painting in 3D using Paint Effects: [Introduction](#) on page 575
- Lesson 3 Painting textures on surfaces: [Introduction](#) on page 610

Preparing for the lessons

To ensure the lessons work as described, do the following:

- 1 Consider doing the lessons in the Rendering section of this guide. Familiarity with Maya rendering concepts is important to understanding the following lessons.
- 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](#) on page 25.

- 3 Select File > New Scene to create a new scene.
- 4 To use all available screen space, maximize the Maya window.
- 5 From a panel menu bar, select Panels > Layouts > Single Pane.
- 6 Select the Rendering menu set. Unless otherwise noted, the directions in this chapter assume the Rendering menu set is selected.

Lesson 1: Painting in 2D using Paint Effects

Introduction

One of the features of the Paint Effects Tool is the ability to paint colors, patterns, and textures on a 2D canvas image. You can use the image as an independent piece of art, a background image in your scene, or a texture to be applied to a 3D surface.

In this lesson, you learn how to:

- Display the 2D canvas.
- Paint strokes on the canvas, and modify your paint brush.
- Use the Visor to access the many pre-defined paint brushes.
- Modify brush settings such as brush size, color, and transparency.
- Modify the canvas size.
- Smear, blur, and erase brush strokes from the 2D canvas.
- Use the pre-defined Paint Effects brushes to paint flowers and foliage.
- Save modified brush settings to the Shelf.
- View the alpha channel for your 2D image.

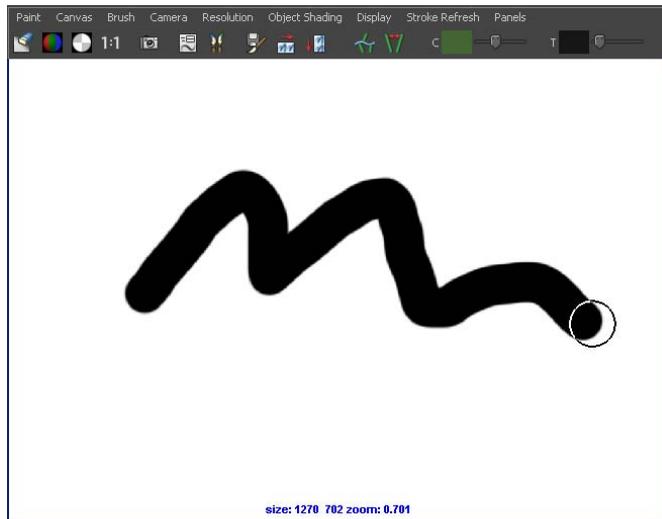
Painting strokes

In the following steps, you display the 2D canvas and use preset brushes to paint a few strokes.

Although not covered in this lesson, you can use a tablet and stylus to paint with pressure sensitivity. The harder you press down, the more paint is applied.

To paint strokes on the 2D canvas

- 1 Do the steps in [Preparing for the lessons](#) on page 562.
- 2 If you want to create maximum space in the Maya window, turn off Display > UI Elements > Channel Box/Layer Editor, Tool Settings, and Attribute Editor.
- 3 On the panel menu bar, select Panels > Panel > Paint Effects (hotkey 8). The panel displays a Paint Effects toolbar and menus.
- 4 From the Paint Effects menus, select Paint > Paint Canvas to display the 2D canvas.
You're ready to start painting. It is not possible to display a grid on the canvas.
- 5 Paint a stroke by dragging across the canvas.



When you first display the Paint Effects panel, your strokes use the *default brush*—a solid black stroke similar to an airbrush.

TIP You can dolly and track the canvas using Maya's standard mouse and keyboard conventions.

- 6 Now paint with various preset brushes. To select a preset brush, select Brush > Get Brush or click the following button in the toolbar:



The Visor window opens.

The Visor organizes the hundreds of Paint Effects brushes into category folders. In the Visor, click a folder, then click a brush icon in the area to the right of the folders. (The name of each preset brush has the extension .mel because each is a MEL script that establishes the brush's operation characteristics.) You can dolly within the Visor window to enlarge the viewable size of the brush icons and read the names more clearly.

You can copy a brush or all brushes within a folder to the Shelf. Use the middle mouse button to drag the brush or folder to the Shelf. Use the shelf editor (available from the arrow menu at the left of the shelf) to remove items from the shelf.



Some brushes apply paint directly along the paint path, while others produce added effects. For example, the pens folder includes brushes such as ballpointRed.mel that work like pens, while the flowers folder includes brushes such as dahliaPink.mel that grow flowers as you drag the mouse. Brushes like dahliaPink.mel use Paint Effects tubes.

- 1 Select the dahliaPink.mel brush and drag the mouse slowly for one stroke, then quickly for the next stroke.

When you paint a stroke slowly, the density of flowers is higher than when you paint quickly. The effect of stroke speed on density works the same for all brushes that use tubes.

- 2 To erase your last stroke, select Canvas > Canvas Undo.

When displaying the 2D canvas, you can undo only a single stroke. Edit > Undo, Redo, Repeat (on the main menu) doesn't undo strokes.

- 3 To erase the entire canvas, select Canvas > Clear or click the Canvas Clear

button on the toolbar: 

Modifying the default brush settings

When you select a preset brush, Maya copies its brush settings to a *template brush*. A template brush is a holder for the attribute settings of the next strokes you paint. When you change the settings of the template brush, they affect only the next strokes, not previous strokes.

To edit the template brush settings

- 1 Select Brush > Reset Template Brush to select the default brush, then paint on the canvas so you remember the brush effect.
- 2 Select Brush > Edit Template Brush or click the following button in the

toolbar: 

You can edit various brush settings, for example, the brush width, in the window that appears.

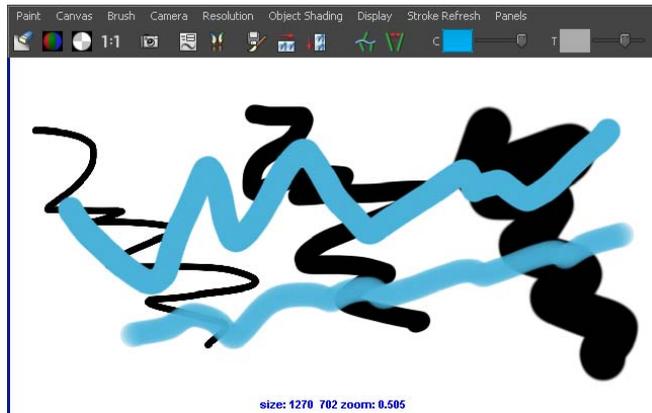
- 3 To resize the brush, change the Global Scale value. Alternatively, you can move the brush over the canvas, press and hold the b Hotkey and drag left or right. The circle on the canvas represents the size of your brush.
- 4 Paint on the canvas to try out a new size.



size: 489 286 zoom: 1.000

- 5 Close the Paint Effects Brush Settings window.
- 6 On the toolbar, click the swatch to the right of C (color) to display the Color Chooser.
- 7 Select a color from the Color Chooser and paint on the canvas.
- 8 On the toolbar, slide the T (transparency) slider to the right until the T box is a light gray. Paint on the canvas.

When you paint over existing paint strokes you will notice that your paint is transparent allowing the previous strokes to show through.



size: 1270 702 zoom: 0.505

NOTE The Toolbar color boxes, for example, C and T, are also available in the Paint Effects Brush Settings window. (Select Brush > Edit Template Brush > Shading.)

Modifying the canvas

In the next steps you change a few canvas options, including color and size.

To modify canvas settings

- 1 Clear the canvas (Canvas > Clear).
- 2 Select Canvas > Set Size and set the canvas size to 512 x 512—the width and height in pixels. When you are prompted to save, click No, then close the Set Canvas size window.

Although you'll often use the default maximum canvas size to take advantage of the available screen space, you might need to use a smaller or larger size in many circumstances. For instance, if you plan to use the image as a background in a computer game, you might need to use a small canvas size such as 256 by 256.

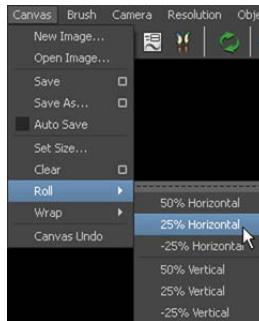
- 3 Turn on the wrap buttons on the toolbar:



- 4 Paint strokes past the edges of the canvas.

The strokes wrap around to the opposing side. This feature is useful for creating patterned texture images to be applied to surfaces. When you tile or repeat the image on the surface, the wrapped strokes help make the borders between the images unnoticeable.

If you want to display the area of the canvas where the edges join, you can do this using the roll feature (Canvas > Roll). You can select the amount and direction of the roll from the menu.



- 5 Select Canvas > Clear > . In the options window, slide the Clear Color slider all the way to the left. Click Apply.

This clears existing strokes and turns the canvas black. Before you paint strokes on the black canvas, you may notice that some brushes seem to have no effect. If this occurs, it's because the paint is black or because it is transparent enough to prevent the color from showing.

Modifying the colors of a preset brush

In the following steps, you alter leaf and bud colors of a preset brush; fern.mel.

TIP When using the fern.mel brush, if any of the sliders do not appear, or appear cut off on the right, you need to increase the screen display resolution setting on your monitor to 1280 X 1024.

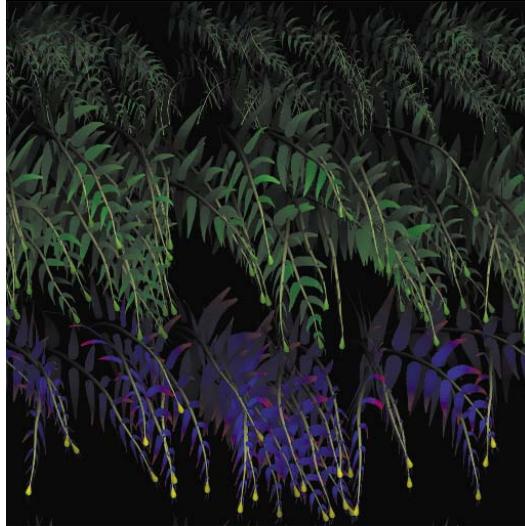
To change the color of a brush

- 1 In the Visor (Brush > Get Brush), click plants folder, and select the fern.mel preset brush.
- 2 Paint on the canvas to see the default display.
Because fern.mel uses tubes, ferns grow along the path of your stroke.
- 3 Resize the brush to a Global Scale of about 0.5 (Brush > Edit Template Brush) and paint on the canvas.
- 4 On the toolbar, click the L1 color box (Leaf Color 1), select a shade of blue from the Color Chooser.
Move your mouse off the Color Chooser to close it.
L1 sets the color of the tips of the leaves.
- 5 Click the L2 color box (Leaf Color 2) and select a shade of red from the Color Chooser.
L2 sets the color of the base of the leaves.



- 6 Click the B color box (Bud Color) and select a shade of yellow from the Color Chooser. B sets the color of the flower buds.
- 7 Paint on the canvas and observe how the changes you've made to the settings affect the final paint effect.

TIP You can change the path width of fern.mel without changing the width of the leaves and stems. Select Brush > Edit Template Brush to display the Brush Settings window, then in the Brush Profile section, change the value of Brush Width. Similarly, you can alter the path width for any other preset strokes with tubes.



- 8 Clear the canvas (Canvas > Clear).

Editing strokes with tubes attributes

Now you'll edit the attributes that specify the foliage that grows along the path with fern.mel.

To edit tube attributes

- 1 To edit brush settings, select Brush > Edit Template Brush.
- 2 Open the Tubes section, then open Growth.
By default, only Leaves and Buds are created as you paint. You can also create Branches, Twigs, and Flowers to simulate a fern in various stages of its natural life cycle.
- 3 Turn on Flowers. Paint on the canvas to see the result.

- 4 Open the Leaves section, change Leaves in Cluster to 4, then paint on the canvas.

Leaves are created in radial clusters around a branch or twig. This setting defines how many leaves are in each cluster. The higher the number, the denser the foliage.

- 5 To make the flowers point in a different direction, open the Behavior section and then the Forces section. Set Path Follow to 0.1.
- 6 Paint to see the result.



Saving brush settings for future use

When you first selected the preset brush `fern.mel` previously, Maya copied it to the template brush. You then edited the color and various other attributes of the template brush to create a unique ferns brush. The next steps show how to save the current settings of the template brush for future use.

To save a brush to the Shelf

- 1 On the main menu bar, select `Paint Effects > Save Brush Preset`. This is the menu item you use to save the template brush as a preset brush.
- 2 Type the name `blueFern` in the Label box as well as the Overlay Label box.

This name appears in icons on the Shelf or Visor after you save the brush.

- 3 For the Save Preset option, turn on To Shelf. This means you'll put the brush on the Shelf but not in the Visor.
- 4 Click Grab Icon and drag a selection box around a part of the image that is representative of the brush's result.
A swatch appears in the window for the selected region. Repeat this step until you are satisfied with the icon.
- 5 Click Save Brush Preset. An icon for the new preset brush appears on the Shelf.
- 6 To check that the blueFern brush works, select a different preset brush and paint on the canvas, then select the blueFern from the Shelf and paint on the canvas.
- 7 Clear the canvas and close the Save Brush Preset window.

Blending brushes

You can blend brushes. Before beginning, display the Visor, click gold.mel in the metal folder, and paint on the canvas so you know gold's effect. Do the same for grassOrnament.mel from the grasses folder.

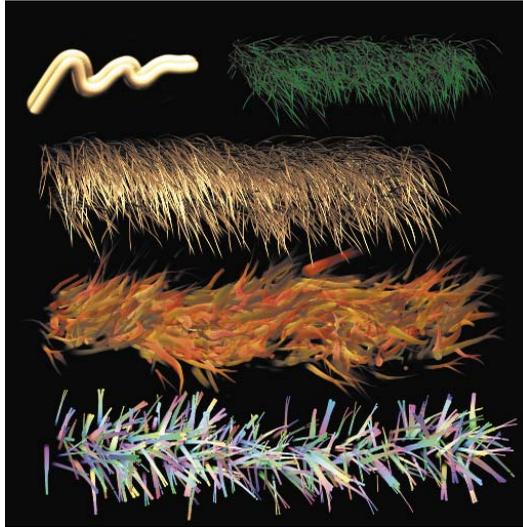
To blend brushes

- 1 Click grassOrnament.mel in the grasses folder.
- 2 From the menu, select Paint Effects > Preset Blending > .
The Preset Blending options window appears.
The brush preset blending is toggled on and the options in this window set how much the template brush is influenced by the next preset brush you select. This affects only the subsequent painting, not the preset brushes in the Visor.
- 3 Change the Shading value to 80 and the Shape value to 0.
When you select the next brush (gold.mel), Maya uses 80% of its shading and 0% of its shape. Conversely, the stroke uses 20% of the grassOrnament.mel brush's shading and 100% of its shape.
- 4 To confirm the effect, select gold.mel (in the metal folder), then paint on the canvas.

- 5 Repeat the preceding steps, experimenting with different preset brushes, Shading, and Shape values. As a first try, blend flameCoarse.mel from the fire folder with the existing template brush. Set Shading to 60 and Shape to 30.

If the Brush Preset Blend window is open, each time you select a preset brush, its Shading and Shape values will be blended with the template brush.

- 6 Close the Brush Preset Blend window when you are done blending brushes.
- 7 Do not clear the canvas. You'll use your existing strokes in the next steps.



Smearing, blurring, and erasing paint

Now you'll work with brush types that smear, blur, and erase paint.

To smear, blur, and erase paint

- 1 Select Brush > Reset Template Brush to select the default brush.
- 2 On the Brush menu, try the Smear, Blur, and Erase menu items and paint on the strokes already on the canvas. You can also change these settings using the Brush Type setting in the Paint Effects Brush Settings window (Brush > Edit Template Brush).

- 3 If you need to perform compositing work, you can view the image's alpha

channel by clicking this button: .

White regions represent full opacity. Black represents full transparency.



- 4 If you turned on the alpha channel (Display > Alpha Channel), return to the display of full color by selecting Display > All Channels.
- 5 Clear the canvas and restore the canvas color to white (Canvas > Clear > ).
- 6 To quit the canvas display and return to a scene view, select Panels > Perspective > persp (Hotkey 8).
- 7 In the Toolbox, click the Select Tool.
Clicking the Select Tool quits the Paint Effects Tool, and the mouse cursor displays as an arrow.
- 8 Select Paint Effects > Preset Blending to turn off Preset Blending before you proceed to the next lesson.

Beyond the lesson

In this lesson, you learned how to:

- Use Paint Effects to paint on the 2D canvas.
- Use the preset Paint Effects brushes from the Visor.
- Edit, blend, and save preset brushes for later use.

Painting on the 2D canvas is useful when you need to create a texture to be applied to a surface. For example, suppose you paint a repeating brick pattern on the canvas and save the canvas as an image file. You can create a material, for instance, a Blinn material, then apply the image file as a file texture to the color of the Blinn material. You can then apply the Blinn material to a 3D wall you've modeled, which creates the illusion of a brick wall.

Painting a 2D canvas is also useful for creating a background image for a scene. For instance, you might create lush foliage on the canvas, put it on a camera's image plane, then use it as the backdrop for animated jungle animals in the foreground.

Although not covered in this lesson, you can use a tablet and stylus to paint with pressure sensitivity. Pressure sensitivity allows you to more realistically simulate the effects of traditional drawing media (pens, pencils, markers, etc.)

Lesson 2: Painting in 3D using Paint Effects

Introduction



Image created by: Duncan Brinsmead

The Paint Effects Tool goes beyond traditional paint software by allowing you to paint directly in 3D space, either on the ground plane or on an object's surface. When you use Paint Effects for 3D work, the paint strokes are three-dimensional and editable.

With Paint Effects, all of the hundreds of preset brushes found in the Visor under the Paint Effects tab (airbrushes, pens, markers, watercolors, plants, fire, metal, flowers, trees, and underwater etc.) can be used to create 3D objects in your scene. For example, you can use the preset plant brushes to paint a 3D tropical fern garden around a walking character in your scene.

You can animate brush strokes to create dynamic effects. For example, you can animate the ferns to shake in the wind or have them move when a character walks near them.

An electronic tablet with stylus is an ideal tool for painting with Paint Effects. With a real paint brush, pressing harder makes the brush wider and applies the paint more thickly. You can achieve the same effect with Paint Effects by specifying the effect of pressure on the stylus for each stroke and the threshold of response to pressure. You can map up to three attribute values to stylus pressure and edit them as you work.

When you use Paint Effects to paint 3D strokes, you can edit the brush and stroke attributes to create a variety of effects.

In this lesson, you begin to explore what is possible with painting in 3D using the Paint Effects Tool by learning how to:

- Paint directly in the 3D scene view.
- Choose brushes from the Visor and the Shelf and modify their brush attributes.
- Paint in 3D using the Paint Effects panel.
- Differentiate between paint brushes and paint strokes.
- Make surfaces paintable so paint strokes can be applied to them.
- Select brush strokes and modify their attributes.
- View strokes in wireframe and rendered mode.
- Apply various effects to stroke attributes.

Preparing for the lessons

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

- 1 If you have not done so already, consider doing the lessons in the Rendering section of this guide. Familiarity with Maya rendering concepts is important to understanding the lessons.
- 2 Select File > New Scene to create a new scene.
If you just finished Lesson 1 in this chapter you may be presented with a window asking if you wish to Save previous paint effects. Select No.
- 3 To use all available screen space, maximize the Maya window.
- 4 From a panel menu bar, select Panels > Layouts > Single Pane.
- 5 Select the Rendering menu set. Unless otherwise noted, the directions in this chapter assume the Rendering menu set is selected.
- 6 From the Shelf, select the Paint Effects tab to display the various preset brushes.

If the Paint Effects Shelf tab does not appear as part of the shelf set, see the Maya Help for Shelves. (For Linux users, any new shelves do not load by default.)

Regardless of your operating system, the Paint Effects preset brushes are accessible from the Visor panel. (From a view panel, select Panels > Panel > Visor.)

Brushes and strokes

Like a traditional paint program, Paint Effects allows you to use a brush and paint strokes. In Paint Effects, the brush and stroke have unique qualities and characteristics.

A *brush* contains the various attributes that define the appearance and behavior of the paint. The *template brush settings* define the initial settings for a brush.

When you drag the Paint Effects brush, you create a path for the brush called the paint *stroke*.

A stroke is a 3D curve (or collection of curves) with attributes that define how the paint is applied along the stroke path. You can paint strokes directly in the 3D scene or on objects.

When you select a brush preset from the shelf or from the Visor, that brush is copied to the template brush. Any changes you make to the template brush settings affect what the next stroke looks like, but they do not affect any previously painted strokes.

To select a Paint Effects preset brush from the Shelf

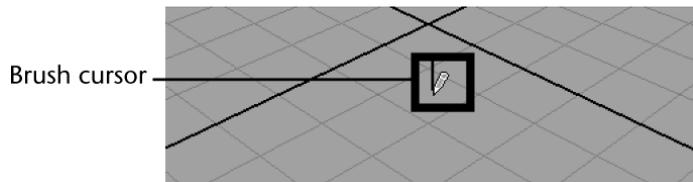
- 1 From the Toolbox panel layout shortcuts, select the Single Perspective View.
- 2 From the Shelf, select the Paint Effects tab to display the Paint Effects preset brushes.



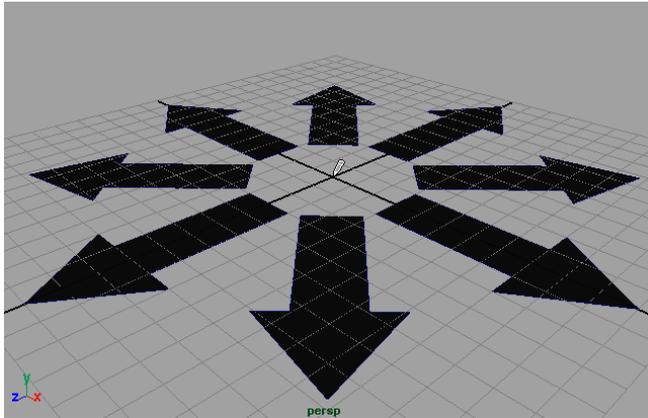
- 3 From the Paint Effects shelf, select the Daisy Large Brush.

Daisy Large brush

In the scene view, the cursor changes to a red circle with a vertical line indicating it is set to paint a stroke. The circle indicates the width of the brush path. With some types of strokes, this path will appear as the width of the actual paint; with other stroke types, it indicates the width of the tube seeding path, for example, plants, trees, etc.



The settings of the preset brush you select are copied to the template brush. The template brush settings are used when a paint stroke is drawn. If you want to modify the brush presets before you begin to paint, you must edit the template brush after you've chosen the preset brush (Paint Effects > Template Brush Settings).

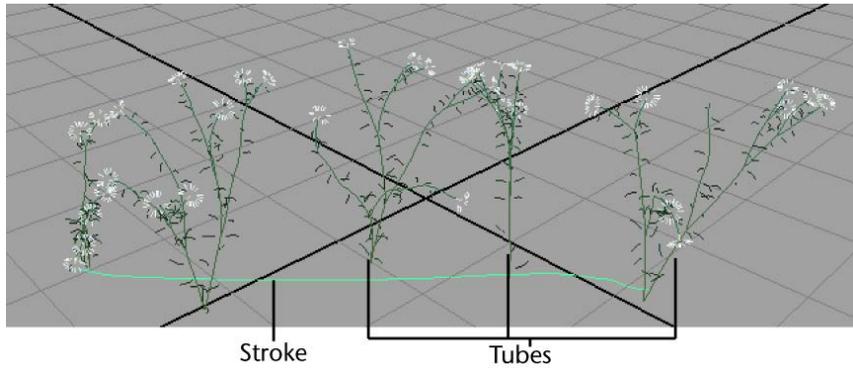


By default, the brush cursor moves along the ground plane. The ground plane lies in the X, Z dimension. This is the default behavior for the brush cursor. If you draw a paint stroke, it is applied to the ground plane (X, Z).

To paint 3D strokes in the scene view

- 1 With the Daisy Brush selected, draw one short paint stroke on the ground plane using one of the following methods:
 - Drag the stylus across the tablet while exerting slight pressure on the stylus as you move it.
 - Drag the mouse while pressing the left mouse button.

One or more wireframe daisy stalks appear in the scene view along the path of the paint stroke. When you paint a stroke in the scene view it initially appears as a wireframe. Paint is not applied to the stroke until it is rendered.

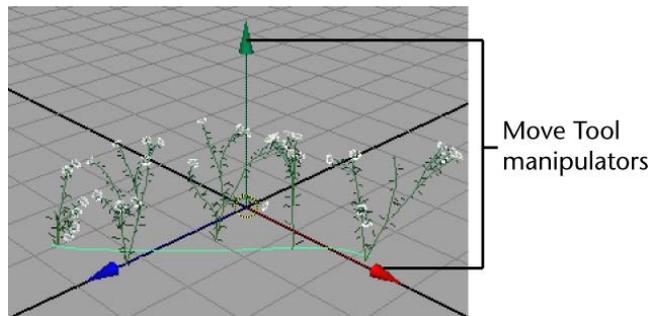


The stalks that extend out of the curve or stroke are called *tubes*. In Paint Effects, strokes are either simple strokes or strokes with tubes.

Because the stroke has a curve associated with it, you can move, scale, or rotate the stroke like other objects in the scene. You can edit the curve to modify the shape of the stroke path.

- 2 Dolly and tumble the scene to better view these strokes with tubes in 3D.
- 3 Select the Move Tool from the Toolbox.

A move manipulator icon appears at the X, Y, Z origin. (If the move manipulator does not appear, you can click on the stroke to reselect it with the Move Tool.)



- 4 In the scene view, drag any of the Move Tool manipulator arrows to reposition the stroke.

Each time you paint a stroke, Paint Effects creates a new brush and stroke node and attaches them to the stroke that is created. You can modify these brush and stroke nodes once they've been created. To edit these brush or stroke attributes, you use the Attribute Editor.

To modify the attributes of an existing paint stroke

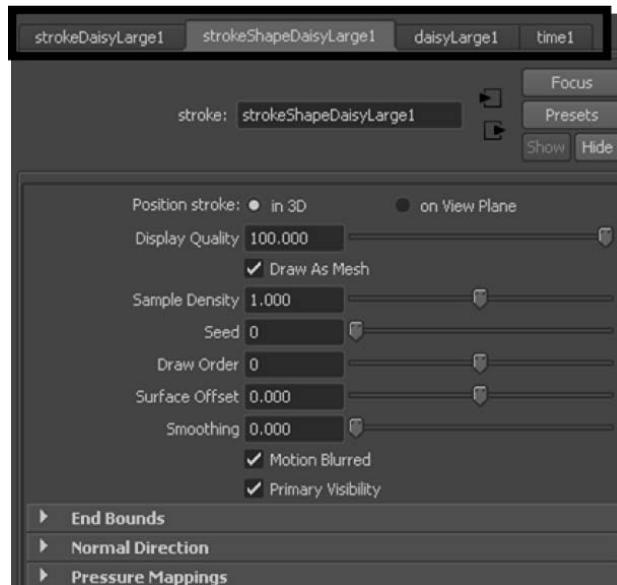
- 1 To view the Attribute Editor, click the Show/Hide icon on the Status Line.



The Attribute Editor displays.

- 2 In the scene view, select the paint stroke associated with the daisies. You can select a stroke by one of the following methods:
 - Select the curve associated with the stroke path.
 - Select the tubes associated with the stroke path.

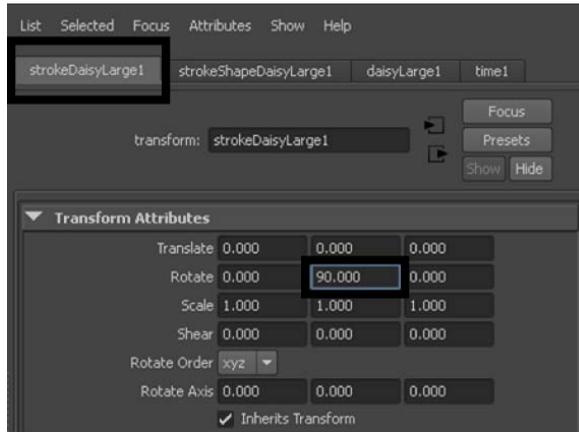
The Attribute Editor updates to display the nodes associated with the selected daisy stroke. The various attributes for the stroke appear under different tabs. Each tab represents a node associated with a specific set of attributes.



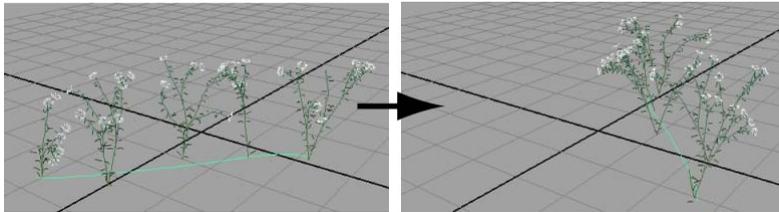
- 3 Click on the strokeDaisyLarge1 tab to see its attributes.

This tab contains information related to the transform node, because the most important attributes on this tab control the stroke's curve transformation.

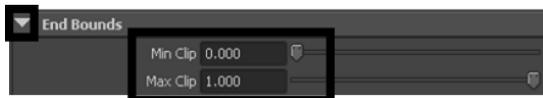
- 4 Under Transform Attributes, set Rotate to be 0, 90, 0.



The stroke is rotated 90 degrees about the Y axis.



- 5 Click the strokeShapeDaisyLarge1 tab to see its attributes.
This tab is called the stroke attribute node because the attributes establish the paint stroke's properties when the stroke is first created.
- 6 Expand the End Bounds section by clicking the arrow to see the Min and Max Clip attributes.

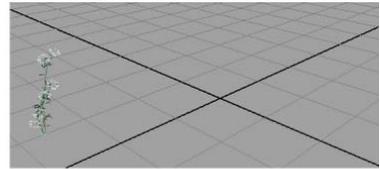
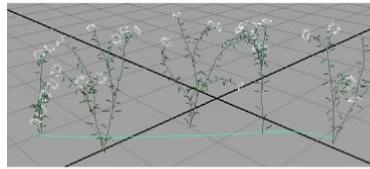


Min Clip specifies the start of the stroke along the path. Max Clip specifies the end of the stroke along the path.

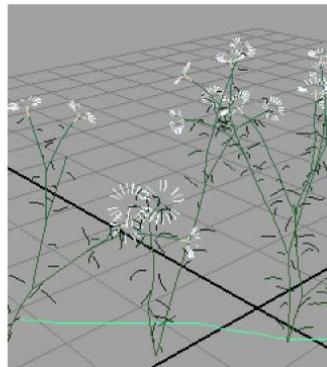
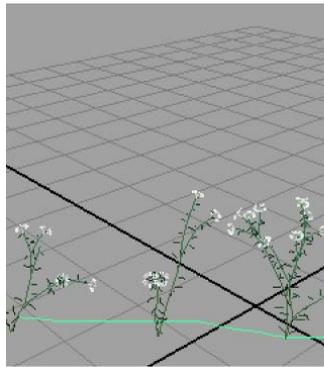
- 7 Drag the Max Clip slider from its present value of 1.0 to a value of 0.1.

The stroke path is shortened.

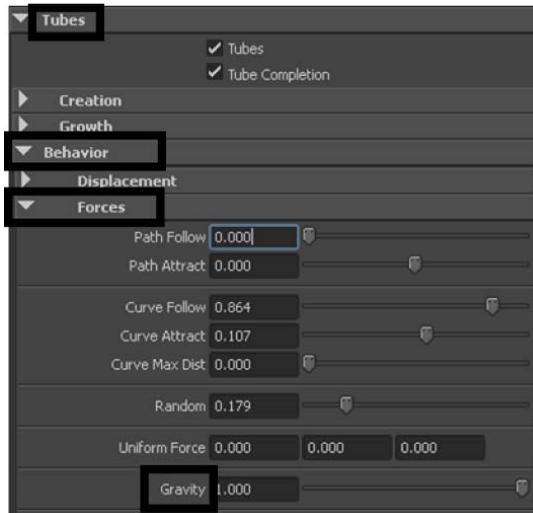
Min and Max Clip can be animated to create the effect of the paint advancing along a path.



- 8 Drag the Max Clip slider back to its original value of 1.0.
- 9 Click the daisyLarge1 attribute tab to see its attributes.
This is the node that contains attributes related to the paint brush associated with the stroke.
- 10 Set the Global Scale attribute to 6.0.
The daisies scale larger along the stroke's path.

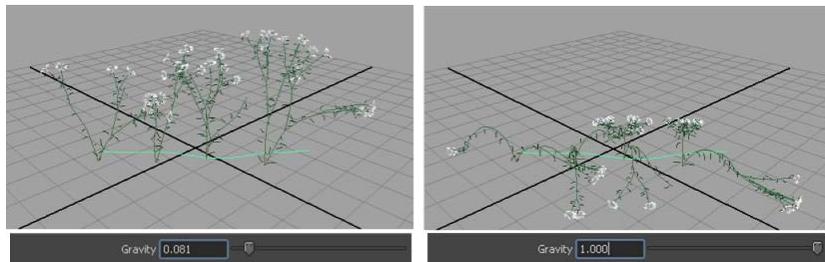


- 11 In the daisyLarge1 node, expand the Tubes attributes, then Behavior and Forces.



12 Drag the Gravity slider to the right so that the value of gravity increases to a value of 1.0.

The daisies lay on the ground plane, as if wilting, with the increased gravity.



Drag the Gravity slider back to its original position approximately so that the stalks appear upright.

The last tab in the Attribute Editor is the time node. The time node is used for animation and dynamics. You do not use this node in this lesson.

Rendering Paint Effects strokes

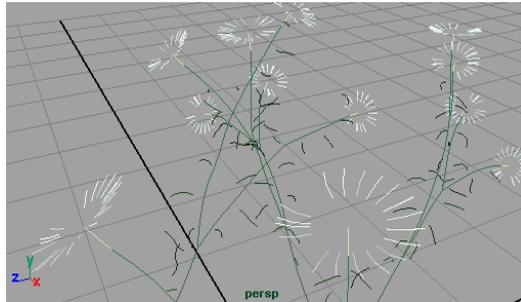
The paint strokes display in wireframe mode when you paint in the scene view. Working in wireframe mode is very efficient in terms of interactive

performance but you don't see how the paint actually appears. To view the paint strokes in a rendered mode you can either:

- Render the scene view using the software renderer.
- Switch to painting in the scene painting view using the Paint Effects panel.

To render the paint strokes in the scene view

- 1 Dolly or tumble the scene so the view of the daisies appears roughly similar to the image below.



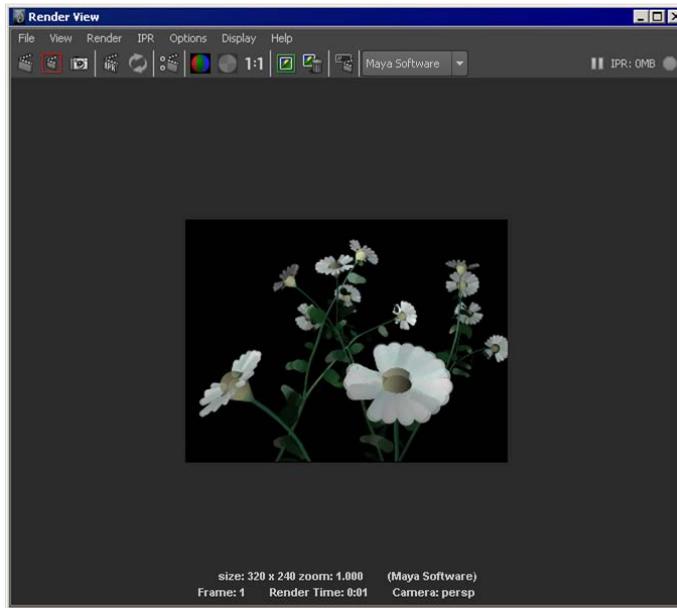
- 2 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon on the Status Line) to display the Render Settings window.



- 3 In the Render Settings window, do the following:
 - Select Maya Software from the Render Using drop-down list.
 - In the Image Size section, select 320×240 from the Presets drop-down list.
- 4 From the Status Line, click the Render Current Frame button.



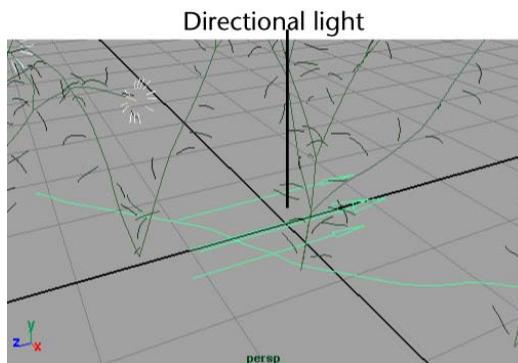
The Render View window opens, renders the image, and then displays it.



The daisies may initially appear a bit dark in the image.

The Maya renderer creates a temporary light with a default brightness level in order to render the image when no lights are present in the scene. This ensures that you don't produce an image that is totally black; however it may not produce the correct illumination for your scene. If you want to obtain more control over the illumination of your scene, you can create a light for the scene and adjust its brightness.

- 5 From the main menu, select Create > Lights > Directional Light. A directional light is created at the X,Y,Z origin of the scene.

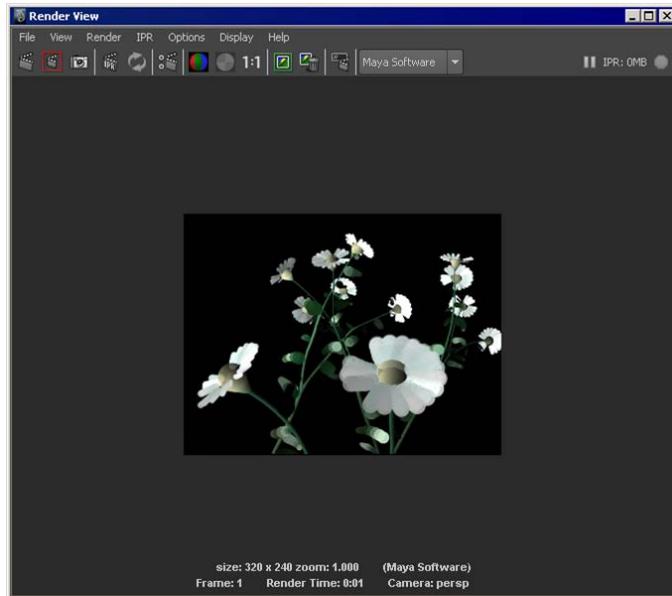


The Attribute Editor updates to display the attributes for the active directional light.

- 6 In the Attribute Editor, set the Intensity for the directional light to be 1.5.
- 7 From the Status Line, click the Render Current Frame button to render the scene again.

The image renders and the Render View window image is refreshed.

The daisies appear brighter in this version of the rendered image.



- 8 Close the Render View window.

Sometimes you may want to view your painted strokes as you paint. To do this, you can paint in the Paint Effects panel view and use the scene painting view mode.

NOTE You can also render Paint Effects using the mental ray for Maya renderer, but you must first convert your Paint Effects strokes to polygonal meshes by selecting **Modify > Convert > Paint Effects to Polygons**. See **Convert Paint Effects to polygons** for more information.

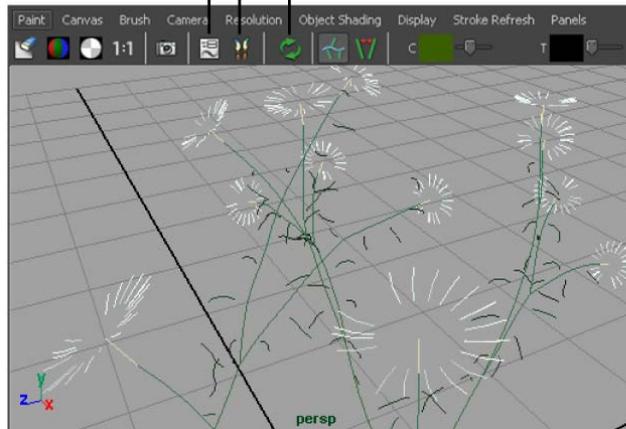
To display rendered paint strokes as you paint

- 1 Press 8 to switch to the Paint Effects panel.

NOTE If the 2D Paint Canvas appears instead (i.e.: panel appears white), select Paint > Paint Scene, from the panel menu.

The Paint Effects panel menu and toolbar displays as well as the scene painting view. This panel allows you to paint in scene painting view mode.

Edit Template Brush Get Brush (Visor) Redraw Paint Effects View



8

The scene painting view is a snapshot of the modeling view. In this view you can both paint and view your strokes in a rendered fashion to see what they will really look like as you work.

You can dolly, track, and tumble in the scene painting view. You can select objects by pressing the Ctrl key and clicking on them. If you want to move, or otherwise manipulate 3D objects in the scene view, you must exit the Paint Effects panel by pressing the 8 key again.

- 2 With the Daisy Brush selected, paint a stroke in the scene painting view. The new stroke displays in rendered mode. If you dolly or track the scene painting view, any rendered strokes will revert back to wireframe display until you refresh the rendered display.
- 3 To automatically update the paint strokes so they display in rendered mode, select Stroke Refresh > Rendered.

It will take a few moments for the rendering of all of the paint strokes to be calculated and then displayed.

Each time you adjust the camera view while working in scene painting mode with Refresh > Rendered turned on, the system will automatically update the rendering based on the new view.

Automatically refreshing the rendering can be time-consuming depending on the complexity of paint strokes in your scene. Alternatively, you can set the rendering to refresh only when you manually choose to do so.

- 4 From the Paint Effects panel menu, select Stroke Refresh > Off.
Now you must manually refresh the Paint Effects rendered view using the Redraw Paint Effects view button in order to see the paint strokes rendered.



- 5 Dolly the view to obtain a closer view of the daisies.
- 6 Manually refresh the rendering of the daisies by clicking the Redraw Paint Effects view button.



- 7 Exit the scene painting view mode by pressing the 8 key.

Paint Effects on 3D objects



Image created by: Mark Jamieson

With Paint Effects, you can paint 3D strokes directly on 3D objects in a Maya scene.

In this section, you create a surface for an underwater terrain and sculpt it to the desired form using the Sculpt Geometry tool. You use the underwater preset Paint Effects brushes found in the Visor to begin the underwater scene.

Creating a surface to paint on

In Paint Effects, NURBS surfaces are the only surface type you can paint on. You will create a NURBS plane, and sculpt it to make its surface appear like an underwater terrain.

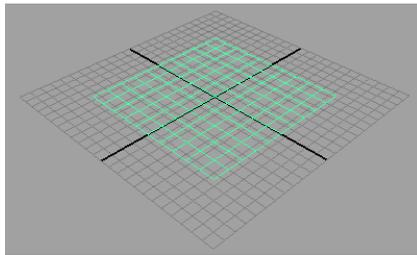


To create a surface terrain for the underwater scene

- 1 Select File > New Scene.
A new empty scene is created.
- 2 From the main menu, select Create > NURBS Primitives > Plane > .
The NURBS plane options window appears.

- 3 In the NURBS Plane Options window, select Edit > Reset settings and then set the following options:
 - Width: 15
 - Length: 15
 - U Patches: 10
 - V Patches: 10
- 4 In the NURBS Plane Options window, click Create.

Maya creates a plane primitive and positions it at the origin. The plane is 15 units by 15 units in dimension with 10 subdivisions along each side.



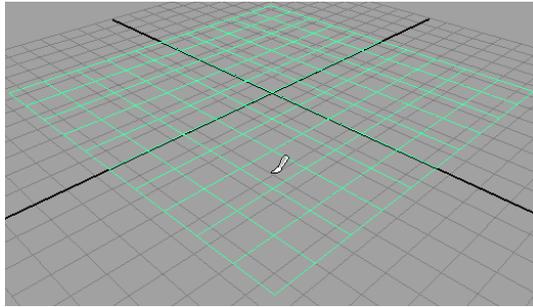
This surface will become the seabed terrain for your scene.

To sculpt a terrain using the Sculpt Geometry Tool

- 1 Select the plane surface.
- 2 Turn off the Interactive Creation option for primitives: Select Create > NURBS Primitives > Interactive Creation (no check mark).
- 3 On the Status line, select Surfaces from the drop-down menu.

The Main Menu changes to display Surfaces menu set.
- 4 From the Main menu, select Edit NURBS > Sculpt Geometry Tool.

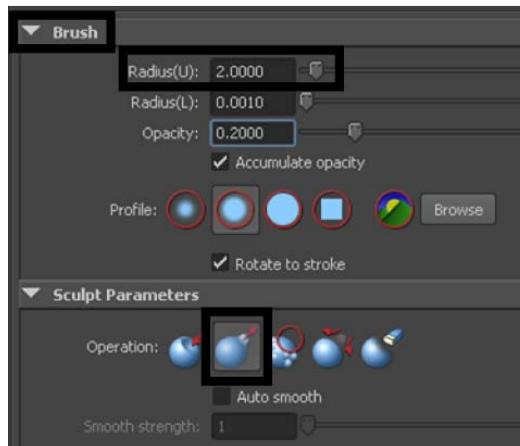
The cursor display changes to the brush cursor for the Sculpt Geometry tool.



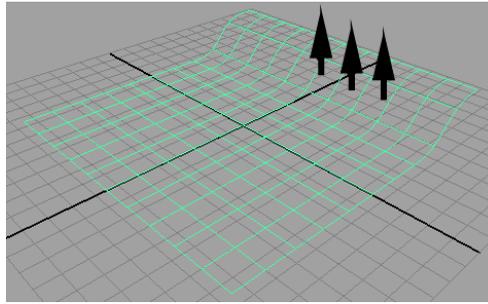
- 5 On the Status Line, click the Show/Hide Tool Settings button to display the tool settings for the Sculpt Geometry tool.



- 6 In the Brush section, set the Radius (U) to 2. and set the sculpt Operation to Pull.

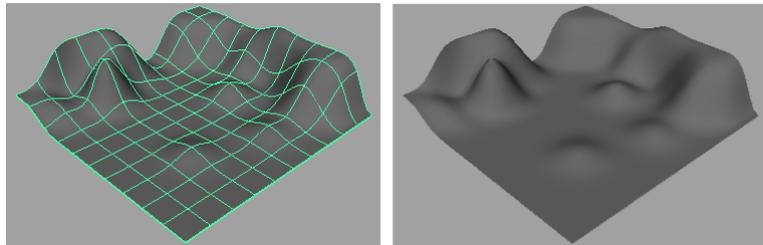


- 7 Drag the cursor along the back right edge of the plane to pull it upwards as shown.



- 8 Drag the cursor along the back left edge of the plane to pull it upwards.
- 9 Dolly and track the view as required so you can easily view the areas you need to sculpt.
- 10 Sculpt along the front and middle of the plane to make some bumpy areas.
- 11 Repeat steps 7 through 10 until you have your 3D terrain looking roughly as shown in the image below.

TIP If you want to undo a particular sculpt operation, click Ctrl + z to undo the sculpt stroke. You can click Ctrl + z repeatedly to undo a number of sculpt operations.

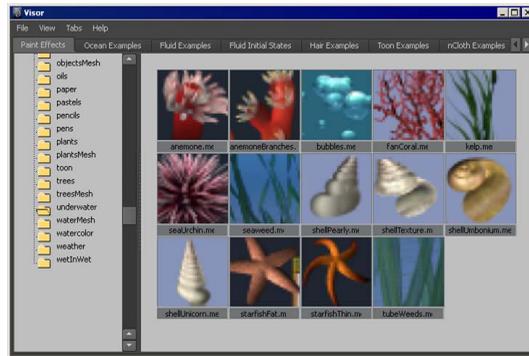


- 12 When you complete your terrain, return to the Rendering Menu set by selecting Rendering from the drop-down menu on the Status Line.
- 13 Hide the Sculpt Geometry Tool settings window before proceeding.

To select preset brushes from the Visor

- 1 From the main menu, select Paint Effects > Get Brush.
The Visor panel appears.

- 2 In the Visor, click the Paint Effects tab to display the various brush folders.
- 3 Scroll down the brush folders list to the underwater folder, and click it to view its contents.



In the next section of the lesson, you use these brushes to create an underwater scene.

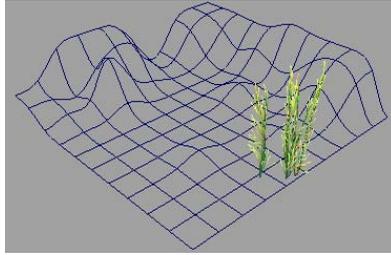
Painting on objects

To paint on a 3D object, you must first make the object paintable. When you paint directly on a paintable object, Paint Effects creates a curve on the surface along the stroke path and attaches the stroke and new brush to it.

If you make an object paintable and then try to paint on one that is not paintable, no stroke occurs. If you don't make any objects in your scene paintable, your strokes are placed on the ground plane.

To paint on a 3D object

- 1 With the terrain surface selected, select Paint Effects > Make Paintable.
- 2 From the Visor, select the brush named kelp.mel.
- 3 Paint a short stroke onto your 3D terrain surface.
When you paint directly on a paintable object, Paint Effects creates a curve on the surface along the stroke path and attaches the stroke and new brush to it. If you move the plane, the paint stroke moves with it.
- 4 Paint a few more kelp strands throughout the scene as shown.



The kelp brush orients the tubes to point along the *normal* of the surface you paint on. In the ocean, kelp grows towards the surface of the water in a vertical manner. If you want your kelp to appear realistic, it should point vertically upwards. If you paint the kelp on an angled part of the surface, the kelp tube will not point in a vertical direction.

TIP If you want to change the scale of the Paint Effects stroke to obtain larger plants or animals, press the b key and drag your stylus or mouse to the left or right to make the global scale for the current brush smaller or larger.

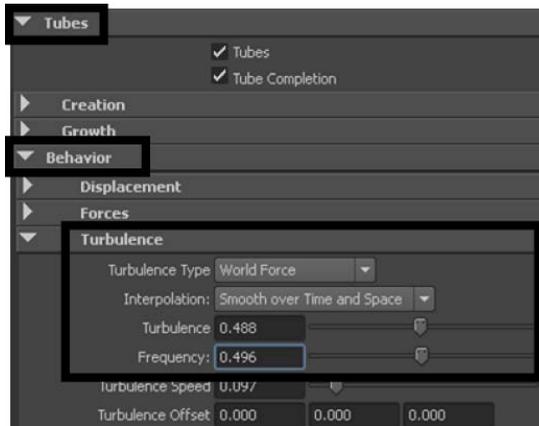
If you want to change the width of the brush stroke, to paint a narrower or wider tube path, for example, a wider swath of kelp plants, press the shift and b key and drag to the left or right.

Using turbulence with brush stroke tubes

In the ocean, kelp would react to the turbulence created by ocean currents. The tube attributes associated with many preset brushes have turbulence attributes turned on for interesting visual effects. Turbulence attributes can be increased from their default settings to make the kelp sway back and forth, providing additional realism when used in an animation.

To increase the turbulence on the kelp brush tube

- 1 In the scene view, select the kelp stroke.
- 2 Open the Attribute Editor, select the Kelp tab, and expand the Tubes > Behavior > Turbulence section.



- 3 In the Turbulence section, set the options as follows:
 - Turbulence Type: World Force
 - Interpolation: Smooth over Time and Space
 - Turbulence: 0.5
 - Frequency: 0.5
- 4 Click the Play Forward button on the Animation Time Slider to see the turbulence effect on the kelp (You don't need to set specific keyframes). The kelp animates as if it were affected by ocean currents.
- 5 Press the Stop button to cancel the animation playback.

You can adjust the turbulence settings while viewing the playback. If the frame range is too short, you can modify it to suit your needs.

Using additional preset brushes

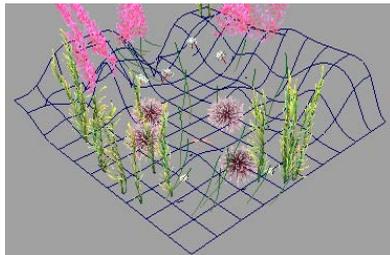
- 1 From the Visor, select the brush named fanCoral.mel.
- 2 Paint a few fan coral strokes in your scene near the back edges of the plane.

TIP If you paint a stroke in your scene and want to delete it you can do so by one of the following methods:

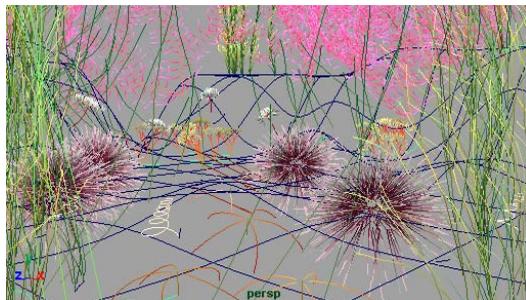
- Press Ctrl + z to immediately undo the stroke.
- Click on the stroke to select it and then press the Delete key.
- Marquee-select a tube associated with the stroke and then press the Delete key.
- Marquee-select multiple tubes and strokes and then press the Delete key.

If you want to delete all the paint strokes in your scene you can select Edit > Delete All by Type.

- 3 Paint additional strokes in your underwater scene using a few of the following brushes found in the underwater folder in the Visor:
 - sea urchins, starfish, seashells, and anemones



- 4 Dolly and track the scene for a closer view so it appears approximately as shown.



- 5 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon on the Status Line) to display the Render Settings window.



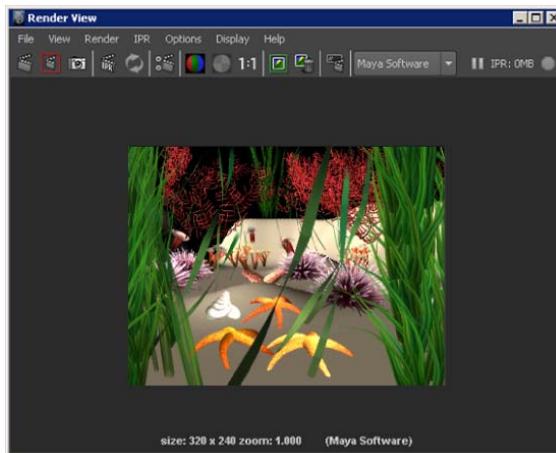
- 6 In the Render Settings window, do the following:
 - Select Maya Software from the Render Using drop-down list.

NOTE When rendering Paint Effects strokes, use the Maya Software renderer. Paint Effects strokes are not supported by the mental ray for Maya renderer.

- In the Image Size section, select 320×240 from the Presets drop-down list.
- 7 From the Status Line, click the Render Current Frame button.



- 8 The renderer executes, renders the image, and then displays it in the Render View window.



NOTE You can also render Paint Effects using the mental ray for Maya renderer, but you must first convert your Paint Effects strokes to polygonal meshes by selecting **Modify > Convert > Paint Effects to Polygons**. See **Convert Paint Effects to polygons** for more information.

You can create a seabed texture for the plane primitive, a few rocks, and some animated fish swimming to complete your scene.

You can spend additional time to further refine this scene, but this example gives you an idea of how a scene with various 3D props can be quickly established using Paint Effects and the various preset brushes found in the Visor.

Mesh brushes



Image produced using Tack Brush

Mesh brushes are a brush type that provide special features not available with other Paint Effect brush types. Mesh brushes render using polygonal triangles rather than brush stamps. As a result, mesh strokes can be more accurately texture-mapped for realistic results. They also appear more realistic compared to other brush types whether they are viewed close-up or from a distance.

The mesh brush stroke can also be converted to a polygonal surface in the Maya scene. Once the stroke is converted to polygons, the resulting polygons can be modified using any of the Maya polygonal modification tools and utilized like any other polygonal surface (animation, rendering, and so on)

In this section, you are introduced to mesh brushes and learn additional techniques related to the various brush and stroke attributes. You learn how to:

- Understand the structure of strokes with tubes.
- Modify a variety brush tube attributes.

- Convert the mesh brush stroke to polygons.
- Modify the converted mesh using construction history.
- Render the polygonal meshes.

To select a mesh brush preset from the Shelf

- 1 Select File > New Scene.
A new empty scene is created.
- 2 From the Toolbox panel layout shortcuts, select the Single Perspective View.
- 3 From the Shelf, select the Paint Effects tab to display the Paint Effects preset brushes.



- 4 From the Paint Effects shelf, select the Teapot Brush.

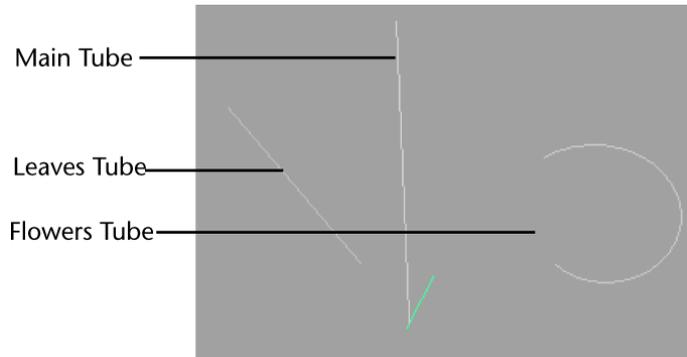
Teapot Brush icon



In the scene view, the cursor changes to a sphere icon indicating it is set to paint a stroke. The Teapot Brush is a mesh brush type.

- 5 In the scene view, drag your mouse or stylus along the ground plane to create one short stroke.
- 6 Dolly and tumble the view to see the wireframe version of the stroke.
The teapot stroke displays three wireframe tubes to represent the teapot. These tubes represent the body, spout, and handle for the teapot.

Paint Effects uses a plant or tree archetype to represent the various tube attributes. If you envision how a plant or tree has a trunk, out from which branches, leaves, flowers, and buds grow, you understand how each attribute is controlled in the attribute editor.



The center vertical line represents the main tube, whose attributes are found in the Creation tab of the attribute editor. The diagonal and semicircular lines represent the Leaves and Flowers tube attributes for the spout and handle of the teapot.

To render the paint effects stroke

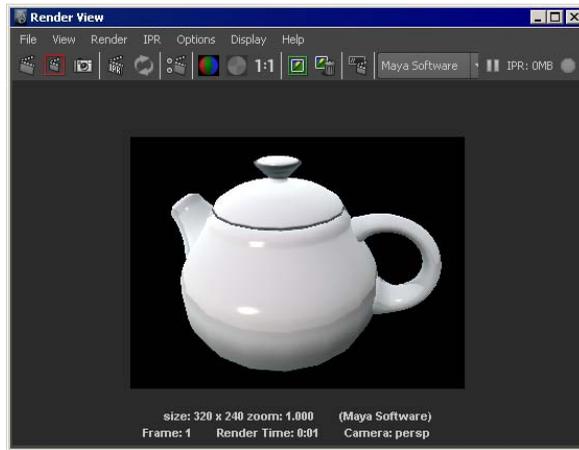
- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon on the Status Line) to display the Render Settings window.



- 2 In the Render Settings window, do the following:
 - Select Maya Software from the Render Using drop-down list.
 - In the Image Size section, select 320×240 from the Presets drop-down list.
- 3 From the Status Line, click the Render Current Frame button.



The renderer executes, renders the paint stroke, and then displays the image in a Render View window.



- 4 Close the Render View window.

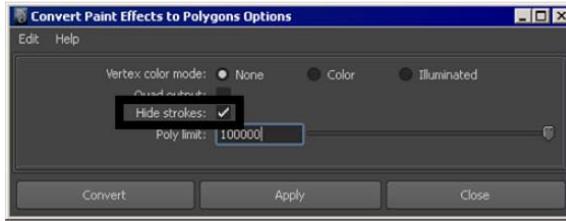
Mesh brush types are useful for representing hard surfaces compared to other brush types (the teapot is one example).

Converting mesh strokes to polygons

Mesh strokes can be converted to polygons. Converting the mesh stroke to polygons allows you to visualize any brush modifications you make in the Attribute Editor without having to render the stroke. It is possible to edit, animate, and render the converted polygon mesh the same way you would any other polygonal surface types.

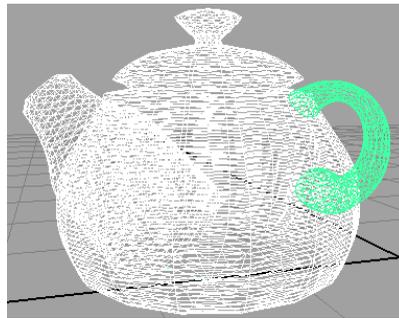
To convert a mesh stroke to polygons

- 1 Press the q key (Hotkey) to choose the Select Tool.
The cursor changes to an arrow icon indicating the change to selection mode.
- 2 Select the teapot stroke, if it is not already selected.
- 3 Select Modify > Convert > Paint Effects to Polygons >  .
- 4 In the Convert Paint Effects to Polygons Options window, ensure that the Hide Strokes option is set to On.



Setting the Hide Strokes option to On ensures that after the conversion, only the converted polygon mesh will be displayed.

- 5 In the Convert Paint Effects to Polygons Options window, click Convert. The mesh brush stroke is converted to a polygonal surface. A teapot model made of polygons results.



The original mesh brush stroke still remains but is now hidden.

- 6 Click anywhere in the scene view away from the teapot surface to de-select the teapot surface.
- 7 In the scene view, press 5 to smooth shade the display of the polygonal teapot.



Modifying a converted polygonal mesh

By default the original brush stroke is linked to the new polygonal surface via construction history. If you modify the original brush stroke attributes, the polygonal surface will update so long as the construction history link exists. This construction history link allows you to easily modify the polygonal mesh surfaces in a variety of ways. You modify the brush stroke attributes using the attribute editor.

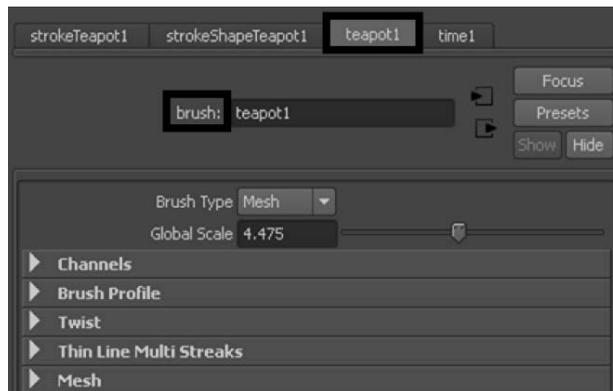
To modify the teapot brush attributes

- 1 Marquee-select the polygonal teapot.
- 2 To view the Attribute Editor, click the Show/Hide button on the Status Line.



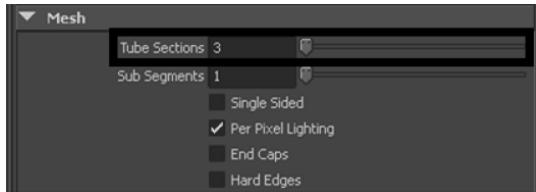
The Attribute Editor displays.

- 3 In the Attribute Editor, click on the teapot1 tab to display the brush attributes for the teapot stroke. (You may need to click the display arrow (Windows, Mac OS X) or select the tab from the pop-up list (Linux) in order to see the tab.)

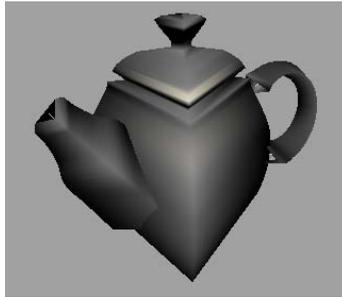


- 4 In the Attribute Editor, expand the Mesh attributes section to display its attributes.

- 5 In the Mesh attributes, drag the Tubes Sections slider to the left to a value of 3.

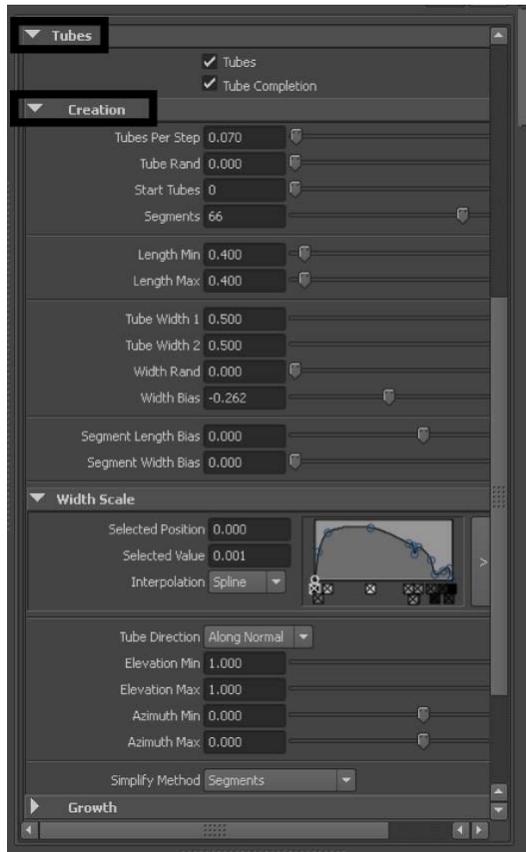


The body, spout, and handle of the teapot update to display as 3 sided surfaces.



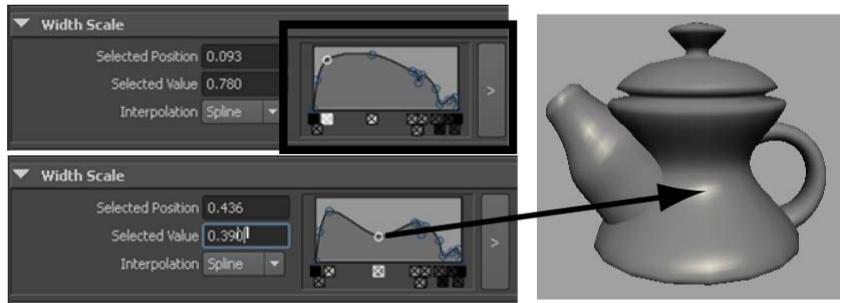
Tube Sections defines the number of sides around the width of the tubes in a mesh stroke. Changing the Tubes Sections gives a range of results depending on the brush type you choose.

- 6 Set the Tubes Sections to a value of 18.
- 7 In the Attribute Editor, close the Mesh attributes, and then expand the Tubes attributes, then the Creation attributes.



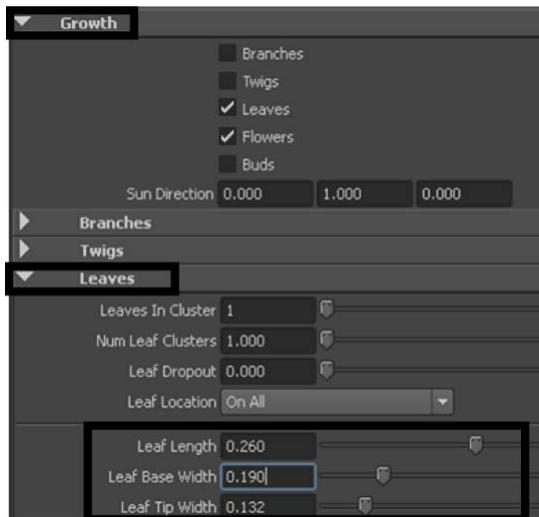
The Creation attributes control the main body of the teapot; the “trunk” or “stalk” of your paint effect mesh type.

- 8 Scroll down to the Width Scale attributes.
- 9 In the Width Scale attributes, drag an index point handle (as shown below) to change the profile shape of the teapot.
The profile of the teapot body is modified.

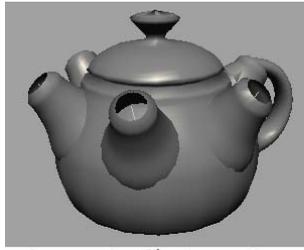


The Width Scale graph allows you to control the width of a tube by interpolating the shape between the index points. (If you look closely at the graph, you'll see the profile of the teapot laying on its side.)

- 10 Drag the index marker back to its original position.
- 11 Close the Creation attribute section in the Attribute Editor, and open the Growth attributes, followed by the Leaves attributes.



- 12 In the Leaves attributes, drag the sliders for the Leaves In Cluster, Leaf Length, Leaf Base Width, and Leaf Tip Width attributes to experiment with the spout of the teapot.



Leaves In Cluster = 6



Leaf Length = 0.4
Leaf Base Width = 0.1
Leaf Tip Width = 0.075

If you want to adjust the attributes of the teapot handle, you can edit the Flower attributes.

Using this converted poly mesh is one example of how the attributes of the paint stroke can be modified to affect the shape of another object through construction history.

If you delete the paint stroke, the construction history between the stroke and the polygonal teapot will be removed. If you then want to make further changes to the teapot, you must use the Edit Polygon tools.

To render the polygonal mesh teapot

- ▶ Render the teapot by clicking the Render Current Frame button on the Status Line.



Beyond the lesson

In this lesson with Paint Effects, you learned how to:

- Use preset brushes from the Shelf and the Visor.

- Paint directly in 3D space onto the ground plane or onto 3D objects.
- Select and move strokes in the scene.
- Edit brush and stroke attributes using the Attribute Editor.
- Render paint strokes.
- Use the Paint Effects Panel to paint fully-rendered paint strokes.
- Create a terrain to paint on using the Sculpt Geometry tool.
- Understand the structure and archetype for strokes with tubes.
- Use mesh brush types to paint hard-surface objects.
- Convert mesh brush strokes to polygonal surfaces.

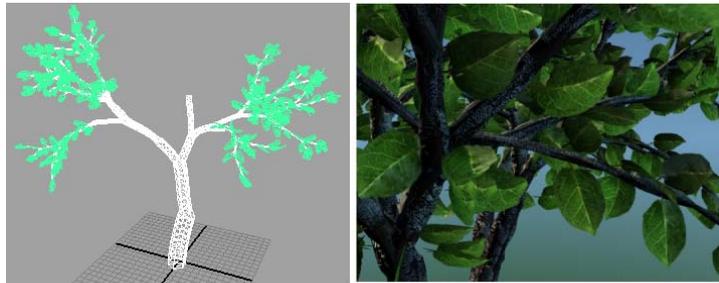


Image created by: Duncan Brinsmead

You can use Paint Effects to create natural or fantastic scenery, backdrops, and atmospheres. Mesh Brush types are particularly useful for creating polygonal trees that appear strikingly convincing, whether they are viewed close-up or from a distance.

Paint Effects has many other useful techniques not covered in this lesson. With Paint Effects you can:

- Modify strokes to share a single brush, make strokes glow, and animate the movement of strokes.
- Attach a brush to an existing curve (you don't need to paint a curve).
- Add spring-like behavior to strokes with tubes, or you can make the tubes react to dynamic forces such as wind.
- Create animated textures; for example, falling rain.

- Paint using a Thin Line brush type which allows you to render large numbers of fine tubes much more quickly than the Paint Brush type. This brush type is useful for painting hair.

Lesson 3: Painting textures on surfaces

Introduction

The 3D Paint Tool allows you to paint textures directly onto a model's surface. In this lesson, you paint colors, patterns, and textures on a dinosaur model. In this lesson you learn how to:

- Prepare a surface model for painting.
- Paint on a surface using both Artisan and Paint Effects brushes.
- Use reflected paint strokes for symmetrical effects.
- Flood fill an entire 3D model with paint.
- Modify brush shapes and stamp settings.
- Paint a bump map texture directly on a surface.

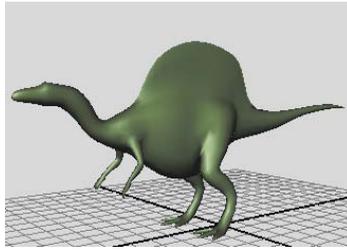
Open the scene for the lesson

You will paint a dinosaur model we prepared for this lesson.

- 1 Do the steps outlined in [Preparing for the lessons](#) on page 562.
- 2 Open the scene file named `3DPaint.ma`.

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

```
GettingStarted/Paint/3DPaint.ma
```



The scene contains a model named Dino. Although Dino is a polygonal model, the 3D Paint Tool works on all surface types: polygonal, NURBS, and subdivision surfaces.

To prepare for this lesson, we assigned a Blinn material named DinoSkin to the model. In Maya, you can paint a surface only if it has a material assigned to it. We also gave the Blinn material a green color. A color is optional, but it provides a base color for the texture you will paint.

Finally, we made other preparations to Dino so you can use the 3D Paint Tool successfully. You'll learn about the preparations in [Preparing UVs for painting](#) on page 623.

- 3 Save the scene in your default scenes directory (File > Save Scene As).
This lesson works best when you start with a saved scene. When you use the 3D Paint Tool, Maya automatically creates a directory to store the file texture that will be created from your painting. By saving the scene, you ensure that Maya can locate the directory and file texture that it creates automatically.
- 4 Select Display > UI Elements > Tool Settings. This displays a Tool Settings panel where the Channel Box is normally located. You will be working with this panel throughout the lesson.

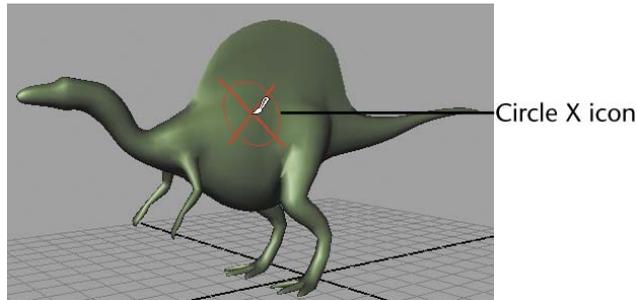
Preparing for painting

- 1 Select Dino. You can paint only on selected objects. When you actually use the 3D Paint Tool, the wireframe of a selected object is not highlighted. You can check the Outliner to be sure the object is selected.
- 2 Select Texturing > 3D Paint Tool. The following warning appears on the Command Feedback line:

WARNING Some surfaces have no file texture assigned to the current attribute.

The 3D Paint Tool requires that the material assigned to your model has a file texture applied to it. Because there is no file texture yet assigned, the warning appears.

When you move the cursor over Dino, you see a circle-X icon that indicates you cannot paint on the model.



- 3 To make sure you have the default settings for the 3D Paint Tool, click Reset Tool at the top of the 3D Paint Tool settings panel.
- 4 In the File Textures section of the 3D Paint Tool settings panel, set the Attribute to Paint setting to Color, then click Assign/Edit Textures.
This step and the next two steps assign a texture to Dino. (Note that you can also paint over an existing texture that you assigned from Hypershade.)
- 5 Type 512 in the Size X box, which is the height of the new file texture in pixels.
By increasing the file texture size, you will see finer detail and resolution in a rendered image of your painting. The number 512 is twice the default size, 256. You could use a higher resolution, but sizes larger than 512 slow the interactive speed as you paint. Notice that Size Y automatically changes to 512. This is because Keep Aspect Ratio is turned on, ensuring that the proportions of the texture are maintained.
- 6 Click Assign/Edit Textures in the Assign/Edit File Textures window.
Maya creates a file texture and assigns it to the Color attribute of the DinoSkin material. At the end of the lesson you will learn about the name and location of the file texture that is created.
You are now ready to paint.

Painting with an Artisan brush

In the following steps, you paint various colors on the texture you just assigned. You also change the size of the brush strokes.

There are two types of brushes you can use. The default is the Artisan brush, which you'll use in the next steps. (Artisan refers to a group of painting tools in Maya that share common tool settings.) The other brush is based on the Paint Effects brushes; you'll use it later in the lesson.

To paint using an Artisan brush

- 1 If you have a tablet and pen, you can experiment with pressure sensitivity as you paint. Open the Stylus Pressure section and turn on Stylus Pressure. (You do not need to have a tablet and pen to complete the lesson.)

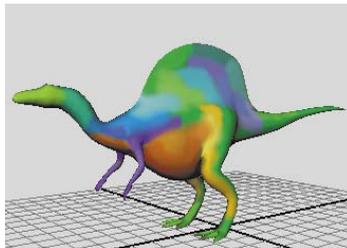
By default, when you paint with Artisan brushes, the pressure you apply to the pen affects the Opacity setting of the 3D Paint Tool. The harder you press, the more opaque the stroke.

- 2 In the Color section of the Tool Settings, click the box next to Color and select a color in the Color Chooser. Click OK.
- 3 Paint a stroke by dragging on the model.
- 4 Select another color from the Color Chooser and paint over your first stroke.

The new color paints over the first color by default. Note that you can blend the colors by choosing a different Blend Mode setting, under the Paint Operations heading. You won't blend the colors in this lesson.

- 5 Move the brush onto Dino, hold down the b key and drag to the left to make the brush smaller. Drag until the number at the brush icon changes to about 0.5. You don't need to be precise.

Use this hotkey when you need to change the brush width. To make the brush wider, drag to the right. You can also change the width by changing the Radius (U) setting.



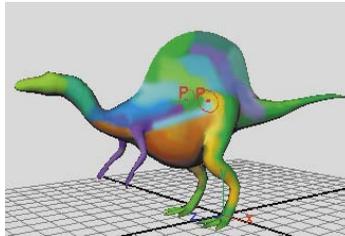
- 6 Experiment with different colors and brush sizes on Dino.
To undo a brush stroke, choose Edit > Undo, Redo, Repeat. The number of strokes you can undo is specified by the Queue Size in the Undo category of the Preferences window (Window > Settings/Preferences > Preferences).
- 7 To remove paint, turn on Erase in the Paint Operations section and stroke the desired region.

Painting symmetrical strokes

You can reflect painted strokes along an invisible axis so that when you paint, the stroke is mirrored on the opposite side of the axis. Reflection is useful for creating symmetrical effects.

To paint using the reflection option

- 1 In the 3D Paint Tool settings panel, open the Stroke section and turn on Reflection and set the Reflection Axis to X.
Notice that there are now two brush outlines that run along an axis. The reflection axis is positioned around a point at the center of the object.



- 2 Paint on one side of Dino, such as one of the legs.
As you paint, the strokes are reflected along the axis and appear on the other side of Dino. Tumble to see the reflected stroke on Dino's other side.

Using Flood All to apply a single color

You can apply a single color to the entire texture without painting individual strokes.

To use Flood All to apply a single color

- 1 In the Flood settings of the 3D Paint Tool settings, click the Color box and select a pale green color from the Color Chooser.
- 2 Set the Flood setting to All and then click on the Flood Paint button.
The Flood Color overwrites all other painting you've done. You can also use Flood to erase painting.
- 3 In the Flood settings, click the Flood Erase button.
Maya eliminates all brush strokes, leaving you with the last saved version of the texture. (To save the texture, click the Save Textures button in the File Textures section.)

Brush shapes

You can paint with different brush shapes to produce paint strokes with hard edges, soft edges, or completely different patterns. You can choose from four standard Artisan brush shapes, plus 40 preset brush shapes.

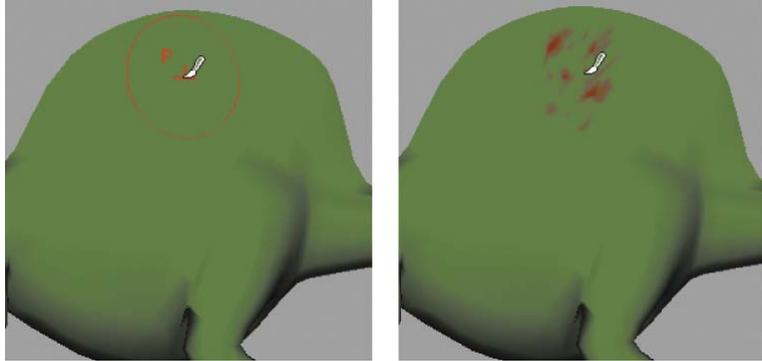
To modify the shape of the brush

- 1 At the top of the 3D Paint Tool settings, click Reset Tool.
Reset Tool cancels the Reflection X and other settings from the previous steps. While painting, it's useful to reset the tool settings from time to time so previous settings do not interfere with your next strokes.
If you are using a tablet and pen, you'll need to turn on Stylus Pressure again to see the effect of your pressure as you paint.
- 2 Change the Color setting to dark red or another color that contrasts with the current green color.
- 3 In the Brush settings, click the following icon and paint a stroke:



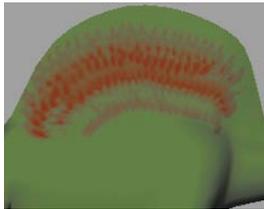
By changing to the square brush shape, your brush stroke has a straight edge. Along with the shapes available from the icons, you can choose from many other unusual brush shapes that are based on image files. You'll do this in the next step. Before you do the next step, undo the brush stroke you just made so it doesn't distract from the next paint strokes.

- 4 In the 3D Paint Tool Brush settings, click the folder  icon.
A browse window opens with a list of various image files. Each file represents a different brush shape.
- 5 Select `dapple.jpg` and click Open.
- 6 Click on Dino's hump and barely move the brush to create a single spot of paint.



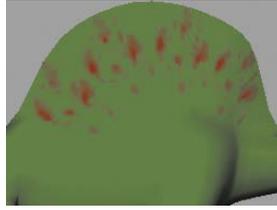
Instead of a solid color, a pattern appears on Dino. The pattern is a result of using the `dapple.jpg` file for the brush shape. Undo the stroke.

- 7 Now drag the mouse over Dino to paint a single, continuous stroke.



The brush shape pattern overlaps itself as you paint, resulting in a striped pattern. Undo the stroke.

- 8 To avoid the overlapping pattern of the prior step, increase the Stamp Spacing (in the Stroke section) to 50. Paint on Dino with a continuous stroke and you'll see the brush pattern without it overlapping.



The Stamp Spacing setting defines how paint is applied when you stroke the surface. If you set Stamp Spacing to 1, the edges of the brush shape pattern just touch each other. If you set the spacing to be greater than 1, there will be spaces between the brush shape pattern. If you set the spacing to be less than 1, the brush shape pattern will overlap.

Painting with a Paint Effects brush

You can use the 3D Paint Tool with the preset brushes available with Paint Effects. Paint Effects brushes include not only patterns but also images such as flowers, feathers, hair, and fire.

In the following steps you paint on Dino with a Paint Effects brush. Before you begin, consider first becoming familiar with the Paint Effects brushes, as taught in [Lesson 1: Painting in 2D using Paint Effects](#) on page 563.

To paint with a Paint Effects brush

- 1 In the Brush section of the 3D Paint Tool settings, click the Get Brush



icon to select from the preset Paint Effects brushes.

The Visor window opens. The Visor organizes the hundreds of Paint Effects brushes into category folders.

- 2 Scroll to the watercolor folder and click it.

To the right, swatch pictures of the Paint Effects brushes appear for this category.

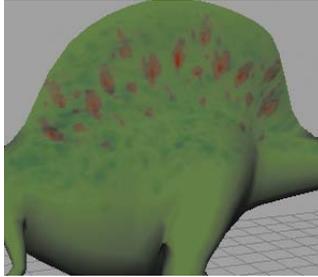
- 3 Select the spatterLight.mel brush. (To see the names of the swatches, drag the pointer over them.)

Leave the Visor window open, but move it so you can see Dino again.

If you are using a tablet and stylus, you need to turn on Stylus Pressure again to see the effect of your pressure as you paint. Stylus Pressure is off because you changed from the Artisan brush to a Paint Effects brush.

For Paint Effects brushes, the pressure you apply to the pen affects three different settings. If you change the settings, it affects only the selected brush. The settings for each brush are separate from each other.

4 Paint on Dino.

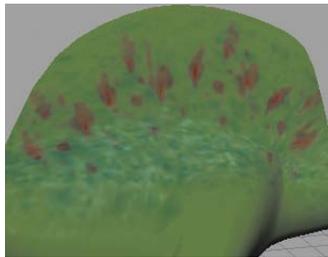


5 In the Brush section of the 3D Paint Tool settings, click the Edit Template Brush  icon to modify other brush settings.

The Paint Effects Brush Settings window opens. Any changes you make to the settings in this window affect the display of your next stroke. As an example, you'll increase the brush's incandescence.

6 Open the Shading section in the Paint Effects Brush Settings window. For the Incandescence1 setting, move the slider to the right and stop midway.

7 Close the Paint Effects Brush Settings window and paint on Dino. Now the paint has an incandescent glow.



Smearing and blurring

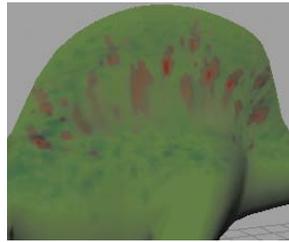
In the next steps you'll smear and blur the paint on Dino. Smearing with a Paint Effects brush blends adjacent colors together along the stroke path. Blurring softens the edges of adjacent colors by averaging their color values.

To smear and blur with Paint Effects brushes

- 1 Move the Visor window back into view.
- 2 Click the wetInWet folder and then click the cleanWaterDrip.mel brush. Close the Visor window.

In the 3D Paint Tool settings, under Paint Operations settings, observe that the Paint Operation is now set to Paint Effects Smear. The brush you selected, cleanWaterDrip, has settings that turn on the Smear setting automatically for you.

- 3 Paint across the side of Dino. The paint smears and produces an effect similar to dripping wet paint.



- 4 In the Paint Operations section of the 3D Paint Tool settings, select Paint Effects Blur.

This selects the default Paint Effects Blur brush.

- 5 Paint across the side of Dino to see the effect of Blur.



- 6 Save the texture by clicking the Save Textures button in the File Textures section.

Painting a bump map texture

Next, you'll paint a bump pattern on Dino. In Maya, a bump texture creates the illusion of surface relief. It does this by perturbing the surface normals to make the surface appear bumpy. This is useful for making Dino's skin appear more "reptile-like".

To paint a bump map texture

- 1 In the 3D Paint Tool window, click Reset Tool to return to the default settings.
- 2 In the File Textures section, set the Attribute to Paint setting to Bump Map.
- 3 Click Assign/Edit Textures.

The Assign/Edit Textures window appears.

Earlier in the lesson, your painting affected only the Color attribute. To paint bumps, it's necessary to select a BumpMap as the new attribute you want to paint, and then assign a new texture for the bumps you'll create.

- 4 In the Assign/Edit Textures window, set both Size X and Size Y to 512 and then click Assign/Edit Textures.

Maya creates a file texture and applies it to the Bump Map attribute of the DinoSkin material.

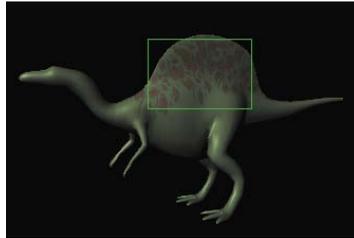
In the scene view, Dino's color changes to white. Bump map textures are based on grayscale colors. By default, when you create a rendered image, bumps will appear in the areas of the surface where there is significant grayscale color contrast. For instance, if a region has black and white stripes, you'll see grooves there. Regions with little or no contrast display no bump.

The bumps will not show in the scene view (unless the scene view is set to High Quality); you must create a rendered image in order to see them. The same is true when you paint certain other attributes, such as Transparency and Incandescence.

Before you paint bumps on Dino, you must set up the Maya panels so you can interactively see the results of your strokes in a rendered image.

- 5 Select Panels > Layouts > Two Panes Stacked.

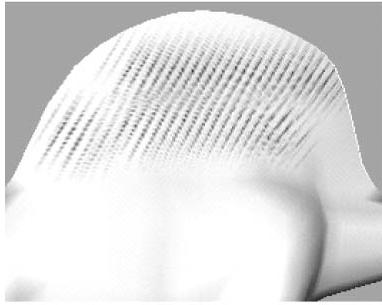
- 6 In the bottom panel, select Panels > Panel > Render View.
Maya now displays the perspective view, the Render View, and the 3D Paint Tool settings. With this arrangement, you'll be able to see your strokes render as you paint.
- 7 In the perspective view, move the camera to a close-up view of Dino's side.
- 8 In the Render View, select IPR > IPR Render > persp. This renders an image of Dino in the Render View.
- 9 In the Render View, drag a selection box around Dino's midsection and hump.



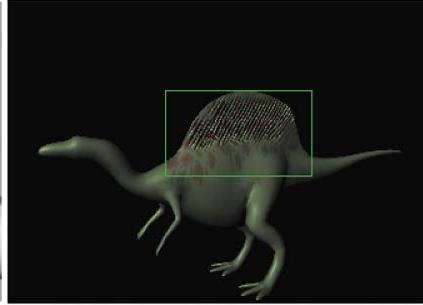
This region will update when you paint.

For the IPR renderer to render bumps as you paint, you must update the bump map texture after each stroke. You can set this to occur automatically.

- 10 In the File Textures section of the 3D Paint Tool settings, turn on Update on Stroke.
For a grooved bump effect, you need to change the brush shape.
- 11 In the Brush section of the 3D Paint Tool settings, click the Browse button.
- 12 Select the hatch.jpg brush and click Open.
- 13 Paint on Dino's side.
Notice the Render View updates with each stroke.



Scene view



Render View

The bumps appear as grooves in Dino's surface because you painted contrasting black stripes on the white texture. In general, gray or black over a white background results in bumps that appear indented.

If you wanted bumps that appear raised on the surface, you could start with a black background and paint gray or white over it. For example, you could flood the texture with a black Flood Color, then paint white or gray strokes rather than the default black.

Greater contrast between the background color and the color you paint results in deeper-looking bumps. Less contrast results in more subtle bumps. In the previous step, you painted using a black color, resulting in prominent bumps. If you change the Color setting to a light gray and paint on the white background, only subtle bumps will appear.

This concludes the lesson.

- 14 Before doing other lessons, restore the Channel Box in the Maya window.
- 15 Select Display > UI Elements > Channel Box/Layer Editor.

Beyond the lesson

In this lesson you were introduced to a few basic techniques related to the 3D Paint Tool. You learned how to:

- Paint textures directly onto surfaces using the 3D Paint Tool.
- Paint symmetrically onto surfaces.
- Paint grey scale attributes to create bump texture effects.

There are several other capabilities in the Paint Operations section, such as the Clone option for Artisan brushes. With the Clone option, you can duplicate

the color and pattern from an area of the texture and then paint that pattern elsewhere on the texture. For details on these options and other 3D Paint Tool techniques, refer to the Maya Help.

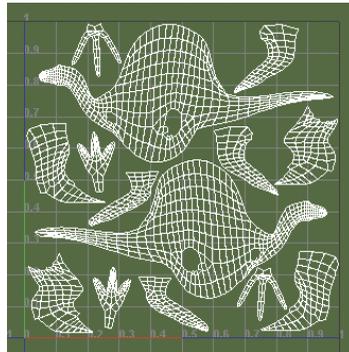
Location of 3D Paint Tool file textures

If you need to move or rename a file texture that Maya creates from your painting, look in the 3dPaintTextures directory under the current project directory. If you paint an existing file texture that you created previously, for example, with a paint program, Maya creates a copy and stores it in the 3dPaintTextures directory.

Preparing UVs for painting

In this lesson you worked with a polygonal model that was specially prepared for painting in this lesson. To paint successfully on a polygonal or subdivision surface, you must set up the surface UVs beforehand. UVs are points positioned around the surface like vertices. They control how Maya stretches a texture around a surface. For polygons and subdivision surfaces, you must arrange the UVs so that textures—including the ones you paint with the 3D Paint Tool—look correct when applied to the surface material. (The UVs on NURBS surfaces are automatically arranged for you, so you do not have to change them.)

You view and edit UVs using the UV Texture Editor (Window menu). You can obtain an introduction to UVs and the UV Texture Editor by completing the UV Texture Mapping lesson in this book. You can also learn more about UVs in the Maya Help by referring to the sections on Polygonal and Subdivisions surfaces.



If you already have experience with editing UVs, remember these two requirements for 3D painting:

- UVs must not overlap.
- UVs must fit with the 0 to 1 texture coordinates.

This illustration shows the UV Texture Editor with the UVs for the Dino model used in this lesson. These UVs meet the above two requirements. Notice that the UVs form several meshes; they do not need to be one solid mesh.

Painting in Screen Projection mode

When you paint on a model, the brush follows the contours of the surface. This typically produces the most desirable results. However, this default behavior leads to uneven paint application in some situations. A common example is along the line where two surfaces join together. It can also occur if the UVs are not arranged correctly, as described previously.

The solution to these situations is to turn on Screen Projection in the 3D Paint Tool settings. Screen Projection applies paint as if you are projecting cinematic film onto a screen. Paint appears where the projection strikes the surface.

The following figure shows an example of this situation with a Paint Effects brush.



Uneven paint stroke with Screen Projection off Screen Projection on

Introduction

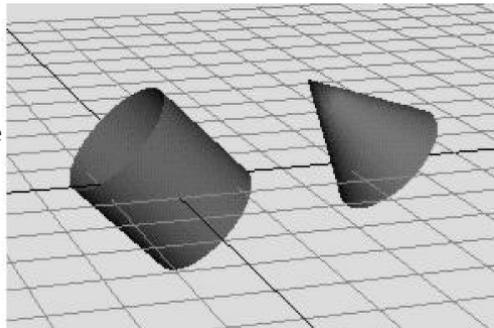
Expressions are program-like instructions you create to control keyable attributes over time. Expressions can be comprised of mathematical equations, conditional statements, or MEL™ commands.

Expressions offer an alternative to difficult keyframing tasks similar to Set Driven Key. With Set Driven Key, attribute relationships can be set up and are defined using animation curves. Unlike Set Driven Key, an expression does not use an animation curve to define the animation of the attribute but calculates the attribute values over time from the formula defined in the expression.

Although you can create an expression to animate attributes for any purpose, they're ideal for attributes that you want to change incrementally, randomly, or rhythmically over time.

It is possible to create custom attributes for objects and control them in conjunction with expressions. It is not possible to mix expressions with other animation techniques for the same attribute on an object (keyframing, path animation, set driven key).

In this example, an expression links the rotate attribute of the cylinder to the cone



This chapter contains the following lessons:

- Lesson 1 Creating a simple expression: [Introduction](#) on page 627
- Lesson 2 Conditional expressions: [Introduction](#) on page 634
- Lesson 3 Controlling particle attributes: [Introduction](#) on page 643

Preparing for the lessons

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

- 1 Select File > New Scene to 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](#) on page 25.
- 3 Select Window > Settings/Preferences > Preferences and do the following:
 - Click the Settings category and set the Time option to Film (24 fps) so your animation plays at the default rate of 24 frames per second.
 - Click the Time Slider category and enter 0 and 300 for the Playback Start/End values. (Press the Enter (Windows and Linux) or Return (Mac OS X) as necessary.)
 - Set the Playback Speed to Play every frame.
 - Click Save in the Preferences window.

At this playback speed and frame range, the animation results of the expressions you'll create in the lessons will be easier to examine and understand.

- 4 With the pointer in the perspective view, press 5 (for Shading > Smooth Shade All). It's easier to see the expression effects on shaded objects than wireframe objects in the following lessons.

Lesson 1: Creating a simple expression

Introduction

This lesson introduces you to some basic concepts regarding the creation and editing of expressions. You learn how to:

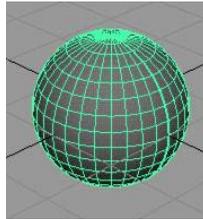
- Use the expression editor to create and edit expressions.
- Control both single and multiple attributes of objects in your scene.

Creating expressions to control a single attribute

In the following steps, you create a default sphere and write an expression to increase its Scale Y attribute as the animation time increases during playback.

To create an expression

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 626.
- 2 Create a NURBS or polygonal sphere. In the Attribute Editor, click on the transform node (nurbsSphere1 or pSphere1).
- 3 Ensure that the X scale, Y scale, and Z scale values are equal to 1. Rename the sphere `Ball`. (Upper and lower case text is important)

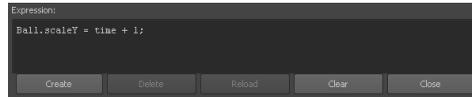


- 4 With `Ball` selected, choose `Window > Animation Editors > Expression Editor`.
- 5 In the Expression Name box, you can optionally enter an expression name, for instance, `ScaleBallHeight`.
If you write several expressions in the same scene, giving each a name might make it easier to find the desired expression if you decide to alter

it later. If you don't provide a name, the expression receives a default name, for instance, expression1.

Note that the Attributes list displays Ball's keyable, unlocked attributes—the attributes you'll most likely want to animate with an expression. Use the scroll bar to see the entire list.

- 6 Enter this expression in the expression text area:



Type all characters exactly as shown above. Entries are case sensitive. The semicolon (;) signifies the end of the expression statement. End each statement in an expression with a semicolon.

An error message appears in the Script Editor and Command Line's response area if the expression has incorrect syntax or typing mistakes.

- 7 Click Create to create the expression.

Creating the expression checks it for syntax errors and converts it to a form Maya can execute when you return to the start time or play the animation. Clicking Create also executes the expression for the current frame.

The expression sets Ball's scaleY attribute to the value of time + 1. Ball.scaleY is the full name of the attribute. A period separates the name of the object and attribute. Note that you must spell the object and attribute with uppercase and lowercase letters as they appear in the Expression Editor's Objects and Attributes lists.

The word time is a commonly used predefined variable that updates as an animation plays. It contains the elapsed number of seconds from the first frame to the current frame. The value increases with the increasing frame number.

At the default animation playback rate of 24 frames per second, time has these values (rounded to four decimal places):

Frame	Time (seconds)
0	0
1	0.0417
2	0.0833

Frame	Time (seconds)
3	0.125
24	1.0
240	10.0

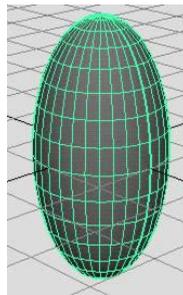
You can find the time elapsed in the animation at any frame by calculating this formula with a desktop calculator:

For example, if the frame rate is 24 frames/second and the animation is at frame 1, the elapsed time is 1 divided by 24, or 0.0417. At frame 6, the elapsed time is 6 divided by 24, which equals 0.25. Note that if your frame rate was 30 frames/second the calculated values would be different.

- 8 Go to the start time and play the animation. Ball's scaleY attribute increases as the time increases:

Frame	Time (seconds)	Ball.scaleY (time + 1)
0	0	1
1	0.0417	1.0417
2	0.0833	1.0833
3	0.125	1.125
24	1.0	2.0
240	10.0	11.0

Maya executes the expression each frame. This causes the object size to scale along its Y-axis, stretching its height during playback.



To see the value of Ball's scaleY attribute at any particular frame, select Ball, display the Channel Box, and stop the animation at the desired frame. The Channel Box updates its values after you stop the animation.

The scaling is smooth because the geometry stretches in synch with the small time increments of the animation playback.

- 9 Close the Expression Editor window.

Editing expressions

To edit an expression after you've created it and closed the window, you must know how to find the expression in the Expression Editor. The next steps show how to find and edit the expression. Specifically, you'll change the expression to alter how quickly the scaleY value changes.

To edit the expression

- 1 To display the Expression Editor again, select Window > Animation Editors > Expression Editor.
- 2 To find the expression, make sure the Ball is selected. In the Attributes list of the Expression Editor, select scaleY, the attribute controlled by the expression.

If you don't see the scaleY attribute, scroll through the Attributes list as necessary. You can also use the Select Filter menu option to customize how you search for your expressions within the Expression Editor.

The expression you created previously appears in the expression text field:

```
Ball.scaleY = time + 1;
```

- 3 Change the expression to this:

```
Ball.scaleY = time/5 + 1;
```

By dividing time by 5, you'll make the Y scaling increase one-fifth as fast as with the previous version of the expression.

Note that you can use the keyboard commands Ctrl-c (Windows and Linux) or Control-c (Mac OS X), Ctrl-x or Control-x, and Ctrl-v or Control-v to copy, cut, and paste text in the Expression Editor and elsewhere in Maya.

- 4 Click Edit to update the modified expression.

Clicking Edit does the same action as clicking Create. Create exists only for new expressions. Edit replaces Create when you display an existing expression.

- 5 Play the animation to see the results of the new expression.
The sphere scales at a slower rate than before.
- 6 Stop the animation and go to the start time.
- 7 At this point and in subsequent steps, save the scene if you think you'll want to examine it at a later date.

Using expressions to control multiple attributes

You can change the expression to control two or more attributes of Ball as in this example:

```
Ball.scaleX = time/2 + 1;  
Ball.scaleY = time/3 + 1;  
Ball.scaleZ = time/5 + 1;
```

Maya scales the Ball at different rates for the X, Y, and Z dimensions.

Linking multiple attributes on the same object

You can change the expression to link the value of one attribute to another as in this example:

```
Ball.scaleX = time + 1;  
Ball.scaleY = Ball.scaleX;  
Ball.scaleZ = Ball.scaleX;
```

The second statement sets Ball.scaleY to the value of Ball.scaleX. Because you've set Ball.scaleX to the value of time + 1, Ball.scaleY also has the value of time + 1. You're linking one attribute's value to another. The third statement also sets Ball.scaleZ to the value of the attribute Ball.scaleX.

The advantage of this expression is that if you assign a different value to Ball.scaleX in the first statement, the second and third statements automatically receive the new value.

Controlling attributes in two objects

You can add an object to the scene and use the expression to control any of its attributes also. For example, suppose you added a default NURBS or polygonal cylinder named Drum to the scene. You could control the attributes of Ball and Drum with the same expression as in this example:

```
Ball.scaleY = time/2 + 1;  
Drum.scaleX = time + 1;  
Drum.scaleY = time + 1;
```

You can link attributes in different objects—so a change in one attribute alters the behavior of the other. For example, you could cause Drum’s scaleY value to always equal two times Ball’s scaleY value as in this example:

```
Ball.scaleY = time + 1;  
Drum.scaleY = Ball.scaleY * 2;
```

If you decide to control attributes in two (or more) objects, you can select either object to write the expression. In fact, any object or node in a scene can be selected when you write an expression to control an object other than a particle object.

Alternatively, you can write two (or more) expressions, for instance, one that controls Ball.scaleY and one that controls Drum.scaleX and Drum.scaleY.

The advantage of creating separate expressions is that you’ll have two expression names, each presumably named after the object and attribute you’re controlling. Having two expression names makes it easier to find the expression that controls the desired attribute.

The advantage of using a single expression to control the attributes is that all statements are in a single expression. You don’t need to edit two expressions.

Beyond the lesson

In this lesson you learned how to:

- Create and edit simple expressions using the Expression Editor.
- Use variables, for example *time*, to control the specific attributes within expressions.
When you write expressions, it is common to assign the built-in *time* variable to an attribute so the attribute value predictably increases as the animation plays. It’s also useful to link an attribute’s value to the value of

another attribute, such as setting the ball's scale Y and Z attributes to equal the scale X value.

- Control multiple attributes with a single expression.
Your expressions can control multiple attributes of the same object or of multiple objects. Alternatively, you can write a single expression for each attribute or object.

For more information on the various mathematical operators that are possible with expressions, refer to the Maya Help.

Behind the lesson

Although not shown in the lessons, you can decrease an attribute value during playback by subtracting time from some number. Example:

```
Ball.scaleY = 3 - time;
```

This decreases the value of Ball's scaleY attribute for the first three seconds of playback.

When you use the predefined time variable, note the animation start frame value. The lessons in this chapter use a start time of 0. In your work, you might create an animation with a start time of 1. With Maya's default frame rate of 24 frames per second, time is 0.0417 at frame 1.

Because of this small time offset from 0, the prior lesson would have required more steps and instructions to work with frame 1 as the start time. For instance, suppose you use following expression with the start time at 1.

```
Ball.scaleY = time + 1;
```

If you go to the start time, the expression sets the initial value of Ball's scaleY attribute to $\text{time} + 1$, which equals $0.0417 + 1$, or 1.0417. Because Ball's scaleY attribute was 1 when you created it, going to the start time sets scaleY to a value 0.0417 larger than its initial value.

This discrepancy means the Ball scaleY is larger than its scaleX and scaleZ attributes in the first frame of the animation. Although the difference is minor in this example, other cases might be more significant.

To start your animation at frame 1 and get the same result as the example, you can subtract 0.0417 from the attribute:

```
Ball.scaleY = (time - 0.0417) + 1;
```

When you go to the start time, the expression sets Ball's scaleY value to $(0.0417 - 0.0417) + 1$. This equals 1, its original scaleY value.

Lesson 2: Conditional expressions

Introduction

Conditional statements set one attribute or variable for an expression based on the condition of another attribute or variable. This means that when a particular defined condition exists for one attribute then another attribute is changed based on how the expression defines it to do so.

In this lesson you learn how to:

- Use conditional statements to control an expression.
- Refine the conditional statements using if and else statements.

Creating a conditional expression

In the following steps, you create a default sphere and write an expression to increase its Scale Y attribute based on the animation playback time. In the first two seconds of the animation time, Scale Y increases with the value of the time. At two seconds and thereafter, Scale Y no longer increases.

To create an expression using conditional statements

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 626.
- 2 Create a NURBS or polygonal sphere at the origin. In the Attribute Editor, click on the transform node (nurbsSphere1 or pSphere1).
- 3 Ensure that the X scale, Y scale, and Z scale values are equal to 1. Rename the sphere `Balloon`.

TIP You can disable the Interactive Completion option under Create > Polygon Primitives and Create > NURBS Primitives and you will, by default, create a sphere at the origin with an X, Y, Z scale of 1.

- 4 Go to the start time.
- 5 With Balloon selected, choose Window > Animation Editors > Expression Editor.
- 6 Enter this expression:

```
if (time < 2)
```

```
Balloon.scaleY = time;
```

This expression is an if statement. The if keyword causes the expression to make a decision based on a comparison of two or more items. In this case, the expression compares the value of time to the value 2.

The expression checks whether the value of time is less than two seconds. If so, it does the assignment `Balloon.scaleY = time`. If time is not less than two seconds, the assignment doesn't occur.

NOTE When you compare the value of time to a number in an expression, Maya interprets time as seconds rather than milliseconds, minutes, or any other unit of time. In the example, Maya interprets 2 as two seconds.

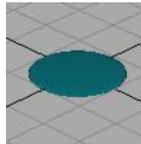
Notice how the assignment `Balloon.scaleY = time` has been formatted so it appears indented under `if (time < 2)`. Maya ignores all indentation, extra spaces, and blank lines between statements. We used the indentation to make the expression easier to read. You could have also written the expression as follows:

```
if (time < 2) Balloon.scaleY = time;
```

This isn't as easy to read. Consistent, organized spacing is a good habit to develop. This lesson attempts to show examples of good spacing style whenever possible.

- 7 Click Create to compile the expression.

The ball flattens.

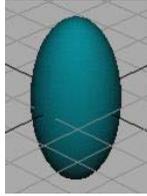


The expression executes when you click Create. Because the animation is at frame 0, animation time is 0. Because time is less than 2, Maya sets `Balloon.scaleY` equal to the value of time, which equals 0.

A `scaleY` value of 0 flattens the object in the Y dimension.

- 8 Playback the animation.

The flattened Balloon's scale increases along its Y-axis. It inflates as the animation plays.



At 2 seconds and thereafter, Balloon no longer inflates.

When time equals 2 or more, the if condition is no longer true. The statement that follows it, `Balloon.scaleY = time`, no longer executes. The value of the `scaleY` attribute stays at the last value it had before time became 2, specifically, 1.9583.

Recall that this example uses a frame rate of 24 frames per second. The time and `Balloon.scaleY` have these values at various frames:

Frame	Time (seconds)	Balloon.scaleY (time)
0	0	0
1	0.0417	0.0417
2	0.0833	0.0833
3	0.125	0.125
24	1.0	1.0
47	1.96	1.9583
48	2.0	1.9583
49	2.04	1.9583

The if statement's condition, `(time < 2)`, is a comparison. The condition must be surrounded by parentheses to isolate it from assignment that follows it.

The `<` in the condition is a relational operator. A relational operator tests how one value relates to another. In the example, the `<` tested whether time is less than 2.

Besides the `<` operator shown in this example, there are several other relational operators such as `>`, `>=`, `==`, and so on. See the Maya Help for more details.

- 9 Stop the animation and go to the start time. Balloon flattens again because the `scaleY` attribute becomes 0 when you go to the start time. Time is 0, so `scaleY` is 0.

Other conditional statement options

You can make Balloon rise after it inflates by adding another if statement to the expression.

To add an if statement to the expression

- 1 Change the expression to this:

```
if (time < 2)
    Balloon.scaleY = time;
if (time >= 2)
    Balloon.translateY = time;
```

- 2 Click Edit to compile the expression.
- 3 Play the animation.

Balloon inflates for two seconds. After two seconds, Balloon stops inflating and its position skips from a Y-axis position of 0 to 2. You'll eliminate the motion skip in a later step.

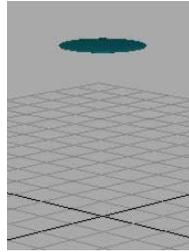
The new if statement increases the `translateY` position of Balloon after the animation time rises above two seconds. The `>=` symbols mean greater than or equal to. Whenever time is greater than or equal to 2, the expression assigns Balloon's `translateY` the value of time. The `translateY` value therefore increases for the rest of your animation's playback range.

Notice that a semicolon ends each statement for a particular condition. Forgetting a semicolon after each statement causes a syntax error, and the changes you've made to the expression won't take effect.

NOTE Always examine the Script Editor for error messages after you edit an expression and click Create. If you alter a previously successful expression and a syntax error occurs, Maya executes the previous successful expression when you play the animation. This might make you believe your changes took effect. Error messages are preceded by the text `// Error:.`

- 4 Stop the animation and go to the start time. Balloon flattens but doesn't return to the origin. (If Balloon has risen out of view, adjust your camera to see it.)

Balloon doesn't return to the origin because the expression doesn't assign Balloon a starting point for the beginning of the animation.



- 5 To make Balloon return to the origin, change the expression to this:

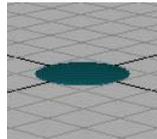
```
if (time < 2)
  Balloon.translateY = 0;
if (time < 2)
  Balloon.scaleY = time;
if (time >= 2)
  Balloon.translateY = time;
```

The new first statement sets `Balloon.translateY` to 0 whenever time is less than 2.

Note that you can put the three statements in any order in this example. When Maya plays each frame, it executes each statement in the expression in the order listed. In this example, the statements work independently, so their order doesn't matter.

We put the statements in the order of time execution because it's easier to see the logic of the expression. If you ever need to change the expression, you'll be able to grasp the expression's actions more quickly.

- 6 Click Edit.
- 7 Stop the animation and go to the start time.



The flattened Balloon returns to its correct position at the origin.

- 8 Play the animation.
Balloon inflates for two seconds, then rises. As it rises, it jumps slightly higher at approximately frame 48.

Fixing a problem in an expression

As mentioned before, Balloon skips from Y-axis position 0 to 2 after two seconds of animation play. You can eliminate the skipping and make Balloon rise smoothly from the origin.

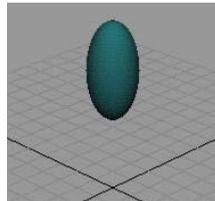
To fix the skipping in the animation

- 1 Stop the animation and go to the start time.
- 2 Change the expression as follows. (Changes are displayed in bold print.)

```
if (time < 2)
    Balloon.translateY = 0;
if (time < 2)
    Balloon.scaleY = time;
if (time >= 2)
    Balloon.translateY = time - 2;
```

- 3 Click Edit.

Playback the animation. Balloon inflates for 2 seconds, then rises slowly and smoothly with time, from its position at the origin.



When time is greater than or equal to 2, the translateY position of Balloon becomes 2 minus 2, which is 0. As time increases beyond 2 seconds, the translateY position increases in the same increments that time increases.

- 4 Stop the animation and go to the start time.

Using else statements

The expression achieved the desired result, but with unnecessary complexity. You can use an if-else statement to make the statement more compact and easier to read.

- 1 Change the expression to this:

```
if (time < 2)
```

```
Balloon.translateY = 0;
if (time < 2)
    Balloon.scaleY = time;
else
    Balloon.translateY = time - 2;
```

2 Click Edit.

3 Play the animation.

The animation plays back exactly as before.

This additional variation on the if conditional statement includes the *else* statement. It is used as an option when the if statement condition is not true. The else keyword sets Balloon.translateY to time - 2 when (time < 2) is false. In English terms, the combination of the if and else statements means: If time is less than two seconds, set Balloon.scaleY to the value of time. Otherwise (when time is greater than or equal to two seconds), set Balloon.translateY to time minus two.

At any instant in the animation's playback, either Balloon.scaleY = time executes or Balloon.translateY = time - 2 executes. Under no circumstances can they both execute. The else statement executes only when the if condition that precedes it is false.

The first if statement executes whenever time equals 0. It is unrelated to the if-else statements.

Using else statements instead of multiple if statements makes an expression simpler to read. If you use an if-else construction instead of a lengthy list of if statements, you'll also improve the execution speed of the expression. This improves your animation's playback and rendering speed.

Either expression is valid. If using the if-else construction seems confusing, stick with multiple if statements.

You can accomplish most expression animation tasks with several if statements strung after one another.

4 Stop the animation and go to the start time.

Simplifying expressions

You can simplify the expression to make it easier to read.

1 Change the expression to this:

```
if (time < 2)
{
    Balloon.translateY = 0;
    Balloon.scaleY = time;
}
else
    Balloon.translateY = time - 2;
```

We removed the second `if` statement: `if (time < 2)`. In its place, we enclosed the remaining statements with braces `{` and `}`. Maya evaluates both the statements between the braces if the condition `(time < 2)` is true. Setting `Ball.translateY` to 0 here instead of in a separate `if` statement makes the expression easier to read and comprehend. Reducing the statement in this way also makes it more efficient for Maya to process and easier to troubleshoot if you encounter a problem.

Note that you can put multiple statements between braces for an `else` statement, just as you do for an `if` statement.

- 2 Click Edit.
- 3 Play the animation.
The animation plays exactly as before with the new expression.
- 4 Stop the animation and go to the start time.

Editing expressions to refine an animation

You can further refine the animation by expanding Balloon more slowly.

To edit the expression to scale the balloon more slowly

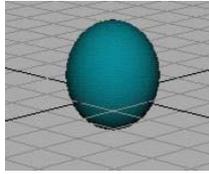
- 1 Change the expression to this:

```
if (time < 2)
{
    Balloon.translateY = 0;
    Balloon.scaleY = time * 0.6;
}
else
    Balloon.translateY = time - 2;
```

(The asterisk `*`) multiplies time by 0.6.)

- 2 Click Edit.

3 Play the animation.



The `scaleY` attribute increases at 60% of the value of time, so Balloon expands slower during playback. (The number 0.6 equals 60%.) By the time Balloon starts to rise, it has expanded to the size of a typical balloon.

To know whether to multiply time by 0.6 or some other number, you need to experiment.

For example, you might multiply by various percentages such as 0.2, 0.5, 0.75, and finally 0.6. The 0.6 creates a life-like balloon shape at two seconds.

4 Stop the animation and go to the start time.

You can further refine Balloon's appearance by eliminating the flattened Balloon that appears at the origin when you go to the start time. You can also scale Balloon at different rates along each of its three axes.

To edit the expression to change the balloon's initial scale

1 Change the expression to this:

```
if (time < 2)
{
    Balloon.translateY = 0;
    Balloon.scaleY = time * 0.6;
    Balloon.scaleX = time * 0.5;
    Balloon.scaleZ = time * 0.5;
}
else
    Balloon.translateY = time - 2;
```

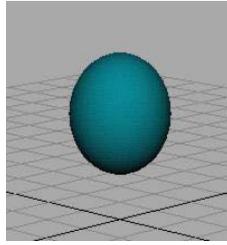
2 Click Edit.

At frame 0, Balloon disappears from view because its scale attributes are 0.

The `scaleX`, `scaleY`, and `scaleZ` attributes are 0 at frame 0 because time is 0. Any number multiplied by 0 is 0.

3 Play the animation.

As time increases, the value of Balloon's scale attributes increase.



Because the expression sets `scaleX` and `scaleZ` to 50% of the value of time, these dimensions scale slower than `scaleY`, which is set to 60% of the value of time. Balloon scales faster in height than in width or depth. This is true for many real balloons.

- 4 Stop the animation and go to the start time.
- 5 Save the scene if you plan to examine it later.

Beyond the lesson

In this lesson you learned how to:

- Use conditional statements to control an expression.
You can accomplish many expression tasks with several `if` statements strung after one another. However, using `else` statements instead of multiple `if` statements makes an expression simpler to read.
If statements and *else* statements are two of the most common programming features available in expressions. If you're familiar with programming, you can write expressions with an extensive syntax similar to the C language.

For further information and related techniques, refer to the Maya Help.

Lesson 3: Controlling particle attributes

Introduction

You can use an expression to control attributes of particle objects. Particle expressions can be more complex than other types of expressions. For example,

you can write an expression to control all particles in an object the same way, or you can control each particle differently.

In this lesson you learn how to:

- create a particle object
- create a creation expression to set the initial color of the particles
- create a runtime expression to control the color during playback
- use a random function to randomize how the color changes
- control the timing of an expression

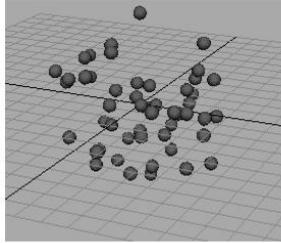
Creating particle objects

In the following steps, you'll create a particle object made of 100 randomly positioned particles displayed as tiny spheres. You'll also dynamically add an attribute to the particles. To color particles with an expression, Maya requires you to dynamically add the appropriate attribute.

To create a particle object

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 626.
- 2 From the Dynamics menu set, select Particles > Particle Tool >  .
- 3 In the Tool Settings window, click Reset Tool, and set the following options:
 - Particle Name: `Bubbles`
 - Number of Particles: 100
 - Maximum Radius: 5
 - Click in the scene view (roughly in the center) to create the 100-particle object.
- 4 From the Toolbox, select the Selection Tool so that particles become selected.
- 5 With Bubbles selected, display the Attribute Editor.

- 6 In the Attribute Editor, select the BubblesShape tab, and then open the Render Attributes section and set the following:
 - Particle Render Type: Spheres
 - Add Attributes For: Click Current Render Type.
 - Radius: 0.5



- 7 In the Add Dynamic Attributes section of the Attribute Editor, click Color. A window appears that prompts you to choose whether to add the attribute per object, per particle, or connected to a shader.
- 8 Turn on Add Per Particle Attribute, then click Add Attribute. This adds an RGB PP attribute to the particle shape node for Bubbles. Because you're adding this attribute as a per particle attribute, you can give each particle a different color.

Using creation expressions to set a constant color

To set an attribute that doesn't change during the animation (the particle color), you'll use a *creation expression*. A creation expression executes when you go to the start time. It doesn't execute while the animation plays. (For emitted particles, a creation expression executes for a particular particle when it is emitted.)

To create a creation expression

- 1 In the Per Particle (Array) Attributes section of the Attribute Editor, right-click the RGB PP box and select Creation Expression from the pop-up menu. Note that the shape node of Bubbles (*BubblesShape*) is displayed in the Expression Editor. When you use an expression to control particle attributes, make sure the selected object in the Expression Editor is a particle shape node, not the

transform node of the particle object. If a particle object's transform node is selected, move the mouse pointer to the scene view and press the down arrow key to select the particle shape node.

- 2 Enter this expression and then click Create:

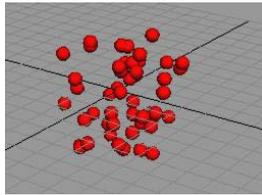
```
BubblesShape.rgbPP = <<1,0,0>>;
```

When you click Create in the Expression Editor, Maya checks the syntax of the expression. Assuming you made no typing errors, the expression executes once for each of the 100 particles.

The expression colors all particles in the object red. The double angle brackets << and >> enclose a vector that sets the red, green, and blue components of the RGB PP attribute to 1, 0, and 0. In the RGB color scheme, this gives the object a red color.

If the particles are uncolored, check that you've turned on Shading > Smooth Shade All. This shading mode is necessary for particles to show the color assigned in an expression.

- 3 Play the animation. The particles remain red.



Using runtime expressions

There is one other type of expression you can create for a particle object (shape node)—a runtime expression. By default, a runtime expression executes each frame during playback. A runtime expression does not execute when you go to the start time (or at the time in which a particle is emitted).

You'll often create both types of expressions for a particle object—a creation expression that initializes an attribute value in the first frame, and a runtime expression that controls the attribute value in subsequent frames.

The following steps show how to create a runtime expression to change particle color during playback.

To create a runtime expression

- 1 In the Per Particle (Array) Attributes section of the Attribute Editor, right-click the RGB PP box and select Runtime Before Dynamics Expression from the pop-up menu.

- 2 Enter this runtime expression:

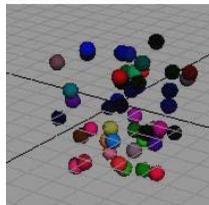
```
BubblesShape.rgbPP = sphrand(1);
```

- 3 Click Create to compile the expression.

- 4 Play the animation.

The particles flicker in random colors as the animation plays back. The runtime expression controls the RGB PP attribute during playback. Because RGB PP is a per particle attribute, the runtime expression executes for each particle in the object for each frame. For each particle, the expression assigns the RGB PP attribute the output from the execution of the *sphrand* function with an argument of 1. The *sphrand* function is one of Maya's many built-in mathematical functions that are useful in expressions.

The *sphrand* function with an argument of 1 assigns each particle's RGB PP color a random vector. The vector represents a random point in a spherical region of radius 1. The left, middle, and right RGB PP color components get a value no less than -1 and no greater than 1. (R, G, and B values less than 0 are treated as 0—a black color).



The *sphrand* function returns a different random vector each execution, so each particle receives a different random RGB PP value, and therefore, a different color. The color changes each frame.

- 5 Rewind and play the animation again.

The particles become red when you go to the start time, and random colors during playback.

NOTE You can use only one creation and one runtime expression per particle object (shape node) in your scene. To control multiple attributes of a single particle object, you must do so within one creation expression and one runtime expression. You can't create a separate expression for each attribute as you can for other types of objects. You therefore don't need to select an attribute from the Expression Editor's Attributes list.

Modifying runtime expressions

You might have trouble seeing the color of a particle in any instant because the color changes so quickly. You can slow the change of colors to create a flashing Christmas light effect. The following steps make the particles change colors every second of animation.

To adjust the timing of the change of color

- 1 Change the runtime expression to this:

```
if ((frame % 24) == 0)
    BubblesShape.rgbPP = sphrand(1);
```

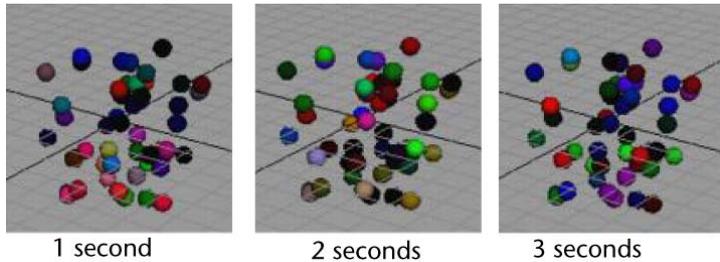
This expression uses the modulus operator (%) to control when the RGB PP attribute of the particles receives a random color. The modulus operator returns the remainder after division. For example, 24 divided by 24 returns 0, but 25 divided by 24 returns 1. (Dividing 25 by 24 equals 1 with a remainder of 1.)

If the value of frame divided by 24 is equal to any number with a remainder of 0, the assignment to `BubblesShape.rgbPP` occurs. In other words, the assignment occurs when frame equals 24, 48, 72, and so on. At an animation rate of 24 frames/second, the assignment happens once each second.

The `==` symbols mean is equal to. In conditional statements, be careful to type `==` rather than `=`. The `=` symbol means assign the value to.

- 2 Rewind and play the animation again.

When you go to the start time, the particles turn red because the creation expression executes. When the animation plays, the particles receive a random color once each second.



Note that you can change the beginning red color to random colors by changing the creation expression to this:

```
BubblesShape.rgbPP = sphrand(1);
```

This is the same expression as the runtime expression.

- 3 Save the scene if you plan to examine it later. This concludes the lesson.

Beyond the lesson

In this lesson you learned how to:

- Control particles using creation and runtime expressions.
- Apply a random function to your expression.

For a particle object, you typically create *two* expressions—a creation expression that initializes an attribute value in the first frame, and a runtime expression that controls the attribute value in subsequent frames. (Creation and runtime expressions exist only for particle objects, not for other types of objects.)

This lesson described how to color stationary particles as a simple way to describe creation and runtime expressions. However, it's more typical to use particle expressions to:

- Create sophisticated particle motion and life span; for example, the movement and fading of exploding fireworks.
- Create complex colors for emitted particles, for example, rocket exhaust flames.
- Create complex colors and movement of particles following collision with geometry, for example, sparks resulting from fired bullets. Creating such effects requires experience and experimentation. Expression writing can be as complex as computer programming. A detailed

understanding of expression execution and syntax is essential. You also need to become familiar with the built-in mathematical functions such as the sphrand function introduced in this lesson.

For further information and related techniques, refer to the Maya Help.

Introduction

Scripting languages in Maya allow you to:

- Automate repetitive or frequently performed tasks.
You can write scripts to execute operations that would normally be selected from the user interface or performed using tools and user manipulation.
- Change Maya's user interface.
You can create windows and controls that affect objects in the scene.
- Establish development standards.
You can enforce naming conventions, workflows and model scales.
- Create new tools and functions.
You can extend Maya's functionality in ways that help you do your work.

In this chapter, you learn some common scripting techniques and features that highlight the capabilities of scripting in Maya:

- [Some basic concepts](#) on page 652
- Lesson 1 Commands in MEL: [Introduction](#) on page 656
- Lesson 2 Saving scripts to the Shelf: [Introduction](#) on page 666
- Lesson 3 Using Variables with MEL: [Introduction](#) on page 675
- Lesson 4 Procedures and user interface creation: [Introduction](#) on page 685
- Lesson 5 Using Python in Maya: [Introduction](#) on page 705

Some basic concepts

The following section introduces some basic concepts that help you understand what's going on when you use scripting in Maya.

The command architecture

Maya has a command-based architecture for handling all of its operations. The commands for all of Maya's functionality are accessed through scripting language commands tied to the Maya user interface—menus, tools, dialog boxes; in fact, just about anything you interact with in Maya. As well, you can explicitly enter Maya commands in the scripting languages that Maya supports.

Every Maya command has multiple arguments that set the way in which the command executes. Arguments and their implicit or explicit values are required for a command to execute.

To set argument values, the scripting languages in Maya use flags. Flags tell the scripting language which argument of the Maya command is set with what value. The value that is assigned to the argument follows the flag.

Flags have short and long names in MEL and Python. Commands execute identically whether or not you use the short or long name of the flag. Long names are often descriptive and useful for new users; short names abbreviate flag names for power users.

Scripting languages in Maya

Maya supports two scripting languages:

- MEL (Maya Embedded Language) is a proprietary scripting language modeled after Unix shell script.
- Python is a recent addition to Maya that provides the same access to Maya's commands as MEL. Python is a widely-used modern programming language.

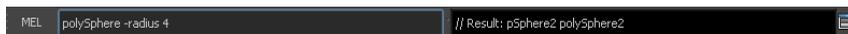
Maya's user interface is built using scripts that execute (call) Maya commands. As Python is a recent addition to Maya, the majority of the scripts accessed through the user interface are written in MEL.

For example, when you click the sphere icon on the Shelf or select an item from a menu, Maya calls MEL commands to create a sphere or execute the command associated with the menu item.

Entering commands in Maya

There are several ways to explicitly enter scripting commands in Maya.

- To enter single line MEL and Python commands, use the Command Line. To show the Command Line if it is hidden, select **Display > UI Elements**. The Command Line is located near the bottom edge of the Maya interface.

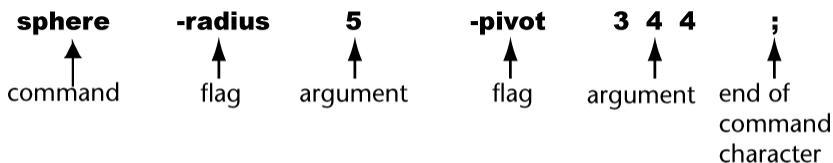


- To enter multi-line MEL and Python commands, use the Script Editor. The Script Editor displays a history of executed commands, as well as the results and outputs of commands. To open the Script Editor, select **Window > General Editors > Script Editor**.
- All MEL commands are highlighted as you enter them into the MEL tab in the Script Editor.
- As well, the Expression Editor uses a subset of MEL commands to create animation in the scene. MEL commands in the Expression Editor are evaluated for every frame in the animation. The Expression Editor does not support Python commands. To open the Expression Editor, select **Window > Animation Editors > Expression Editor**.

You can also load external MEL and Python files and run them using the Script Editor. For more information, see [Loading a script file](#) on page 692.

What MEL looks like

Invoking commands in MEL has the basic structure of a command followed by a combination of flags and arguments. Flags are prefaced by the hyphen character in MEL.



The above MEL command creates a NURBS sphere of radius 5 with its pivot at the origin and the sphere offset from the pivot in XYZ by 3 4 4.

All MEL commands are case-sensitive; `SPHERE` is not the same as `sphere` (and returns an error message).

If no flags are provided, the command executes with default arguments. The command fails if it requires arguments that it did not get.

NOTE MEL also has an alternate syntax, which implements commands and flags in a method similar to the C programming language. For more information, see MEL for programmers in the MEL and Expressions guide.

Getting help on MEL

You can get help with MEL in three ways:

- To access help on a particular MEL command, open the MEL Command Reference by selecting Help > MEL Command Reference.
- Help can also be accessed from within the Script Editor by using the Maya Command `help`. For example, entering `help sphere` shows a list of flags that can be used with the `sphere` command.
- More information on all aspects of using MEL commands is also available in the MEL and Expressions guide, part of the Maya Help. Open the Maya Help and select Using Maya > General > MEL and Expressions.

Basics of Python

Python and MEL are scripting languages with equal prominence and capabilities in Maya. Python can access all the Maya commands that MEL does.

Python accesses Maya commands through the Python module `maya.cmds`. Modules in Python are sets of commands that add functionality to Python, and must be imported before using any of the commands from a module.

Flags in Python are implemented through Python's named arguments. You specify which argument to modify and use the assignment operator (=) to assign a new value to the argument.

In order to run this or any example command, you must have imported the Python module first. Enter `import maya.cmds` before starting to enter Python commands.

maya.cmds.sphere (radius = 5, pivot = (3, 4, 4))

from the
maya.cmds
module

command

flag

assignment
operator

argument

flag

assignment
operator

argument

The above Python command creates a NURBS sphere of radius 5 with its pivot at the origin and the sphere located at XYZ co-ordinates 3,4,4.

All Python commands are case-sensitive; `maya.cmds.SPHERE` is not the same as `maya.cmds.sphere` (and returns an error message).

Getting help on Python

You can get help with Python in several different ways:

- To access help on a particular Maya (Python) command, open the Python Command Reference by selecting Help > Python Command Reference.
- You can also access help on Maya Python commands from within the Script Editor by using the Maya Command `help()`. For example: typing `maya.cmds.help("sphere")` outputs a list of possible flags that can be used with the `sphere` command.
- Help for native Python commands is accessed from within the Script Editor by typing `help("command")` where `command` is the Python command you want to access help for.
- More information on using Python commands is also available in the Python guide, part of the Maya Help. Open the Maya Help and select Using Maya > General > Python.
- All Python commands are highlighted as you enter them into the Python tab in the Script Editor.
- There are existing reference materials and documentation for Python on the web, as well as many downloadable modules to add extra functionality to Python.
 - Getting Started tutorials: www.python.org/about/gettingstarted
 - Python documentation: docs.python.org
 - Python modules: cheeseshop.python.org

Preparing for the lessons

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

- 1 Select File > New Scene to 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](#) on page 25.

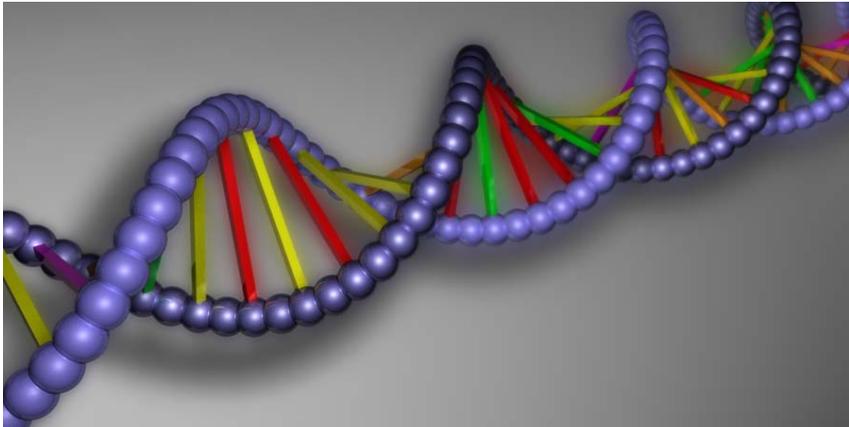
- 3 Make sure the Construction History icon (below the menu bar) is on. (If it is turned off, it has a large X across it.)



- 4 Ensure that the interactive creation option for primitives is turned off by selecting Create > Polygon Primitives > Interactive Creation. The option is off when a check mark does not display beside the item's name in the menu.
- 5 Ensure that the interactive creation option for surfaces is turned off by selecting Create > NURBS Primitives > Interactive Creation. The option is off when a check mark does not display beside the item's name in the menu.
- 6 For any code snippets, we suggest you write them out in the Script Editor rather than copy and pasting them. Certain symbols such as “ and ‘ may not copy and paste correctly.

Lesson 1: Commands in MEL

Introduction



Procedural geometry created with MEL scripting

This lesson provides a brief introduction to using MEL scripting commands. In this lesson, you learn how to:

- Use the Script Editor to view MEL command history
- Enter MEL commands in the Script Editor
- Create geometry with MEL commands
- Modify attributes of objects with MEL commands
- Interpret Script Editor history

Entering MEL commands

You can enter MEL commands in the Command Line or the Script Editor. The Command Line can only accept single line MEL commands, while the Script Editor provides a method to input multiple MEL commands and view the results.

To use the Command Line to input MEL commands

- 1 Select Display > UI Elements.
Ensure that Command Line is checked.



Maya Command Line

- 2 Click in the Command Line text field to position the text cursor in the field
By default the Command Line is in MEL script entry mode. You can change the mode by clicking on the text in the lower left hand corner. For this lesson, ensure the Command Line is in MEL script entry mode.
- 3 Type the following in the Command Line.
`sphere;`
All MEL and Python entries are case sensitive.

- 4 To execute the command, press Enter on the numeric part of your keyboard, or press Ctrl+Enter.
- 5 A NURBS sphere is created at the origin.

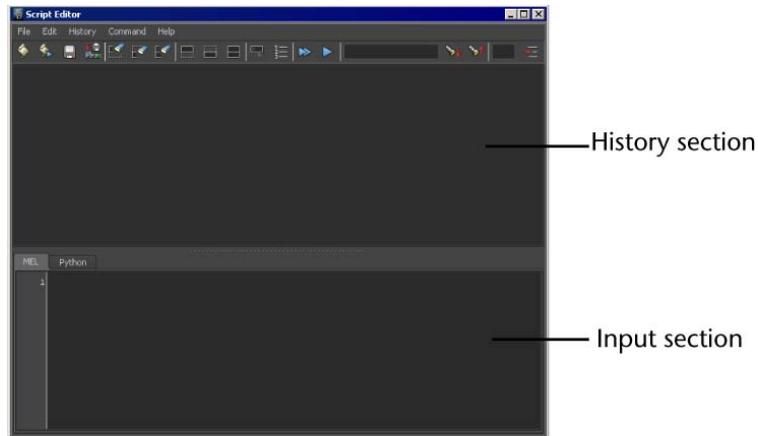
In the Script Editor, the Enter (Windows and Linux) or Return (Mac OS X) key above the Shift key does not execute a command. It starts a new line so you can type several commands before executing them.

Observing script history

The Script Editor displays a running history of executed commands and the results of commands that Maya executes. You can copy most commands from the history section of the Script Editor and paste them in the input section to execute them.

To set up the Script Editor

- ▶ Select Window > General Editors > Script Editor.
The Script Editor window opens.



Script Editor

The lower half of the Script Editor is the input section where you enter scripting commands.

By default, Maya displays basic feedback from executed commands. However, when Echo All Commands is on, Maya shows all the history from a command,

including when a script calls another script or modifies elements of the user interface. While learning MEL, the extra lines of output produced when Echo All Commands is on can be confusing because there is so much output. As you become more experienced in MEL, the extra lines of history can be useful for debugging a MEL script.

- In the Script Editor window, ensure that there is not a check mark beside History > Echo All Commands.

To observe script history

- 1 In the Script Editor, click the Clear History button.



You can more easily isolate the command history created by the commands you perform when there is no other history in the Script Editor.

- 2 From the Shelf, select the Surfaces tab in order to view the tools located on that shelf.
- 3 Click the NURBS Sphere button on the Surfaces Shelf.



Command history is displayed in the history section of the Script Editor:

```
CreateNURBSSphere;
sphere -p 0 0 0 -ax 0 1 0 -ssw 0 -esw 360 -r 1 -d 3 -ut 0 -tol
0.01 -s 8 -nsp 4 -ch 1;objectMoveCommand;
```

Several MEL commands are executed when the Shelf button is pressed. The following happens when the button is pressed:

- The NURBS Sphere button on the surfaces Shelf executes a script called `CreateNURBSSphere;`.
- The `CreateNURBSSphere` script calls the `sphere` MEL command using flags to set the arguments of the command to the values currently specified in the NURBS Sphere menu option window. The `sphere` MEL command is using flags in short form.

- The `objectMoveCommand` is executed every time a primitive is created. It moves the created object to the origin.

Commands in MEL must be separated with semicolons. Additional spaces and additional carriage returns do not affect the execution of the scripts. Blank spaces and returns make scripts easier to read. Flags cannot be separated from the hyphen character by extra spaces or returns.

TIP For more details on these flags, see `sphere` in the MEL Command Reference.

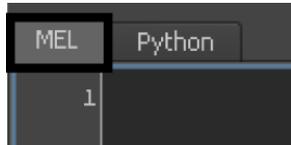
The MEL Command Reference gives a complete listing of the functionality of all MEL commands and flags.

NOTE You can copy commands from the Script Editor history section and paste them in the input section to execute them.

- Delete the NURBS sphere by pressing the delete key on your keyboard.

To type a MEL command in the Script Editor

- 1 Select a MEL tab in the Script Editor.



Script Editor tabs allow you to quickly switch between MEL and Python script entry modes. You can create additional Script Editor tabs to serve as a temporary holding area for scripts.

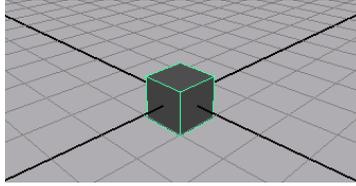
- 2 Type the following command in the input section of the Script Editor.

```
polyCube;
```

TIP When asked to type a command, you can copy it from these tutorials and paste it into the Script Editor input section.

- 3 Execute the command by pressing Enter on the numeric part of your keyboard.

You must always type Enter on the numeric keypad or Ctrl+Enter (the regular Enter on your keyboard) to execute commands. Typing regular Enter just creates a new line in the Script Editor input.



A polygonal cube is created at the origin.

The command is removed from the input section of the Script Editor after execution. The command and the result of the command is output to the history section of the Script Editor:

```
polyCube;// Result: pCube1 polyCube1 //
```

NOTE In future lessons, when a series of MEL or Python commands creates output, only the output created by the commands will be referred to as output to the Script Editor.

Comments in MEL begin with // and anything which follows these characters on a line is ignored by MEL. The same convention is used for displaying results of commands and output in the Script Editor history section. Similarly, results, comments and messages output by Python are prefaced with the Python commenting character (#).

NOTE In the rest of this lesson and the remaining MEL lessons, when asked to type a command, execute the command after typing it with the Enter on the numeric keypad on your keyboard. You can also execute commands with Ctrl+Enter (Windows) or Ctrl+Return (Mac OS X).

TIP If you do not want the command to be removed from the input section of the Script Editor after execution, highlight the command before execution by selecting it with the mouse.

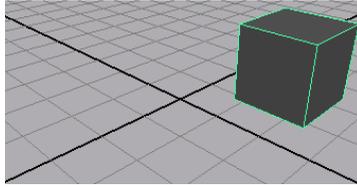
Modifying object attributes

When you perform an action on an object in the Maya user interface, you are using Maya commands to change the attributes of selected objects. You can use the same Maya commands explicitly to edit the attributes of objects in the scene.

To use the move command

- 1 With the polygonal cube selected, type the following in the Script Editor:

```
move 3 2 1;
```



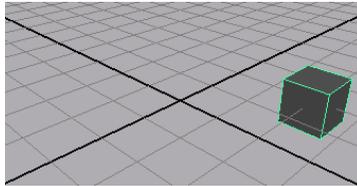
The polygonal cube is moved to the XYZ co-ordinates 3,2,1.

The `move` command has set the translate attributes to the specified arguments of the command. These arguments are called command arguments as they are not used with flags.

When a command is typed into the Script Editor, it acts on the current selection, unless an object name is provided as an argument to the command.

- 2 With the polygonal cube selected, type the following in the Script Editor:

```
move -1 -2 -3 -relative;
```



The polygonal cube moved relative to its current position by -1,-2,-3 in XYZ.

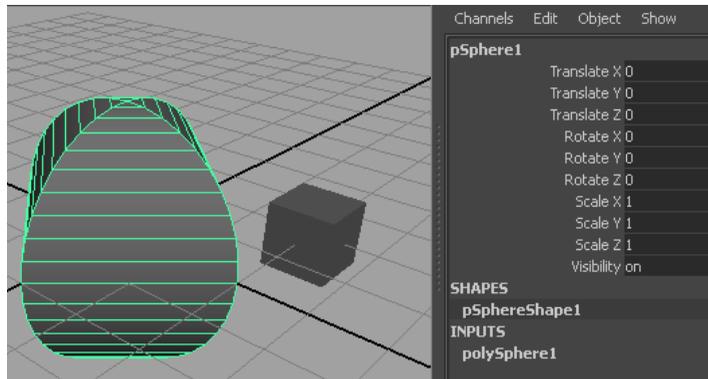
The `move` command is used with the `relative` flag to change the method in which the command executes. The `relative` flag does not have any arguments; the values -1,-2,-3 are command arguments used with the `move` command. The values are prefaced by a negative sign, but are not recognized as flags as they do not contain any of the flag keywords.

This is equivalent to using the Move tool in the Maya interface.

To specify attributes of geometry with flags

- Create a polygonal sphere with a radius of two and four subdivisions in X by typing the following in the Script Editor:

```
polySphere -radius 2 -subdivisionsX 4;
```



The `radius` flag allows you to specify a value for the radius of the sphere. The value 2 is associated with the `radius` flag, and is not a command argument.

This is equivalent to specifying values within the Polygon Sphere Options dialog in the Maya interface.

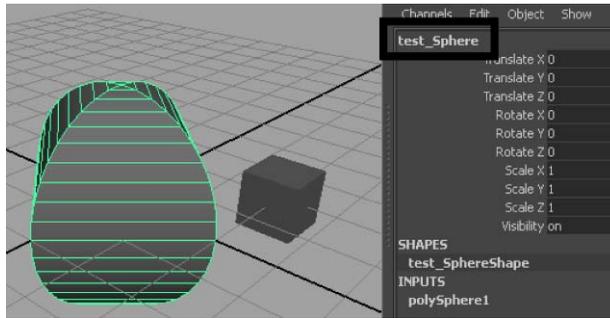
NOTE Flags have short and long names. Commands execute the same way whether or not you use long or short flag names. The above command could have been typed as:

```
polySphere -r 2 -sX 4;
```

Some flags have the same short names, but they are used in different contexts so it does not affect the execution of the script. For example: `-r` is the short name of the `-relative` flag, used with the `move` command, `-r` is also the short name of the `-radius` flag, used with the `sphere` command.

-
- The sphere has a default name of `pSphere#`. Rename the selected sphere by typing the following in the Script Editor:

```
rename test_Sphere;
```



The `rename` command changes the name of the currently selected object. If multiple objects are selected, the `rename` command renames the selected objects to the specified name with a numerical index appended, as every object must have a unique name.

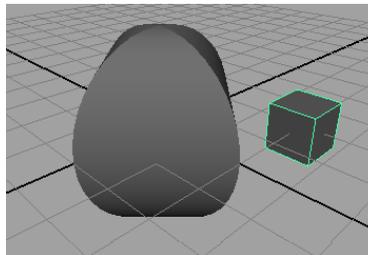
This is equivalent to renaming an object with the Attribute Editor or the Channel Box in the Maya interface.

Editing Objects

Some MEL commands can take an `edit` flag that allows you to make changes to the attributes of an object. The `edit` flag is used in conjunction with other flags specifying which attributes to change.

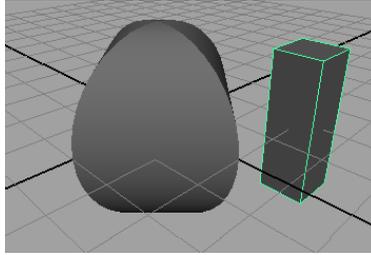
To edit objects with Maya commands

- 1 Select the polygonal cube



- 2 Edit the height of the cube by typing the following in the Script Editor:

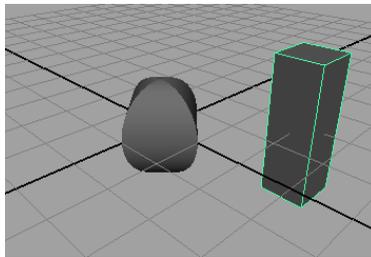
```
polyCube -edit -height 3;
```



This is equivalent to changing attribute values within the Attribute Editor in the Maya interface.

- 3 Edit the radius of the sphere named `test_Sphere` by typing the following in the Script Editor:

```
polySphere -edit -radius 1 test_Sphere;
```



Commands can operate on objects other than what is currently selected by specifying an object name as an argument. The name must be the last argument in the command.

- 4 Select the cube and the sphere simultaneously by holding down shift and clicking the sphere.
- 5 Delete the objects by typing the following in the Script Editor

```
delete;
```

NOTE A list of all flags available for a specific command can be displayed in the Script Editor by typing `help command;`. For example, `help sphere;` provides a list of flags for the sphere command.

For a full list of MEL or Python commands and flags, select one of the following from the menu bar:

- Help > MEL Command Reference
 - Help > Python Command Reference
-

Beyond the lesson

In this lesson, you learned how basic MEL scripting commands work in Maya. You can:

- Write basic MEL creation commands to bypass the Maya user interface.
- Use flags at object creation time to specify the attributes for the objects.
- Edit the attributes of objects using a command in conjunction with the edit flag.
- Observe the history created in the Script Editor and then copy and paste it into the Script Editor to create your own scripts.

Before proceeding with the next lesson you may want to review the material presented in this lesson so you are familiar with the concepts and skills. Some suggested tasks you can try on your own include:

- Practice using creation commands through the user interface and observing the resulting Script Editor history.
- Look through the MEL Command Reference to see more common commands.
- Practice copying and pasting commands from the Script Editor history section to the Script Editor input section.

Lesson 2: Saving scripts to the Shelf

Introduction



In this lesson, you set up a three-point lighting system using MEL and save those MEL commands as a button on the Shelf. This lets you quickly set up a three-point lighting system in future scenes. You learn how to:

- Enter MEL commands in place of selecting menu items
- Create a shelf button to run a series of commands

Three-point lighting uses three different kinds of lights:

- A main light to light the subject, cast shadows and produce highlights
- A fill light to decrease the contrast caused by the main light
- A rim light so the object stands out against the background

Three-point lighting is used extensively in film, photography and computer graphics.

Setting up the scene

Here, you create some primitive objects to display lighting and shading.

- 1 Select File > New Scene to create a new scene.
- 2 Select Window > General Editors > Script Editor to open the Script Editor.
- 3 Create a ground plane by typing the following:

```
polyPlane -height 40 -width 40;
```

A plane appears at the origin.

The height and width flags specify the dimensions of the plane.

- 4 Create a NURBS torus by typing the following

```
torus -axis 0 1 0 -heightRatio 0.5;
```

A torus appears at the origin.

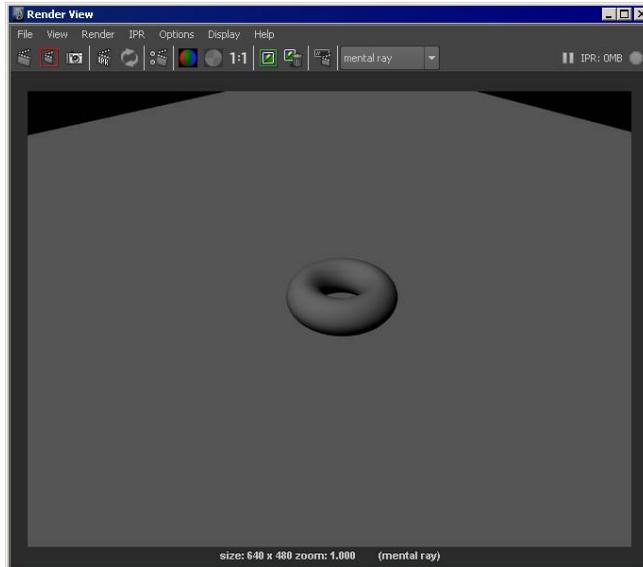
The axis of the torus specifies the orientation of the major axis of the torus. The height ratio flag specifies the ratio of the major radius of the torus to the height of the torus.

- 5 Reposition the torus so it is not intersecting the plane by typing the following with the torus selected.

```
move 0 0.5 0;
```

The move command takes the argument (x, y, z), specifying how much and in what direction to move the selected or named object. This above command moves the torus 0.5 units up on the Y-axis.

- 6 Bring focus to the selected torus in the scene view by pressing ϵ .
- 7 Render the scene by clicking the render button  on the Status Line. The Render View opens with a rendered image of the scene.



- 8 Save the rendered image by clicking the Keep image button . This lets you compare your rendered image with future rendered images. A scroll bar at the bottom of the Render View lets you scroll through your saved rendered images.

In the next steps, you type commands to create three-point lighting in the scene and record them as script history for later use.

Recording the script history

In order to make a shelf button, you need to create Script Editor history to copy to the shelf. In this procedure, you create a three-point lighting system

for the current geometry using MEL, copy the script history, and apply the created three-point lighting to an existing scene.

To create the main lighting

- 1 Clear Script Editor history by clicking the Clear History button  .
- 2 Create a directional light for the main lighting by typing the following:

```
directionalLight -intensity 1;
```

Certain commands require that you refer to the light by name.
- 3 Rename the selected light to “main_light” by typing the following:

```
rename main_light;
```
- 4 Enable the main light to cast shadows by typing the following:

```
setAttr "main_lightShape.useDepthMapShadows" 1;
```

The `setAttr` command allows you to set attributes of a node. In the above command, you are enabling depth map shadows on the main light.
- 5 Adjust the resolution of the shadow maps so that the shadows appear sharper by typing the following:

```
setAttr "main_lightShape.dmapResolution" 2048;
```

In the above command, you are increasing the resolution of the depth map shadows to 2048.
- 6 Position and aim the main light by typing the following:

```
move 0 3 0;  
rotate -30 0 0;
```

Next, you create a rim light for the scene to help differentiate the torus from its background.

To create the rim light

- 1 Create a point light by typing the following:

```
pointLight -intensity 2 -rgb 0.9 0.9 1;
```

The intensity flag has been used to set the brightness of the light. The `-rgb` flag sets the color and takes three values: red, green and blue. The red and green values are less than the blue value, so the light has a blue tint.
- 2 Rename the light by typing the following:

```
rename back_light;
```

- 3 Reposition the light behind the torus by typing the following:

```
move -relative -1.5 1 -1.5;
```

Next, you create a fill light for the scene to soften the main lighting, fill the shadowed part of the torus and decrease the overall contrast of the lighting in the scene.

To create the fill lighting

- 1 Create a point light with an intensity of 0.5 by typing the following:

```
pointLight -intensity 0.5;
```

- 2 Rename the light by typing the following:

```
rename fill_light;
```

- 3 Reposition the light above the torus in a way so that the light rays from the fill light are approximately 90 degrees to those of the main light by typing the following:

```
move -relative 4 6 2;
```

Compare the rendered images

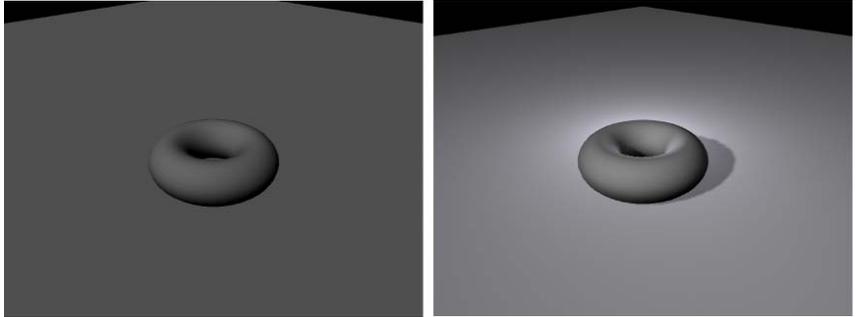
Rendering the image

- 1 Render the scene in the perspective view by selecting the render scene

button  on the Status Line.

- 2 Save the rendered image by clicking the Keep Image button  in the Render View.

- 3 Compare the two images using the scroll bar at the bottom of the window.



The newly rendered image has a shadow. If there was no fill light, detail would be lost in the shadowed area, as the shadow would be black. The rim highlights emphasize the curvature of the torus and help to differentiate the torus from the background.

When creating three-point lighting in a scene with multiple objects, typically there is a main light that provides the main lighting for the scene, and each object in the scene has a linked fill and rim light.

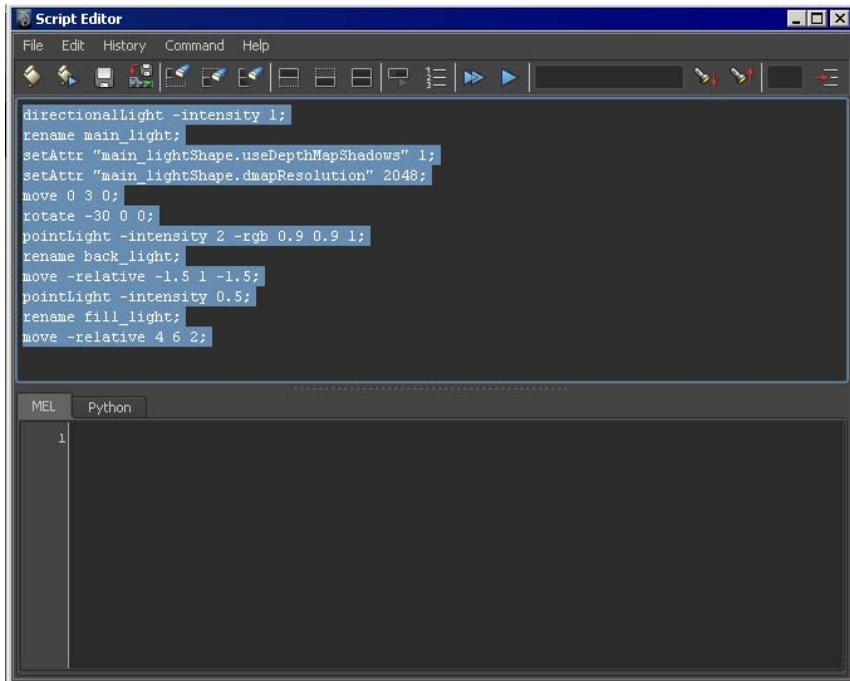
Your three-point lighting script only works for objects of medium scale located at the origin, but once the lights are created, you can tweak them to better suit your scene.

Saving the history as a button

You can save Script Editor history as a Shelf button.

To create the shelf button

- 1 Highlight all the text in the history window by clicking at the beginning of the history in the upper section of the Script Editor and dragging the selection to the end of the Script Editor history section.



- 2 Release the mouse button. The selected text should read:

```
directionalLight -intensity 1; rename main_light; setAttr  
"main_lightShape.useDepthMapShadows" 1; setAttr "main_light  
Shape.dmapResolution" 1024; move 0 3 0; rotate -30 0 0; point  
Light -intensity 2 -rgb .9 .9 1; rename back_light; move -r -1.5  
1 -1.5; pointLight -intensity .5; rename fill_light; move -r 4  
6 2;
```

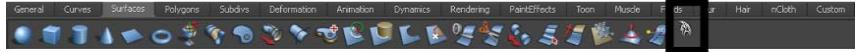
- 3 Middle-drag the selected text to the shelf.

A pop-up window asks you which type of shelf button you are saving.



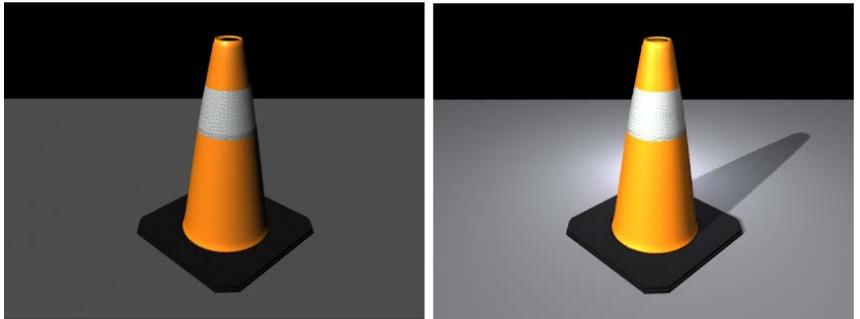
- 4 Click the MEL button on the pop-up window.

A Shelf button is created that executes the scripting commands you dragged to the shelf setting up three-point lighting for a scene.



To test the Shelf button

- 1 Open the scene named `Mel_cone.mb`.
This file can be found in the `GettingStarted` directory that you set as your Maya project:
`GettingStarted/scenes/Mel_cone.mb`
- 2 Render the scene in the perspective view by selecting the render scene button  on the Status Line.
- 3 Save the rendered image by clicking the Keep Image button  in the Render View.
- 4 Click your newly created MEL Shelf button.
Three point lighting is applied to the scene.
- 5 Render the scene again by selecting the render scene button  on the Status Line.
- 6 Compare the two images using the scroll bar at the bottom of the window.



To delete the Shelf button

- Middle-drag the Shelf button you want deleted to the trash can icon on the far right of the Shelf.

NOTE The pointer icon changes to a  icon when you are dragging a Shelf button.

Beyond the Lesson

In this lesson, you learned how to copy history from the Script Editor history section to create Shelf buttons so that you can quickly execute a series of commands. You can:

- Save multiple MEL script commands as a Shelf button.
- Type basic MEL creation commands to bypass the Maya user interface.
- Use additional MEL script commands and MEL script command flags.

Here are some other things you can do with Shelf buttons.

- You can save the custom settings for a menu option box command as a shelf button by setting the parameters in the option box, and then selecting the item from the menu while holding down Shift+Ctrl (Windows) or Shift+Control+Option (Mac OS X).
- You can add custom images to a shelf button, change the name of a shelf button, or change the commands associated with a shelf button by selecting Window > Settings/Preferences > Shelf Editor. For more information, see
- You can also create custom icons for your Shelf buttons in your favorite image editing program. Icons must be 32X32 pixels and in bitmap format.

Lesson 3: Using Variables in MEL

Introduction



In this lesson, you learn how to store data in MEL variables. Variables are a method of storing data in programming, including scripting languages. In this lesson you use the saved value of a variable to create a pyramid of barrels using MEL scripting commands. You also learn how to:

- Use MEL to create copies of existing geometry
- Use variables to store and recall data
- Use the built in MEL math functions to modify the value of variables
- Apply dynamics to scene objects using MEL

Setting up the scene

To import and position the barrel

- 1 Open the scene named `mel_lesson3_barrel.ma`

This file can be found in the `GettingStarted\MEL` directory that you set as your Maya project, in the `scenes` sub-directory.



- 2 Open the Script Editor by selecting Window > General Editors > Script Editor.
- 3 Select the barrel with the select tool.
- 4 In the Script Editor, type the following:
Type exactly as below:

```
rotate -r 0 0 90;
```


The barrel is rotated 90 degrees about the Z axis.

Storing scene information

You will store the value of the measurement annotation as a variable. A variable is a name used for convenience to refer to a certain location in memory. This location in memory can store data.

Variables have four main characteristics; name, type, value and scope. Name, type and scope are defined at creation time. Defining the name and type of a variable is referred to as declaring a variable. In MEL the name of the variable is always prefixed by the “\$” symbol, indicating to the scripting language that the following characters name a variable.

The scope of the variable determines where the variable can be accessed from. If a variable is declared within a block of code, it cannot be accessed from outside the block of code.

Name and type cannot change after creation. The value of a variable can change, but the value must be of the earlier defined type.

NOTE Variables in MEL should always have their type explicitly defined. If not explicitly stated, the scripting language attempts to imply a data type for the variable from the context of the first appearance of the variable. It is good practice to explicitly declare variables as implicitly defining variables can lead to scripting errors and slows down the execution of the script.

To store scene information as a variable

- 1 In the Script Editor, type the following and press enter:

```
float $diameter_barrel;
```

This command declares a variable of data type float, with the name `diameter_barrel`. A floating point value is a number containing a decimal.

- 2 To assign a value to the variable, type the following.

```
$diameter_barrel = 4.2;
```

The diameter of the barrel is 4.2 units. This value from the scene is assigned to the variable. A value is assigned to a variable with the assignment operator (=).

Storing other types of data

MEL also supports other data types that are common to most other programming languages.

To declare a string variable

- Type the following in the Script Editor

```
string $testString = "this is a test";  
print $testString;
```

The following is output to the Script Editor:

```
this is a test
```

The string data type is used to store multiple characters. The `print` command outputs the value of a variable to the Script Editor.

NOTE Variables can be declared and assigned a value in the same statement.

To declare an int variable

- Type the following in the Script Editor

```
int $testInt = 5;
```

```
print ($testInt/2);
```

The following is output to the Script Editor:

```
2
```

The `int` data type is used to store integer values. Integers are positive and negative whole numbers. In programming, when an integer is divided by another integer, the result is an integer, which is why the output here is not 2.5 (2.5 is not an integer).

The `print` command's argument is enclosed in brackets. The `print` command can only take one argument, so brackets are used to tell Maya to evaluate the contents of the brackets before executing the command. Brackets can also be nested in a method similar to mathematical statements to determine the order of evaluation of statements.

In the steps that follow, you create a row of barrels by spacing the barrels evenly. This creates barrels that are just touching, which ensures that there are no intersections during the rigid body simulation.

In the next section, we'll use the `move` command to position the duplicate barrels, using the saved value of the diameter as an argument for the Z value of the command.

Create a row of barrels

Here, we'll use the stored value of the diameter to space the duplicates of the barrel. Variable values can be recalled at any time. Variables can be used as values for arguments.

To duplicate the barrels

- 1 Select the barrel.
- 2 Duplicate the barrel by typing the following:

```
duplicate;
```

You have a duplicate of the first barrel, and it is the currently selected object.

- 3 Align the duplicate barrel with the original by moving the duplicate in Z by a value equal to its diameter by typing the following:

```
move -r 0 0 $diameter_barrel;
```

A duplicate of the original barrel is created and aligned to the first barrel.

- 4 Make another copy of the barrel by typing the following:

```
duplicate;  
move -r 0 0 $diameter_barrel;
```

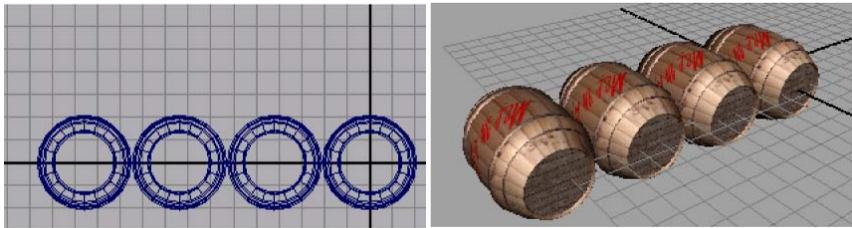
The current selection switches to the newly created geometry.

- 5 Duplicate the currently selected barrel and align the duplicate with the previously created barrel.

```
duplicate;  
move -r 0 0 $diameter_barrel;
```

When duplicating objects, the selection always switches to the duplicate. Each barrel is spaced the same relative to its neighbors.

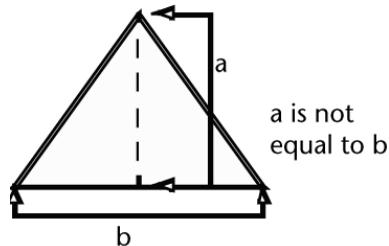
The first row of barrels is complete.



The first row of barrels is complete.

Stacking the row of barrels

Stack the barrels in a triangular arrangement by duplicating the existing row and offsetting the row in both X and Y.

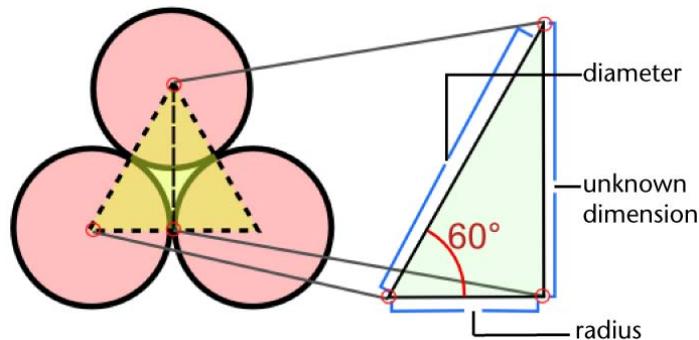


Using basic trigonometry, you can calculate the unknown height offset. MEL has a library of built-in functions that include trigonometry functions, to handle these calculations.

NOTE Trigonometry functions are ratios that relate two sides of a right angled triangle dependant on one of the angles of the triangle. Trigonometry functions are often used for animation, as trigonometric functions are periodic. Periodic functions cycle their output values after a certain period. These functions can easily create cyclical or repeating animation.

Using MEL built-in functions to calculate the Y offset

Sine is the ratio obtained by dividing the side opposite the angle by the hypotenuse. Since the center points of the barrels are spaced along the points of an equilateral triangle, all angles of the triangle are equal to 60° . If you multiply the sine of 60° by the length of one of the sides of the triangle, the result is the height of the triangle.



It is important to know which units of measurement the built-in functions of a programming language use. Many programming languages, including MEL and Python, perform trigonometric operations in radians. You must convert from degrees to radians when using the sine function to calculate the Y offset.

NOTE Radians are a method of measuring sections of a circle. There are 360 degrees in a circle, which is equivalent to 2π radians.

To calculate the offset in Y

- 1 Declare a new variable to store the value of the Y offset by typing the following:

```
float $Y_offset;
```

- 2 Calculate the value of the Y-offset using the sine ratio. You must convert the value of the angle to degrees with the degree to radian conversion function.

```
$Y_offset = $diameter_barrel * (sin(deg_to_rad(60)));
```

The assignment operator evaluates the right hand side of the expression and assigns it to the left hand side. The degree to radian conversion function is the first to be evaluated, as it is within the most interior set of brackets. The sine function calculates the sine of the angle given in radians. The sine value is multiplied by the diameter to calculate the Y offset. The variable is assigned the calculated value.

NOTE The variable `$diameter_barrel` does not change in value. Only the variable on the left side of an assignment operator changes in value.

To stack the rows of barrels

- 1 Select and duplicate all the objects in the scene, by typing the following:

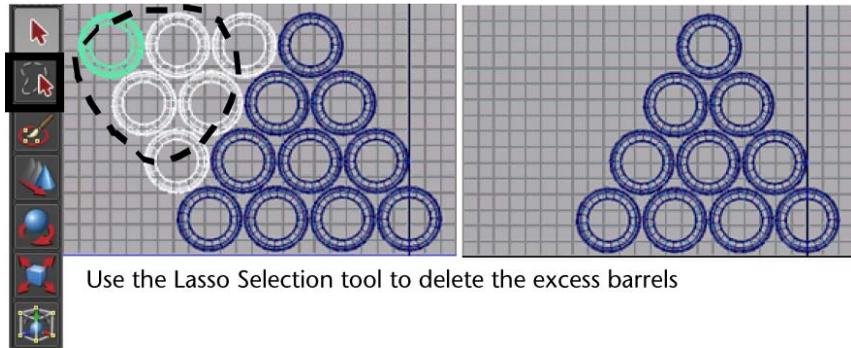
```
select -allDagObjects;  
duplicate;
```

Using the **select** command with the `-allDagObjects` flag selects all DAG (Directed Acyclic Graph) objects, which are a set of objects that exist in the scene such as geometry, IK chains and measure tools. Textures and animation information do not belong to the Directed Acyclic Graph.

- 2 Offset the barrels by the radius of a barrel in X and the calculated `$Y_offset` in Y, divide the diameter by 2 to get the value of the radius. When duplicating the rows, you have created too many barrels.

```
move -r 0 $Y_offset ($diameter_barrel/2.0);  
duplicate;  
move -r 0 $Y_offset ($diameter_barrel/2.0);  
duplicate;  
move -r 0 $Y_offset ($diameter_barrel/2.0);
```

- 3 Using the Lasso tool, select the extra barrels.
- 4 Delete the extra barrels to obtain a pyramid shape.



Creating dynamics with MEL commands

Maya lets you easily add dynamic properties to the objects in your scenes. All attributes that can be modified by selecting menu items, can be accessed by setting the arguments of a Maya command.

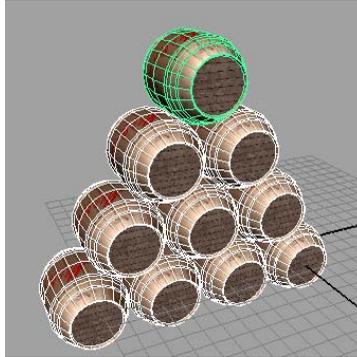
In this section you create a dynamic simulation using MEL so that the stack of barrels reacts to gravity and moves in a realistic manner. Apply a **rigidBody** node to all of the barrels, add a gravity field and a passive rigid body plane for the barrels to collide with.

For more information about rigid bodies and fields within dynamics in Maya, see [Lesson 2: Rigid bodies and constraints](#) on page 554 and the Dynamics guide in the Maya Help.

To add rigid body nodes to the barrels using MEL

- 1 Select all the barrels by typing the following:

```
select -allDagObjects;
```



- 2 Move the stack of barrels upwards along Y (by half the height of a barrel) so they are not intersecting the ground plane.

```
move -r 0 (0.5*$diameter_barrel) 0;
```

- 3 Add a gravity field to the selected geometry by entering the following in the Script Editor.

```
performDynamics 1 Gravity 0;
```

When a gravity field is added to the geometry, rigid body nodes are automatically added to the geometry.

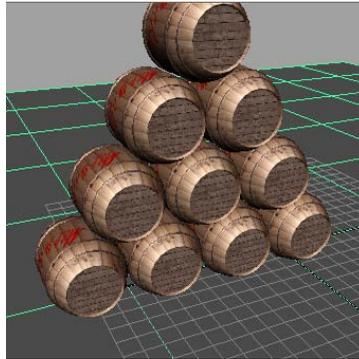
To create a plane for the barrels to simulate against

- 1 Make a plane for the barrels to simulate against by typing the following:

```
polyPlane -height 100 -width 100;
```

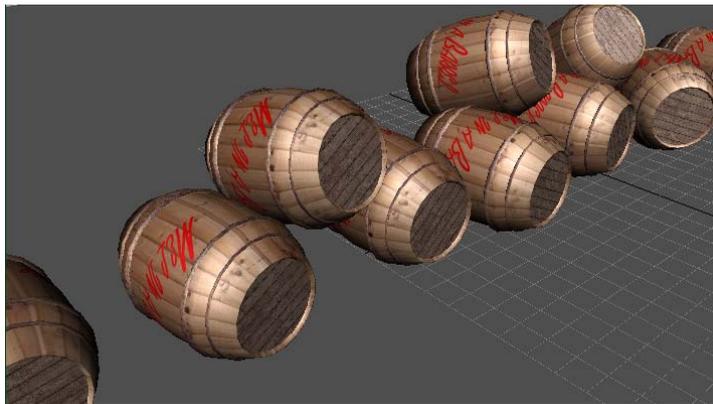
- 2 With the plane selected, add a rigid body node to the plane by typing the following:

```
rigidBody -passive;
```



To view the simulation

- ▶ Click the play button on the time slider to view the dynamics simulation.



Beyond the lesson

In this lesson, you were introduced to how to use variables in MEL with Maya. You can:

- Declare, set and use variables.
- Perform mathematical operations to modify the value of a variable.

- Implement dynamics using MEL commands.

Before proceeding with the next lesson you may want to review the material presented in this lesson so you are familiar with the concepts and skills associated with them. Some suggested tasks you can try on your own include:

- Practice creating, modifying and performing operations with variables. In the next lesson, you use variables extensively to create a user interface.

Lesson 4: User interface creation and procedures

Introduction

In this lesson, you create a user interface for an existing Maya procedure using MEL scripting commands. You learn how to:

- Create a simple window
- Use conditional statements to check the existence of a window
- Use a built-in Maya procedure
- Link a procedure to a user interface

Creating a window

The first step in creating custom user interfaces is to create a window. This window contains the elements of the user interface such as text, buttons and other controls.

To create a window

- 1 Type the following commands into a MEL tab of the Script Editor.

```
window -resizeToFitChildren 1; showWindow;
```

A window appears.



The `-resizeToFitChildren` flag on the window command specifies that the window automatically resizes to fit all of the elements within the window. This flag must have a value: a value of 1 turns on the flag, a value of 0 turns it off.

When you create a window, it is not visible until you use the `showWindow` command. The `showWindow` command without any arguments makes the last window created visible.

- 2 Close the window clicking the X in the upper right hand corner.

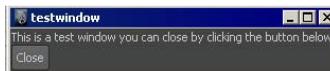
Referencing controls

When creating windows with MEL scripting, you must ensure that every window and user interface control is given a unique name. If you do not explicitly give a window a name, it gets a default name. (In the previous procedure, the name of the window was `window1`, even though you didn't use it.) You cannot reference or modify windows or controls that you don't know the name of.

To refer to a user interface element by name

- 1 Create a window with text and a button that closes it by typing the following in a MEL tab of the Script Editor:

```
window -resizeToFitChildren 1 testwindow; columnLayout; text  
-label "This is a test window you can close by clicking the  
button below."; button -label "Close" -command "deleteUI  
testwindow"; showWindow;
```



- 2 Press the Close button in the window.
The window closes.

Discussion: creating a simple window

In your window, you created elements for your user interface, called controls. All user interface controls must be created within a layout. Layouts instruct Maya how to arrange the user interface elements in the window.

There are many different types of layouts—for example, `formLayout` and `rowLayout`—but the simplest and easiest to use is the `columnLayout`. The `columnLayout` lays out your controls in a column, one underneath the other. For more information on layouts, see *Layouts of the MEL and Expressions* guide.

Within the layout you created a text control, with the `text` command. The `label` flag's argument is the text displayed in the window.

The button is created by the `button` command. The argument of the `command` flag is a MEL command that is executed when the button is pressed. In this case, pressing the button executes the `deleteUI` command, which deletes the window you just created.

Scripts spanning multiple lines

In this script, individual commands span multiple lines. Commands can span multiple lines as long as the end line character is at the end of the command. One convention in MEL is to have the command keyword on one line, and pair the modifier flags and values on separate lines underneath the command, as in this example.

Strings can't be split up over multiple lines without the use of a special character. To split a string over multiple lines, use the backslash (`\`) character. For example:

```
print "With the \\ character, you can \ split a string over \
multiple lines.";
```

To have a backslash appear in a string, you must use a double backslash. The backslash character is an escape character. Escape characters tell the script interpreter that the character that follows is literal. Certain special symbols can be created using the escape character in combination with other characters. For example, to display quotes within a string:

```
print "Jim said \"Hello!\" to the Maya user";
```

Window naming

As user interface scripts get more complex, it is possible that you may make a syntax error. If a user interface script partially executes and halts on an error, the window still exists, but the window is not visible. If you fix the problem and would like to execute the script again, you must either give the window a new name, or delete the hidden window.

Resolving naming conflicts

In the next section, you use conditional statements to resolve window naming conflicts. Conditional statements allow a section of code to execute depending on the value of a variable or the state of an object in the scene.

To resolve conflicting window names

- 1 Create a window with a scripting error inserted before the `showWindow` command by typing the following in a MEL tab of the Script Editor:

```
window testwindow; fake_command; showWindow;
```

An error is output to the Script Editor.

```
// Error: line 2: Cannot find procedure "fake_command;". //
```

The window is not displayed, but still exists.

- 2 Create another window with the same name by typing the following in a MEL tab of the Script Editor:

```
window testwindow; showWindow;
```

An error is output to the Script Editor.

```
// Error: line 1: Object's name is not unique: testwindow //
```

All objects in Maya and user interface elements must have unique names.

To query a window's existence

- 1 Type the following command in the Script Editor input section, to delete the hidden window:

```
deleteUI testwindow;
```

- 2 Check if the window exists by typing the following command:

```
window -exists testwindow;
```

The following result is output to the Script Editor.

```
// Result: 0 //
```

The `exists` flag of the **window** command changes the command so that it either returns a one or zero depending on whether or not the window exists.

3 Type the following in the Script Editor:

```
deleteUI non_existing_window; sphere;
```

An error is output to the Script Editor.

```
// Error: line 1: Object not found: non_existing_window
```

The `sphere` command does not execute as the script has halted on the scripting error.

To be able to delete the existing user interface only if it exists, conditional statements can be used.

To delete a window on condition

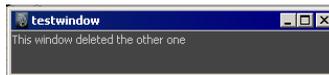
1 Type the following in a MEL tab of the Script Editor:

```
window testwindow; fake_command; showWindow;
```

The window is invisible as there is a scripting error, and the `showWindow` command did not execute.

2 Type the following in a MEL tab of the Script Editor:

```
if (`window -exists testwindow`==1) { deleteUI testwindow; }  
window -resizeToFitChildren 1 testwindow; columnLayout; text  
-label "This window deleted the other one"; showWindow;
```



Two concepts are being introduced here:

- The `if` statement allows a certain part of code to execute depending on a condition. The section of code within the curly braces only executes if the statement inside the parentheses evaluates to true.
- The paired back-ticks within the parentheses (`` ``) indicate that the commands within them are evaluated first. This is a useful part of writing MEL code: you can make certain commands (here, the conditional statement) dependent on other commands (here, the evaluation of whether the window exists), which execute and evaluate first.

Together, the `if` statement and the evaluation mean that only if the result of the window existence query is “1” (the window exists) is the window deleted. Otherwise, the section of code within the curly braces is skipped.

Storing control names

If you refer to a control in your user interface, it must have a name. Instead of manually specifying the name for each user interface element, you can allow the command to generate unique default names and then you can store the name as a variable.

All user interface commands return their full name and path upon creation. You can store the return value as a variable to refer to the control at a later point in the script.

To store a control name as a variable

- 1 Type the following in a MEL tab of the Script Editor:

```
window -resizeToFitChildren 1 pick_me_window; columnLayout;  
$button_one=`button -label "Click Me!" -command "deleteUI  
$button_two`; $button_two=`button -label "NO!, Click Me!!"  
-command "deleteUI $button_one`; $button_close=`button -label  
"Close" -command "deleteUI pick_me_window`; showWindow;
```



We’re using backticks for evaluation again here, though in this case, we’re creating three variables (`$button_one`, `$button_two`, and `$button_close`) that hold the results of calling the button command each time.

- 2 Press one of the “Click Me!” buttons in the window to test the script.
Clicking one of the “Click Me!” buttons deletes the other button by calling the `deleteUI` command with the variable that stores the name of the button control as an argument (`button_one` or `button_two`).
(If you click the button a second time, you’ll get an error as the button you’re trying to delete has already been deleted.)
- 3 Close the window by pressing the Close button in the window.

When you created your user interface window, you assigned it a unique name manually. This is a good practice to ensure that you cannot open multiple copies of a user interface. With more complicated user interfaces,

multiple copies of the same user interface could interfere with each other or slow down Maya.

NOTE You can view the full name and path of a user interface elements by using the print command to output the variable.

```
print $button_one;
```

The following is output to the Script Editor:

```
pick_me_window|columnLayout2|button1
```

This is the full path of the user interface control. You are using variables to store the name and path of controls and therefore you do not have to type the whole name and path to refer to user interface elements.

In this section, you learned how to create, delete, and check for the existence of named windows. In the next section, you'll hook up a more complex window to a procedure.

Introduction to procedures

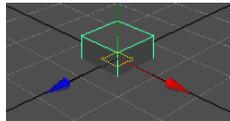
A procedure is a series of instructions that are separate from the main flow of the script. Procedures allow you to reuse certain portions of a script for new applications.

In this section, you use a procedure that is included with Maya (`makeRoll`, which simulates an object rolling on a plane). Later in the lesson, you link this procedure to a user interface to provide an easier way to use the procedure.

To execute the `makeRoll` procedure

- 1 Create a cube and a ground plane by typing the following in a MEL tab of the Script Editor:

```
polyCube -name roll_Cube; polyPlane -height 10 -width 10;
```



- 2 Execute the `makeRoll` procedure by typing the following in a MEL tab of the Script Editor:

```
makeRoll roll_Cube 0 1 1.0;
```

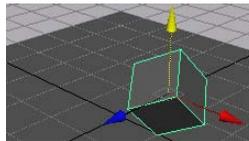
The `makeRoll` procedure requires the following arguments: the name of the object to act on, the height of the ground plane, the type of simulation, and the diameter of the object. The type of simulation is either box or sphere simulation. Rolling simulation is determined using a bounding sphere when simulation type is set to 0, or a bounding box when simulation type is set to 1.

- 3 Set the current frame to two by selecting the current frame box and typing a two.



The expressions set up by the `makeRoll` procedure only take effect when the frame is greater than one.

- 4 Select the cube with the Move tool.
- 5 Move the cube along the plane in X and Z using the Move tool.



The cube rolls on the ground plane.

You have successfully loaded and used the `makeRoll` procedure. Now, you'll link the procedure to a user interface.

Loading a script file

You can read and edit script files other people have created in the Script Editor. In this section, you open a script that contains a completed user interface for the `makeRoll` procedure. However, the user interface is only a layout, and does not contain any functionality.

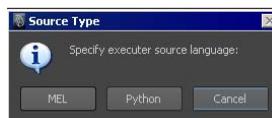
To prepare to open the script

- 1 In the Script Editor, ensure that `Command > Show Line Numbers` is checked.



When Show Line Numbers is on, line numbers appear beside the commands in the Script Editor. The lesson occasionally refers to line numbers to tell you where to make a modification to the script. When line numbers are referred to in the lesson, they are always the line numbers of the original script.

- 2 Create a new tab in the Script Editor by selecting **Command > New Tab**. A pop-up window asks you in which language should the commands entered in this tab execute in.



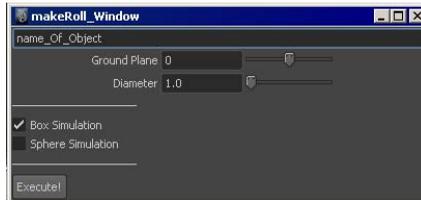
- 3 Click the MEL button on the pop-up window.
You created a second MEL tab in the Script Editor.
- 4 Select **Command > Rename Tab**.
- 5 Enter MEL2.
The tab is renamed.

In the next steps, you load a user interface script into both MEL tabs. The MEL tab will contain the original script, and the MEL2 tab will contain the user interface script with your modifications. This way, line numbers can be referenced from the original script to inform you where to make a modification.

To open a script file in the Script Editor

- 1 Select the MEL tab.
- 2 In the Script Editor, select **File > Load Script**.
- 3 Select the MEL script named `mel_Lesson_4_start.mel`.
This file can be found in the `GettingStarted` directory that you set as your Maya project, within the `mel` sub-directory.
The contents of the MEL file are displayed in the Script Editor.

- 4 Load the same MEL script into the MEL2 tab by repeating the above procedure.
- 5 In the MEL tab, highlight the script by selecting Edit > Select All from the Script Editor menu bar.
- 6 Execute the script by pressing Ctrl+Enter.



A user interface is created for the `makeRoll` procedure. There is a control in the user interface for each argument of the `makeRoll` procedure.

(You can leave this window open as we'll be doing some testing with it, though it doesn't do anything yet.)

Discussion: what the script does

- The controls in the user interface are created by various commands. Below,

the script has been broken up into sections to describe the various user interface controls and their flags. The user interface controls and commands that you used earlier in the lesson are described only briefly below.

Conditional statement

```
if(`window -exists makeRoll_Window`)
{
    deleteUI makeRoll_Window;
}
```

The conditional statement checks if a window with the specified name exists. If the window exists, it deletes it.

Window command

```
window -resizeToFitChildren 1 makeRoll_Window;
```

The `window` command creates a user interface window to contain the controls. For more information, see [Creating a window](#) on page 685.

Column layout

```
columnLayout;
```

The `columnLayout` command creates a layout that arranges the controls within it in a column. For more information, see [Referencing controls](#) on page 686.

Text field commands

```
string $obj_name_text =  
    `textField  
    -editable 0  
    -width 400  
    -text "name_of_Object"`;
```

The command `textField` creates an editable text field. The text field command has multiple flags. The name and path of the text field are stored as a variable. See [Storing control names](#) on page 690.

- The `editable` flag enables or disables editing of the text field.
- The `width` flag sets the width of the control.
- The `text` flag sets the contents of the text field.

Slider commands

```
string $ground_int= `intSliderGrp  
    -minValue -20  
    -maxValue 20  
    -fieldMinValue -20  
    -fieldMaxValue 20  
    -field 1  
    -label "Ground Plane"`;  
string $diameter_float= `floatSliderGrp  
    -value 1.0  
    -minValue 1.0  
    -fieldMinValue 1.0  
    -field 1  
    -label "Diameter"`;
```

The commands `intSliderGrp` and `floatSliderGrp` create sliders. Commands ending in `Grp` create a group of linked controls. The slider commands create controls for a label, a value box and a slider. The slider command has multiple

flags. The name and path of the slider are stored as a variable. See [Storing control names](#) on page 690.

- The `editable` flag can be used to enable or disable editing of the text field.
- The `minValue` and `maxValue` flags set the minimum and maximum values for the slider.
- The `field` flag enables the visibility of an editable value field beside the slider. By default, the field is not visible.
- The `fieldMaxValue` and `fieldMinValue` flags set the minimum and maximum values for the field beside the slider.
- The `label` flag sets the text that appears to the left of the slider.

Separator commands

```
separator -height 20 -width 120;
```

The command `separator` creates a horizontal line. It is used to space the controls vertically in the window. The separator command has multiple flags.

- The `width` flag sets the visible width of the separator.
- The `height` flag sets the amount of vertical spacing created by the separator.

Check box commands

```
string $box_sim_checkbox = `checkBox -value 1 -label "Box Simulation";  
string $sphere_sim_checkbox = `checkBox -value 0 -label "Sphere Simulation";  
separator -height 20 -width 120;  
button  
    -label "Execute!"  
    -command "print (\something\);";
```

The command `checkBox` creates a check box. The check box command has multiple flags. The name and path of the check box are stored as a variable. See [Storing control names](#) on page 690.

- The `value` flag sets the initial value of the check box.

ShowWindow command

```
showWindow;
```

This command enables the last created window's visibility. For more information, see [Creating a window](#) on page 685.

NOTE In the script above, you do not explicitly declare your variables. This is because the return type of user interface creation commands is known to be a string, and cannot be any other data type. Later in the lesson, you will allow a data type to be implied, as you do not know what data type the return will be.

Linking the user interface

Currently the user interface in the window doesn't do anything: moving the sliders and clicking the check boxes has no effect. In this section you learn how to link the user interface to the `makeRoll` procedure you used earlier:

- The `textField` in the user interface will display the currently selected object.
- The check boxes in the window will be linked so that only one check box can be selected at once.
- The button will execute the `makeRoll` procedure with the arguments specified in the window.

In the following steps you modify the user interface script.

A completed version of the user interface script named `mel_lesson_4_finished.mel` is available in the `GettingStarted` project directory, within the `mel` sub-directory.

Displaying the currently selected object

The `makeRoll` procedure does not act upon the currently selected object. You will modify the script so that the `textField` control will display the name of the currently selected object.

To return the name of the currently selected object, use the `ls` command (list) with the `selection`. Since you can select multiple objects in Maya, the return from the `ls` command is not a string, it is a string array, even if only one object is selected. An array is an ordered list of values used to store multiple items of the same data type within one variable. The `makeRoll` procedure cannot accept a string array as an argument, so you must modify the data returned by `ls -selection`.

NOTE For more information on the `ls` command, see `ls` in the MEL Command Reference.

For more information on `arrays`, see `Arrays` in the MEL and Expressions guide.

In this next section, we'll also create a new tempMEL tab so you'll have a place to test commands as you learn them.

To store the names of the currently selected objects as a variable

- 1 Create a new MEL tab (Command > New Tab in the Script Editor).
- 2 Rename it `tempMEL` (Command > Rename Tab in the Script Editor)
- 3 Type the following in the tempMEL tab:

```
select -allDagObjects;
```

This command selects all scene objects.

- 4 Type the following in the tempMEL tab:

```
$all_selected_objects = `ls -selection`;
```

This command lists currently-selected objects and outputs their names to a variable.

- 5 Type the following in the tempMEL tab:

```
print $all_selected_objects;
```

This command outputs the list of stored objects to the Script Editor.

- 6 Execute the commands in the tempMEL tab.

The following is output to the Script Editor:

```
roll_Cube pPlane1
```

The variable `$all_selected_objects` is an array of strings. That is, it stores multiple strings within the variable. Values in an array are called elements.

NOTE Arrays are used frequently in programming to manage large sets of data. If arrays were not used to store large sets of data, you would need to create a variable for each element of data, making scripts hard to maintain and taking up lots of memory. For more information on arrays, see `Arrays` in the MEL and Expressions guide.

In this example, you only want to apply the `roll_Cube` procedure to one object in the scene. You can refer to individual elements of an array by using square brackets (`[]`).

To reduce the selection to one object

- 1 Add the following commands to the top of your user interface script in the MEL2 tab:

```
select -allDagObjects;  
string $all_selected_objects[]=`ls -selection`;
```

- 2 In the MEL2 tab, change the argument of the `text` flag located on line 14 of the original script (you can use the MEL tab for reference) from `"name_of_object"` to the following:

```
$all_selected_objects[0]
```

The command now reads:

```
$obj_name_text = `textField -editable 0 -width 400 -text  
$all_selected_objects[0]`;
```

Elements of an array are accessed using square brackets containing the index number of the element you want to extract from the array. The index number of arrays start counting from zero.

The text field now displays the first object created in the scene.

Linking the check boxes

Check boxes can also use command flags, similar to a button. A check box has more functionality than a button, so the command flags are different. Check boxes have three possible command flags; `changeCommand`, `onCommand`, and `offCommand`:

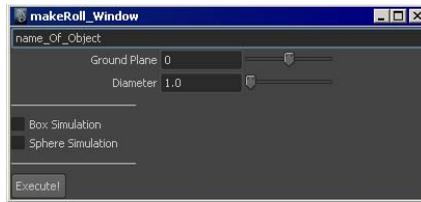
- The `changeCommand` flag triggers whenever the state of the check box changes.

You can use the `checkBox` command with an `edit` flag to change the state of a check box in the user interface.

To change the values of elements in the user interface

- 1 In the tempMEL tab, change the state of the make roll window's check box by executing the following:

```
checkBox -edit -value 0 $box_sim_checkbox;
```



The Box Simulation check box, which was turned on, is now turned off. You can repeat this command with a value of 1 to turn the check box back on.

This shows the use of the `edit` flag with a command as an argument to a command flag, which allows you to change values.

We'll now use the `changeCommand` flag to keep the two check boxes in sync, so that when one check box has its state changed, the other check box is set to the opposite state.

- 2 Add the following under the existing check box declarations (string `$box_sim_checkbox` and string `$sphere_sim_checkbox`) in the MEL2 tab. This will be approximately lines 34-35 depending on whether you added lines at the beginning of the file.

```
checkbox -edit -changeCommand ("checkbox -edit -value (!#1) "+
    $sphere_sim_checkbox)
    $box_sim_checkbox;
```

This block indicates that if the `changeCommand` flag is triggered on the “Box Simulation” checkbox, Maya edits the “Sphere Simulation” checkbox to the opposite of its current value (thus if you turn “Box Simulation” off, Maya turns “Sphere Simulation” on automatically).

In MEL, `#1` is a stand-in character for the value of the checkbox. When Maya runs this line of code, it replaces `#1` with the value of the checkbox. In programming, the symbol `!` means “not.” So in the above code, Maya is changing the “Sphere Simulation” checkbox to whatever value (`#1`) it currently is not (`!`).

Finally, add the following.

```
checkbox -edit -changeCommand ("checkbox -edit -value (!#1) "+
    $box_sim_checkbox)
    $sphere_sim_checkbox;
```

This is the same as the previous lines except it changes the “Box Simulation” checkbox when “Sphere Simulation” is changed.

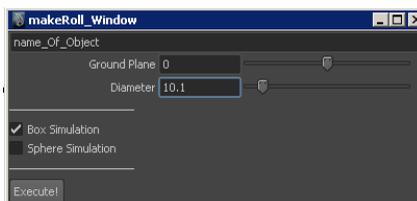
Executing the procedure

The `makeRoll` procedure requires arguments to execute. To link it to the window, you must get the values for the arguments from the controls in the user interface, and execute the `makeRoll` procedure with these arguments.

To get the values of the controls, you must use the query flag. The query flag allows commands to return values of attributes in the scene.

To query a value in the user interface

- 1 Change the value of the diameter slider to 10.1 by dragging the slider to the right, or by typing a value in the diameter slider's text field.



- 2 Print the value of the diameter slider by typing the following in the `tempMEL` tab:

```
print ("The diameter is: " + `floatSliderGrp -query -value $diameter_float`);
```

The following is output to the Script Editor

```
The diameter is: 10.1
```

The query flag allows you to output a value to the Script Editor.

We will add a new procedure in our script that uses the `makeRoll` command with arguments values that are queried from the user interface. This procedure will then be used by the Execute button.

To create a procedure for the Execute button

- 1 Add the following to the top of the `MEL2` tab:

```
global proc makeRoll_callback (string $obj_name_text, string
$ground_int, string $box_sim_checkbox, string $diameter_float)
{
makeRoll
`textField -query -text $obj_name_text`
`intSliderGrp -query -value $ground_int`
`checkbox -query -value $box_sim_checkbox`
```

```
    `floatSliderGrp -query -value $diameter_float`;
}
```

- 2 Every argument of the user interface is defined by querying the user interface elements.

Now you can set the Execute button to call the procedure you just created.

To link the Execute command to a procedure

- In the MEL2 tab, replace the `-command "print (\\"something\");";` line with the following.

```
-command ("makeRoll_callback "+$obj_name_text+" "+$ground_int+"
"+$box_sim_checkbox+" "+$diameter_float);
```

The parameters `$obj_name_text`, `ground_int`, `box_sim_checkbox`, and `diameter_float`, provide the procedure with the values it needs to execute the `makeRoll` script.

Your user interface is now fully functional.

A completed version of the user interface script named `mel_Lesson_4_finished.mel` is available in the `GettingStarted` project directory (`GettingStarted/mel`).

Saving the script

To use your script in other scenes, you must save it. To execute the MEL commands in a script file without having to open it every time you want to use it, you must use global procedures.

Once you've saved a MEL script file that contains a global procedure in the default script folder of Maya, you can access it whenever you type MEL commands. Other procedures in the script file can only be called by the global procedure(s) defined in the script file or other local procedures.

Maya automatically sources procedures in its default scripts folders at start up. A script in the default Maya script folder has the same name as a global procedure defined in the script file, when you type the name of the script file, the global procedure within the script file will be executed. This allows you to create scripts that operate similar to built-in Maya commands.

- 1 Type the following in the MEL2 tab, one line above `select -allDagObjects:`

```
global proc makeRoll_create_UI() {
```

- 2 At the bottom of MEL2, type the following:

```
}
```

Your entire user interface script should be between the curly braces.

- 3 Highlight the entire script by selecting Edit > Select All from the Script Editor menu bar.
- 4 With the script highlighted, save the script by selecting File > Save Script from the Script Editor.
The Script Editor only saves the highlighted commands when saving a script.
- 5 Save the script as `makeRoll_create_UI.mel` in the `GettingStarted/mel` folder.

NOTE When saving a script, you must have the `.mel` extension appended to the script file name in the Save Script dialogue box.

Using the saved script file

You can apply the `makeRoll` procedure to any geometry quickly by selecting the object to apply `makeRoll` to, and executing the `makeRoll_create_UI` global procedure. Before using a script in your scene, you must source it. Sourcing executes all the commands within a script file.

You can only access global procedures declared in the script file from the Script Editor. Any local procedures cannot be accessed from the Script Editor, and must be called from global procedure within the script file.

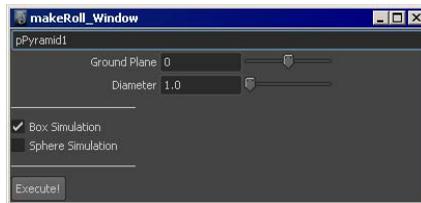
NOTE Global procedures only have access to local procedures declared before them in a script file.

To source and use a script file

NOTE If you have not restarted Maya since you saved your script, you need to rehash before you can source your script. For more information about the `rehash` command, see **rehash** in the Commands documentation.

- 1 Create a new scene.

- 2 In the Script Editor, select File > Source Script.
- 3 In the Source Script dialogue box that opens, select `makeRoll_create_UI.mel`
- 4 Create geometry to test the `makeRoll` procedure on by creating a new scene and typing the following:
`polyPyramid;`
- 5 Execute the `makeRoll_create_UI` global procedure to create the user interface by typing the following in the Script Editor:
`makeRoll_create_UI;`



The user interface you created opens.

- 6 Set the frame to greater than one to view the effects of the `makeRoll` procedure.
- 7 Set the following values in the `makeRoll` user interface:
 - Ground Plane = 0
 - Diameter = 1.0
 - Box Simulation = checked
 - Sphere Simulation = unchecked
- 8 Click the “Execute” button to setup the `makeRoll` expressions.
- 9 Move the pyramid in X and Z to view the results of the `makeRoll` procedure.

Explore different results that occur when you apply the `makeRoll` procedure with different user interface control values. Since you cannot undo the `makeRoll` procedure, you will have to create new geometry every time to apply the `makeRoll` procedure.

When you open a new Maya session, the `makeRoll` user interface procedure is no longer defined and must be sourced to use it.

Beyond the lesson

In this lesson, you learned how to create a user interface from an existing MEL procedure. You can:

- Create a window
- Create user interface controls in a window
- Use conditional statements to check the existence of a window
- Use a built-in Maya procedure
- Modify a user interface to add functionality
- Access elements of an array
- Query and edit user interface values
- Save and source a script

You can create fully customized user interfaces with MEL scripting. There are additional user interface elements not covered in this lesson.

For a full list of user interface elements, look in the MEL Command Reference. You can open the MEL command reference by selecting Help > MEL Command Reference from the menu bar. User interface commands are in the Windows category of the MEL Command Reference.

Lesson 5: Using Python in Maya

Introduction

Python scripting in Maya is very similar to MEL scripting. Python scripting has the same access MEL does to Maya functionality with very few limitations (see Current limitations of the Python guide). Python requires a different syntax than MEL, so commands in Python can look very different from their MEL equivalents.

This tutorial is for migrating your MEL scripting experience to a Python scripting environment in Maya. This lesson does not teach Python, only how to use Python scripting in Maya. Maya uses Python 2.5 on all supported platforms.

Python is a highly readable, easy to maintain, high level programming language. Python is widely used in industry due to its simplicity and standard libraries.

To learn Python, try a Getting Started tutorial in Python:

www.python.org/about/gettingstarted/

In this lesson, you learn how to:

- Enter Python commands in Maya
- Create geometry with Python scripting commands
- Return values from Python commands
- Use flags in Python with the move, scale and rotate commands
- Set the edit and query flags in Python
- Refer to named objects in Python
- Transfer values between Python and MEL
- Call MEL scripts from Python

Entering Python commands

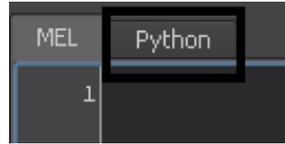
Just like MEL, you can enter Python commands in the Command Line or the Script Editor. The Command Line can only accept single line Python commands, while the Script Editor provides a method to input multiple Python commands and view the results.

The Script Editor displays a running history of executed commands and the results of commands that Maya executes. You can copy most commands from the history section of the Script Editor and paste them in the input section to execute them. However, the results of Python commands are not echoed to the Script Editor exactly like MEL.

For more information, see [To set up the Script Editor](#) on page 658.

To type a Python command in the Script Editor

- 1 Select Window > General Editors > Script Editor.
- 2 Select a Python tab in the Script Editor.



- 3 Type the following command in the input section of the Script Editor and execute it by pressing Enter on the numeric part of your keyboard, or pressing Ctrl+Enter.

```
import maya.cmds as cmds
```

You now have access to Maya commands through the `maya.cmds` module.

NOTE This is how you load modules in Python. For more information about other Python modules, see the online Python documentation.

By convention throughout the documentation and the Python command reference, the `maya.cmds` module is renamed `cmds`, but you can rename the module anything you like.

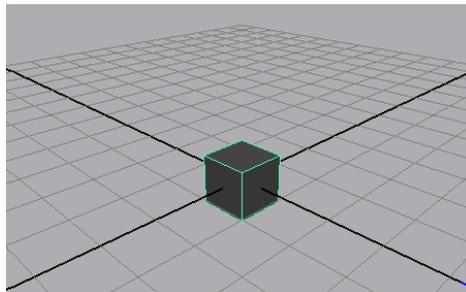
In the Script Editor, the Enter (Windows) or Return (Mac OS X) key above the Shift key does not execute a command. It starts a new line so you can type several commands before executing them.

TIP You can import `maya.cmds` automatically at startup by creating a file called `userSetup.py` in your Maya scripts directory. The `userSetup.py` file can contain any Python commands you want to execute at startup.

For more information, see [Initializing the Maya Environment](#) in [and for Python](#).

- 4 Create a polygonal cube by typing the following:

```
cmds.polyCube ()
```



A polygonal cube is created at the origin. This is equivalent to using the `polyCube` command in MEL.

The command is removed from the input section of the Script Editor after execution. The command and the result of the command is output to the upper section of the Script Editor. The results of Python commands are displayed in the Script Editor between a set of Python commenting characters (`#`).

You can also enter commands in the Command Line.

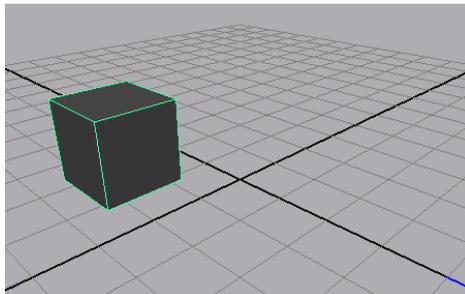
Show the Command Line by selecting `Display > UI Elements`. By default the Command Line is in MEL script entry mode. You can change the mode by clicking on the text in the lower left hand corner. To enter Python commands, the command line must be in Python mode.



To use a Maya command with positional arguments

- 1 Move the polygonal cube that you created with the following Python command:

```
cmds.move(1,2,3)
```

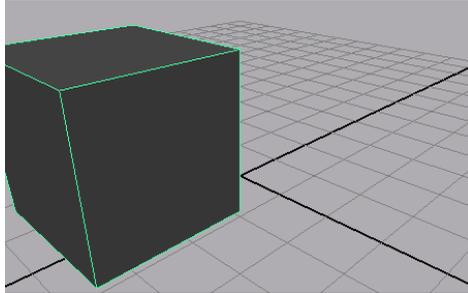


The polygonal cube moves to the XYZ co-ordinates 1,2,3.

The move command accepts XYZ co-ordinates as its commands arguments.

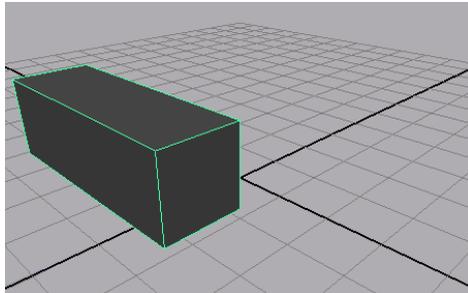
- 2 Scale the polygonal cube with the following command:

```
cmds.scale(2,2,2)
```



- 3 Re-scale the polygonal cube with the following command:

```
cmds.scale(3)
```



When a command is not provided with enough command arguments, the command uses the default arguments. The polygonal cube X scale is set to three, but the Y and Z scale values are set to their default argument one.

- 4 Delete the selected polygonal cube.

```
cmds.delete()
```

Using flags in Python

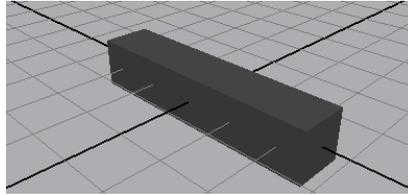
Flags are used with commands to modify their execution. Python implements flags as named arguments. Some flags in MEL do not require values. As all named arguments in Python require a value, these flags are given the value `True` to enable the flag, or `False` to ignore the flag. When setting multiple named arguments, argument-value pairs are separated by commas within the

brackets. Command arguments must appear before named arguments in a command.

To use flags in Python

- 1 Create a polygonal cube with the width defined at creation time by typing the following:

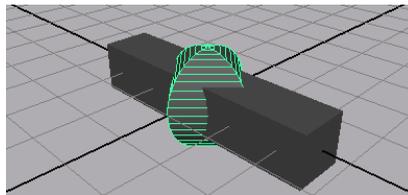
```
cmds.polyCube(width=5)
```



The width flag allows you to set the width of a cube at creation time.

- 2 Create a polygonal sphere with the radius and subdivisions in X and name defined at creation time.

```
cmds.polySphere(radius=1, subdivisionsX=4, name="testSphere")
```



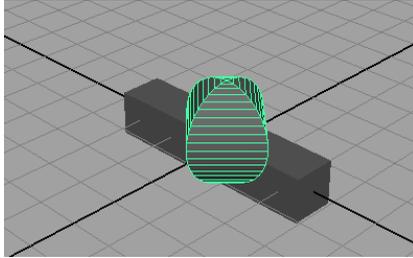
You can use multiple flags to modify the execution of a command by separating the flag-value pairs with a comma. The `radius` flag sets the radius of the created sphere. The `subdivisionsX` flag sets the number of subdivisions in X for the sphere.

NOTE A list of flags available for a command can be output to the Script editor by typing `cmds.help("command")`. For example: `cmds.help("sphere")`

To use flags with command arguments in Python

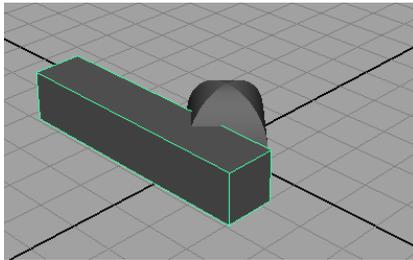
- 1 Move the polygonal sphere by typing the following in the Script Editor:

```
cmds.move(2,2,2)
```



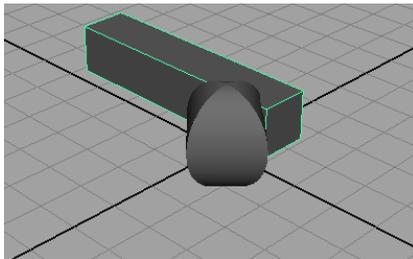
- 2 Select the polygonal cube.
- 3 Move the polygonal cube by typing the following:

```
cmds.move(1,2,3)
```



- 4 Move the polygonal cube relative to its current position by typing the following:

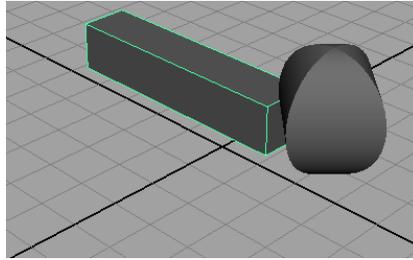
```
cmds.move(-1,0,-2,relative=True)
```



The polygonal cube moves relative to its current position. The relative flag must be placed after the command arguments, as in Python, when passing arguments, named arguments must appear after positional arguments. The command arguments must always be the first arguments of the command.

- 5 Move the polygonal sphere without selecting it by typing the following:

```
cmds.move(3,2,1,"testSphere",relative=True)
```



The name of the object must be passed as a string after the command arguments, but before the flags.

To use flags with multiple arguments in Python

- 1 Create a yellow tinted point light by typing the following:

```
cmds.pointLight(rgb=(1,1,0.5))
```

The point light command creates a point light at the origin. The `rgb` flag specifies the color of the light. The `rgb` flag requires three values; red, green and blue.

When flags require multiple values, the values must be packed within Python's tuple or list data types. Tuples are a method of storing multiple items of homogenous data within a single variable, similar to an array in MEL. The list is similar to the tuple, but supports multiple data types within a single variable. The above command packs the RGB values into a tuple. You could also use a list to achieve a similar effect.

- 2 Move the currently selected light by typing the following:

```
cmds.move(12,15,12)
```

- 3 Create a blue tinted light by typing the following:

```
cmds.pointLight(rgb=[0.2,0.2,1])
```

Square brackets `[]` indicate lists, and parentheses `()` indicate tuples.

The values for the `rgb` flag can be packed into either a list or a tuple and perform the identical operation.

- 4 Move the currently selected light by typing the following:

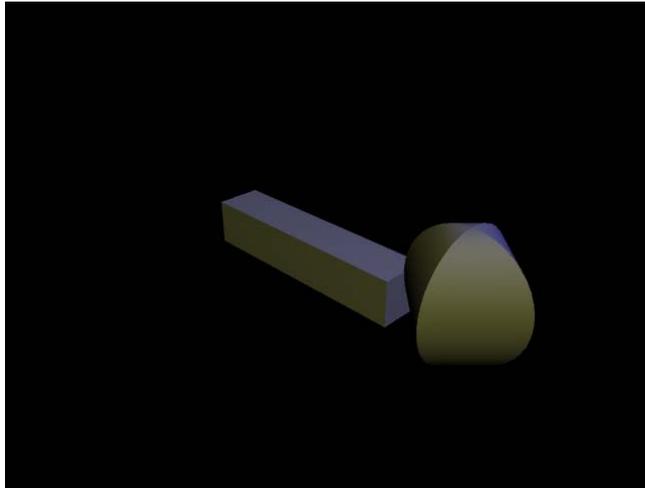
```
cmds.move(12,15,-5)
```

- 5 Select the polygonal cube and polygonal sphere by clicking the polygonal cube, then shift clicking the sphere.

6 Frame the selected objects by pressing the f hotkey on your keyboard.

7 Render the scene by pressing the render current frame button  on the Status line.

The Render View opens and displays the following image.



8 Select all objects in the scene and delete them by typing the following

```
cmds.select(allDagObjects=True)
cmds.delete()
```

Using the Maya command `select` with the `allDagObjects` flag selects objects that exist physically in the scene such as geometry, IK chains and measure tools. These appear on an internal Maya representation called the DAG. For more information, see DAG.

Using the edit flag in Python

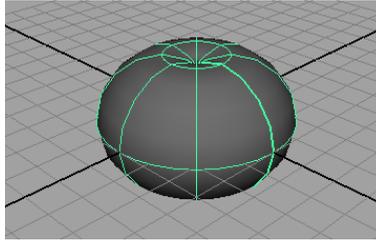
When you perform an action on an object in the Maya user interface, you're using Maya commands to change the attributes of selected objects. You can use the same Maya commands explicitly to edit the attributes of objects in the scene.

Some Maya commands take an edit flag that allows you to make changes to the attributes of an object. The edit flag is used in conjunction with other flags to specify which attributes to change.

To edit the attributes of an existing object

- 1 Create a NURBS torus by typing the following:

```
testVarName=cmds.torus(r=1, axis=(0,1,0) )
```



The axis flag specifies the initial orientation of the torus. Geometry creation commands return the name of the created object, so you can easily refer to the object by storing the name of the object as a variable.

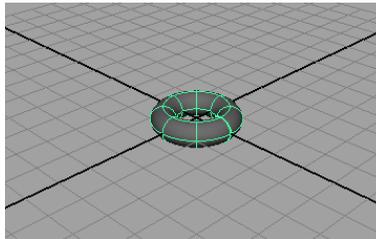
NOTE Most creation commands return the name of the created object as a return value. This value can be stored as a variable to refer to the object at a later time. This is especially useful when referring to controls within a user interface.

- 2 Deselect the torus using a Python command.

```
cmds.select(deselect=True)
```

- 3 Edit the height ratio of the named torus by typing the following:

```
cmds.torus(testVarName,edit=True, hr=0.4)
```



When the edit flag is set to `True`, the command is put in edit mode. When in edit mode, all flags within the brackets of the command now change the attributes of the specified object.

NOTE The edit flag does not need to be the first flag in a command. It is a named argument and can be positioned anywhere within the command after the positional arguments.

Communicating between Python and MEL

MEL and Python in Maya each have built in commands to communicate with each other. MEL and Python communicate by calling commands in the other language and evaluating the results of the last executed command.

Python communicates with MEL using the `eval()` command. Unlike the other Python commands in this lesson, the `eval()` function does not belong to the Maya commands module (`maya.cmds`). The `eval()` function belongs to the `maya.mel` module. The `eval()` function can call MEL scripts or execute MEL commands by sending the commands as a string. Multiple MEL commands can be called in the string by separating the commands with semi-colons. The Python `eval()` function returns the results of the last executed MEL script or command within the `eval` brackets.

MEL communicates with Python using the `python` command. The `python` command accepts a string as its only argument. The string is sent to Python to be evaluated, and the result is returned to MEL. As Python has a more descriptive type system, some results from Python commands returned to MEL have their data type modified. For more information on type conversion, see MEL/Python communication of the Python guide.

To call MEL commands from Python

- 1 Select a Python tab in the Script Editor

- 2 Import the MEL module

```
import maya.mel
```

The `maya.mel` module is a module for evaluating MEL expressions in Python.

- 3 Call a MEL command by typing the following:

```
maya.mel.eval("sphere -radius 3;")
```

A sphere of radius three is created at the origin, just as if you had used the Python command.

- 4 Select a MEL Tab in the Script Editor

- 5 Declare a variable in the MEL tab by typing the following:

```
global float $MyMELVariable=22.7;
```

Only global MEL procedures, variables and scripts are accessible from within Python.

- 6 Select a Python tab.

- 7 Transfer the MEL variable's value to Python by typing the following:

```
TransferMELvar = maya.mel.eval("$temp=$MyMELVariable")
```

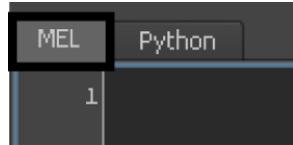
When transferring variables between MEL and Python, the functions return the value of the statement. MEL syntax does not allow you to return the value of a variable by using the variable as a command string. In MEL, when a variable is assigned a value, the value is returned to the Script Editor. Within the `eval()` statement, you assign the value of the global MEL variable to a temporary variable.

- 8 Output the value of the transferred MEL variable using Python commands:

```
print TransferMELvar;
```

To call Python commands from MEL

- 1 Select a MEL tab in the Script Editor.



Commands typed in the MEL tab are executed in MEL.

- 2 Call a Python command by typing the following:

```
python "cmds.sphere()";
```

A sphere is created at the origin, just as if you had typed the command in the Python tab.

- 3 Select a Python Tab in the Script Editor

- 4 Declare a variable in the Python tab by typing the following:

```
MyPythonVariable=22.7
```

- 5 Select a MEL tab in the Script Editor.

- 6 Transfer the Python variable's value to MEL by typing the following:

```
float $TransferVarPy = `python "MyPythonVariable"`;
```

Evaluation back quotes are used in MEL to use the return value of a command in assignment, when the return value is normally returned to the Script Editor history window.

- 7 Output the value of the transferred variable in MEL:

```
print $TransferVarPy;
```

TIP Transferring values between Python and Maya can also be accomplished by using invisible user interface elements. You can edit and query the contents of user interface elements with both scripting languages.

Beyond the lesson

In this lesson, you learned how to access Maya's functionality through Python scripting commands. You can:

- Type basic Python commands in the Script Editor to access Maya Commands.
- Use named arguments as flags at object creation time to specify the attributes for the objects.
- Use commands that require multiple arguments by packing the arguments in a tuple or list.
- Edit the attributes of objects using a command with the edit flag.
- Execute Python commands from the MEL interpreter and execute MEL commands from the Python interpreter.
- Transfer values between Python and MEL.

The MEL tutorials introduce some other uses for scripting in Maya. You can follow through with the MEL tutorials by using the Python Command Reference to translate MEL commands to their Python equivalents.

Python can also be used to write scripted plug-ins. For more information, see Introduction to Maya Python API in the API guide.

Introduction

In Maya, an **asset** is a special type of node that can be used to encapsulate sets of related nodes. A *template* is an external file that specifies a common interface that can be applied to multiple assets (for example, an asset of a car, a truck, and an ambulance all have similar components and can thus have a common interface) and accessed from the Maya UI.

In complex scenes with many nodes and attributes, assets become a key organizational tool by allowing you to package nodes together and expose only the attributes you need. This allows you to keep all the Maya editors clean and clutter-free.

Assets also allow you to customize which attributes are shown to which users while simultaneously locking them out of other attributes. This allows you to ensure the integrity of your scene even when others work on it.

Preparing for the lessons

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

- 1 Select File > 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](#) on page 25.

- 3 As you work through these lessons, remember to save your work frequently and increment the name of the file every so often (`filename1`, `filename2`, and so on).
In this way you'll have earlier versions of your work to return to should the need arise.
- 4 (Optional) Set the current project folder to the Assets tutorial folder by selecting `File > Project > Set`. Then navigate to the `GettingStarted` folder, select the `Assets` folder and click OK.

Lesson 1: Setting up an asset

Introduction

In this lesson, you create an asset representing the provided fire truck model. You then publish attributes from each part and create a template to define future emergency vehicles.

In this lesson you'll be introduced to some of the asset options Maya provides by learning how to:

- Place a polygon object into an asset.
- Publish attributes to the asset.
- Set asset centric selection.
- Create a template for the asset.

Lesson setup

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

- 1 Make sure you've done the steps in `Preparing for the Lesson`.
- 2 Open the scene file named `FireTruck.mb`.

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

```
Getting Started/Assets/Firetruck.mb
```

The scene contains a model of a fire truck split into various pieces such as the body, axles, and doors.

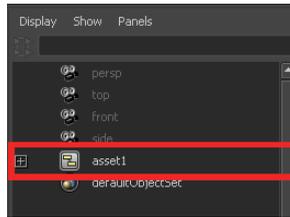


Creating an asset

An asset is the best way to define an object in your scene. An asset is not visible in the main viewport, but you can select it from the Outliner, Hypergraph Connections editor, and Asset Editor.

To create an asset for the fire truck

- 1 Select Body_GRP, Doors_GRP, Axles_GRP, Sirens_GRP, Interior_GRP, and Lights_GRP in the Outliner.
- 2 Select Assets > Create Asset with Transform > .
The Create Asset with Transform options window appears.
- 3 In the Create Asset with Transform options window, select Edit > Reset Settings.
- 4 Under the Include Options, turn on Include these inputs, Include All Inputs, and Include Shading Network.
- 5 Click Apply and Close.
Maya encapsulates the fire truck in an asset.



- 6 Double-click asset1 in the Outliner and change the name to Fire_truck_AST.

Publishing attributes



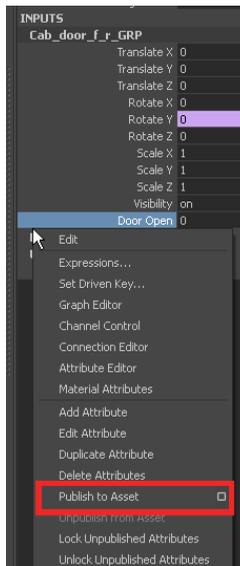
Now that the fire truck is encapsulated, you can publish attributes from the individual pieces to make them available at the asset level.

A published attribute on an asset is bound to an attribute of the object inside. When a user changes the published attribute's value, the value of the corresponding internal attribute is changed as well. You can choose to publish all or only a subset of the available internal attributes, allowing you to limit how much access a user has to the encapsulated object.

The published attributes of an asset are often referred to as the asset's *interface*, and is useful because it allows you access to every important control for your asset in a single place.

To publish the right cab door

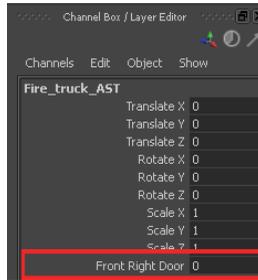
- 1 In the Outliner, expand Fire_truck_AST by clicking  next to its name.
- 2 Expand Doors_GRP.
- 3 Select Cab_door_f_r_GRP and open the Channel Box by clicking  .
- 4 In the Channel Box, under the Cab_door_f_r_GRP heading, select the label Door Open.
- 5 Ctrl + right-click the label and select Publish to Asset >  from the marking-menu.



The Publish Attribute Options window appears.

- 6 In the Publish Attributes Options window, select Edit > Reset Settings.
- 7 In the Attribute name section, select Custom name and then type Front_Right_Door in the Custom name field.
The name Front Right Door represents the published name for this attribute and represents the door's rotation in the fire truck asset's interface.
- 8 Click Apply.

The Front Right Door attribute now appears in the Fire_truck_AST section of the Channel Box.



Now, in the Attribute Editor or Channel Box you can input values into the Front Right Door field to open and close the door. Notice that as you change the value, the value of door_open under Doors_GRP changes to match. This is because your published name (Front_Right_door) is bound to the Rotate Y attribute of Cab_door_f_r_GRP. When one changes, both are updated.

9 Repeat steps 3 - 7 to publish the following:

Node Name	Published name
Cab_door_f_l_GRP	Front_Left_Door
Cab_door_r_r_GRP	Rear_Right_Door
Cab_door_r_l_GRP	Rear_Left_Door

10 Select Edit > Reset Settings.

11 Click Close.

NOTE The Door Open attribute for each door is a custom attribute that controls the Y rotation of each door. Although you could publish the rotate Y attribute of each door directly, these custom attributes are provided so that each door opens as the value increases from 0 to 1. Without these attributes, some doors would open as their rotation increases and others would open as their rotation decreases.

You will learn how to create a custom attribute in the next section.

Publishing multiple attributes to a single published name



Unlike the doors, a fire truck's wheels are expected to move together. For example, the wheels on the front and back axles should spin together, and the front wheels should turn together. To do this, you need to create a custom attribute to control both components at once. Then you can publish your custom attribute to the asset.

To turn the front wheels together

- 1 In the Outliner, expand Axles_GRP, then select Wheels_GRP.

- 2 Open the Attribute Editor by clicking  .

- 3 Make sure the Wheels_GRP tab is currently active, then in the Attribute Editor, select Attributes > Add Attributes.

The Add Attribute window appears.

- 4 Set Long Name to Front_wheel_turn.

- 5 Set Minimum to -30 and Maximum to 30.

- 6 Click Add.

Maya adds the new attribute to Wheels_GRP in the Extra Attributes section of the Attribute Editor. The Add Attribute window stays open.

- 7 Repeat steps 4 to 6 setting Long Name to Wheel_spin. Do not set a Minimum/Maximum value for this attribute.

- 8 Click Close.

Your custom attributes can be found in the Extra Attributes section of the Attribute Editor. However, if you modify their values, nothing happens. This is because you have not connected them to anything yet.

To make the front wheels turn together

- 1 Open the Connection Editor by selecting Windows > General Editors > Connection Editor.
- 2 In the Connection Editor, Wheels_GRP should already be displayed in the Output section. If it is not, select Wheels_GRP in the Outliner and click Reload Left in the Connections Editor.

NOTE If Outputs does not appear over the left column in the Connections Editor, click the to <- from button to reset its position.

- 3 Scroll to the bottom of the left column and select Front_wheel_turn.
- 4 In the Outliner, expand Wheels_GRP and select Wheel_f_l_GRP.
- 5 In the Connection Editor, click Reload Right.
The right column is filled with the left wheel's attributes.
- 6 In the right column, expand rotate and select rotateY.
Maya connects the two attributes.
- 7 In the Outliner, select Wheel_f_r_GRP and repeat steps 5 and 6.

If you adjust the value of Front_wheel_turn, both front wheels turn together.



Next you need to make the front and rear wheels spin in unison.

To make the front and back wheels spin together

- 1 In the Connection Editor, select Wheel_Spin in the left column.
- 2 In the Outliner, select Wheel_f_I_GRP.
- 3 In the Connection Editor, click Reload Right.
- 4 In the right column, expand rotate and select rotateX.
- 5 Repeat steps 2-4 for the other 3 wheels.

Now if you adjust the value of the wheel_spin attribute in the Attribute Editor or Channel Box, all four wheels spin in unison. Although, in some cases, it's useful for the front and back wheels to be able to spin at different rates, the current setup is fine for this tutorial.

Now you can publish your custom attribute to the container. You can do this directly from the Attribute Editor.

To publish the wheel turn and wheel spin

- 1 In the Attribute Editor, right-click Front Wheel Turn and select Publish to 'Fire_truck_AST' from the marking menu.

- 2 Return to the `Wheels_GRP` node, right-click `Wheel Spin`, and select `Publish` to `'Fire_truck_AST'` from the marking menu.

Both attributes are published to the fire truck asset with the same published names as their corresponding custom attributes.

Open the second scene for the lesson

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

- Open the scene file named `FireTruck_published.mb`.
This file can be found in the `Getting Started` directory that you set as your Maya project:
`Getting Started/Assets/Firetruck_published.mb`
The scene contains the same fire truck model but with a complete set of published attributes.

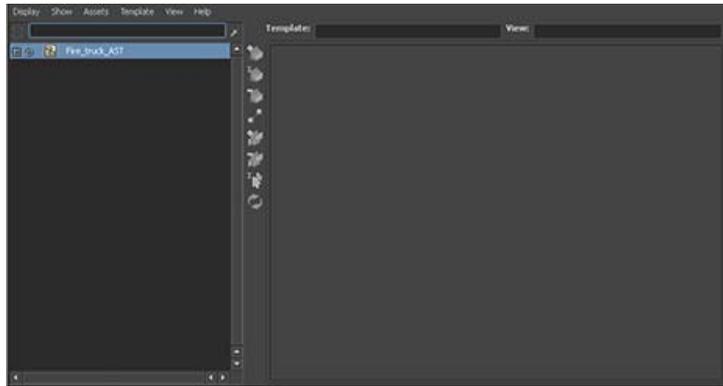
Creating a template

Once you've published all your attributes, you can create a template from your asset. A template defines which published attributes an asset needs to be considered complete. You can then re-purpose this template for use with future fire truck models, or in our case, any emergency vehicle.

To create a template, you first need to open the Asset Editor.

To open the Asset Editor

- 1 In the Outliner, select `Fire_truck_AST`.
- 2 Select `Assets > Asset Editor`.
The Asset Editor appears.



The Asset Editor allows you to manage all the assets and templates in your scene from a single interface. From the Asset Editor you can publish and unpublish attributes, change published names, bind and unbind attributes to published names, and create templates and views.

The Asset Editor is divided into two columns. The left column shows you all the assets in your scene in a format similar to the Outliner. The right column shows you all the published names associated with an asset. In order to see the published names for an asset, you have to pin an asset first.

To pin the fire truck asset

- 1 Select Fire_truck_AST in the left column.

- 2 Click the Pin Asset icon ().

The fire truck's published names appear in the right column.

Now that you've pinned the fire truck asset, you are ready to create a template.

To create a template

- 1 In the Asset Editor, select Template > Save As.

A file browser appears.

- 2 Navigate to the assets folder.

- 3 Set the File name to emergency_vehicle.

- 4 Click Save.

Maya creates a template file.

Notice that once you create a template file, the small arrows next to the published names in the right column change from red to green. This signifies that these published names come from the template and are currently bound.

Creating Views

One advantage to creating a template for your asset is that you can create multiple Views for it. You can use Views to customize which attributes certain users can see in the Maya editors. For example, you can create a View that limits an animator to only the door controls and wheel spin controls and another View to limit a shader to only the truck's body materials and textures.

Not only do Views cut down the amount of editor clutter for each artist, but it also adds a layer of protection so that users don't accidentally change an attribute they shouldn't have control over.

To create a View for an animator

- 1 In the Asset Editor, select View > Add >  .
The Add View Options window appears.
- 2 Select Edit > Reset Settings.
- 3 Set the Name to Animator.
- 4 Set Layout Mode to Flat.
- 5 Click Apply and Close.
Maya creates a new View for the asset.

Notice that even though you created a new View, all the attributes are still visible in the right column. Since we only want the animation controls visible, we need to customize this View.

To customize a View

- 1 Open the Script Editor by clicking  .

NOTE If you prefer, you can also use any external text editing tool to perform the following steps.

- 2 In the Script Editor, select File > Load Script.

A file browser appears.

- 3 Set Files of type to All files.
- 4 Navigate to the `Getting Started/Assets/assets` folder and select `emergency_vehicle.template`.

- 5 Click Open.

The template is now displayed in the Script Editor. The template is an XML file and thus is arranged with a series of tags.

- 6 Scroll to the bottom of the Script Editor until you find the tag

```
<view name='Animator' template='emergency_vehicle'>
```

Underneath this tag, delete the following:

- `<property name='headlight_intensity'/>`
- `<property name='siren_intensity'/>`
- `<property name='interior_light_intensity'/>`

- 7 Highlight the text of the XML file or press Ctrl + A.

- 8 Select File > Save Script.

A file browser appears.

- 9 Set Files of type to All files.

- 10 Navigate to the Assets folder and select `emergency_vehicle.template`.

- 11 Click Save.

- 12 Select Edit > Clear All to clear the Script Editor.

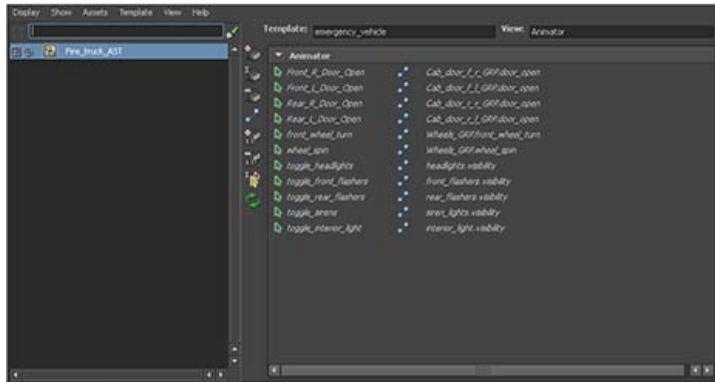
- 13 Close the Script Editor.

You've now customized the View by leaving only the attributes an animator would be interested in. However, this is still not reflected in the Asset Editor. To see your changes, you need to refresh to the Asset Editor.

To update the Asset Editor

- Select Template > Reload.

The right panel updates to show you your customized View.



Now repeat the procedure to create a second view, but with the following changes:

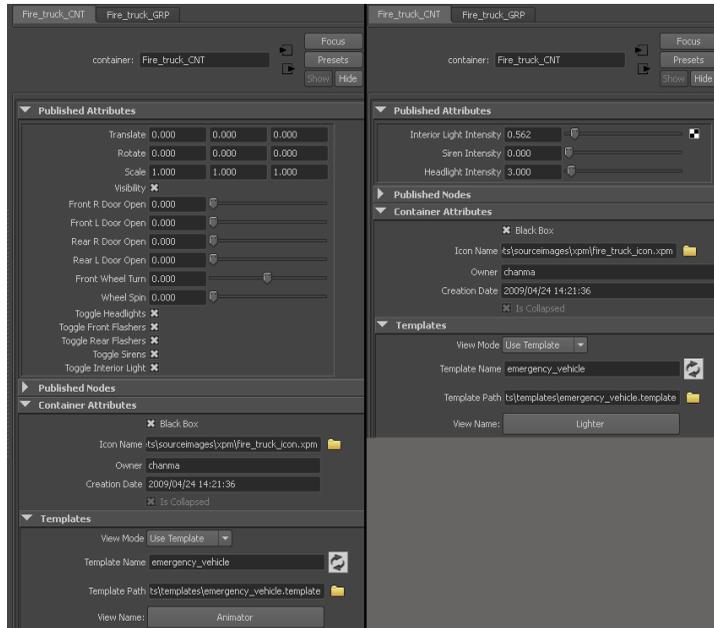
- Set the template Name to Lighter.
- When viewing the template in the Script Editor, scroll down to `<view name='Lighter' template='emergency_vehicle'>`

And remove the following:

- `<property name='translate' />`
- `<property name='rotate' />`
- `<property name='scale' />`
- `<property name='visibility' />`
- `<property name='Front_R_Door_Open' />`
- `<property name='Front_L_Door_Open' />`
- `<property name='Rear_R_Door_Open' />`
- `<property name='Rear_L_Door_Open' />`
- `<property name='front_wheel_turn' />`
- `<property name='wheel_spin' />`
- `<property name='toggle_headlights' />`
- `<property name='toggle_front_flashers' />`
- `<property name='toggle_rear_flashers' />`

- `<property name='toggle_sirens' />`
- `<property name='toggle_interior_light' />`

Now you have two different Views that are useful for two different artists. You can switch between Views in the Asset Editor by selecting View > Name, and then selecting the name of the View you want to set.



Creating a binding set

To ensure that future assets fulfill this template, you can create a binding set. A binding set is a mapping of published names to the internal attributes they represent, and are necessary for performing an Autobind operation.

To create a binding set

- In the Asset Editor, select Template > Create Binding Set. Maya adds a binding set to the template.

You can now close the Asset Editor and save your file.

Assigning a custom icon

Currently, Fire_truck_AST is displayed in the Outliner and Hypergraph with the standard asset icon. You can customize this icon to make the asset more obvious when you place the fire truck in a scene with more than one asset.

To customize the asset icon

- 1 Select Fire_truck_AST in the Outliner.
- 2 In the Attribute Editor, under the Fire_truck_AST tab, open the Asset Attributes section.
This section displays all of the asset's own attributes.
- 3 Click the file icon button next to Icon Name.
A file browser appears.
- 4 Navigate to `Getting Started/Assets/sourceimages/xpm` and select `fire_truck_icon.xpm`.
- 5 Click Open.

The image changes to a small fire truck icon.

Setting Black Box mode

Although Views encourage certain users to only modify certain attributes, it is still possible for a user to inadvertently access and change attributes that they shouldn't. To greatly increase the integrity of your asset, you can set it to Black Box mode.

When Black Box mode is turned on, users can only see the asset itself and all of its published attributes and published nodes. A user cannot open the asset in the Outliner, Hypergraph, or other editors, to view the unpublished attributes and unpublished nodes inside. An additional benefit of this is that it greatly decreases the complexity of scenes in editors since it reduces entire trees of objects to a single node representation.

To turn on Black Box mode

- 1 In the Outliner, select Fire_truck_AST.
- 2 In the Attribute Editor, in the Asset Attributes section, turn on Black Box.

Now if you look at the Outliner, the expand icon next to Fire_truck_AST has disappeared. If you try clicking it, Maya does not allow you to expand it to see its children.



Beyond the lesson

In this lesson you learned how to encapsulate objects into an asset.

- You can create an asset that encapsulates existing objects in the scene.
- You can publish attributes to an asset.
- You can create custom attributes, connect them to objects in the scene, and then publish those custom attributes.
- You can use the Asset Editor to create a template for an existing asset.
- You can create and customize Views of an asset.
- You can assign a custom icon to an asset.
- You can set an asset to Black Box mode to hide its unpublished contents.

Lesson 2: Using assets in a scene

Introduction

In this lesson you first learn how to import existing assets into a scene. You then apply the emergency vehicle template you created in [Lesson 1: Setting up an asset](#) to a police car in the scene and then use the matching published attributes to replace it with your fire truck. You then reference a fire into the scene and finish dressing the scene with a number of external assets.

In this lesson you learn how to:

- Apply an existing template to an asset.
- Bind asset attributes to a template's published names.
- Reference an asset in another file with File Referencing.
- Transfer values from one asset to another.
- Use asset proxies to dress a scene.

Lesson setup

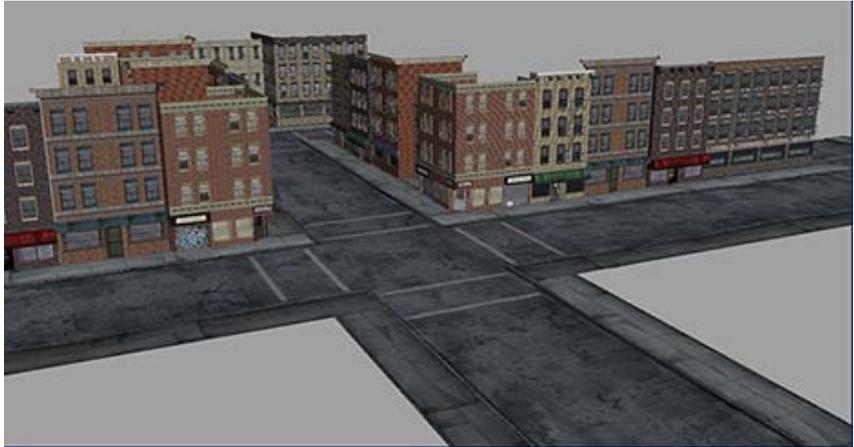
To ensure the lesson works as described, do the following steps before beginning:

- 1 Make sure you've done the steps in [Preparing for the Lesson](#).
- 2 Open the scene file named `EmergencyCity.mb`.

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

```
Getting Started/Assets/EmergencyCity.mb
```

The scene contains a model of a city block and a police car. If you play back the animation, the police car drives along the road, stops outside one of the apartment buildings, and opens its front doors.



Importing assets to dress a scene

If you open the Outliner, you can see that this scene has already been organized into assets (the buildings and the animated police car). Now you can add some lightposts to make the city more believable.

To add lightposts to the scene

- 1 Select File > Import.
A File Browser appears.
- 2 Navigate to the assets directory and select lightpost.ma.
- 3 Click Import.
The lightpost is imported into the scene.
- 4 Select the lightpost and open the Channel Box.
- 5 In the lightpost_AST section, set the translate values to -637, 0.85, -72.
You can now duplicate this lightpost container to set up more lightposts in regular intervals across your scene.



To clean up the Outliner, you can select all the lightpost containers and place them all in a single asset as well.

To create an asset for all the lightposts

- 1 Select all the lightpost asset in the Outliner.
- 2 Select Assets > Create Asset with Transform >  .
The Create Asset with Transform options window appears.
- 3 In the Create Asset with Transform options window, select Edit > Reset Settings.
- 4 Under the Include Options, turn on Include Shading Network.
- 5 Click Apply and Close.
Maya encapsulates all the lightpost assets into a single asset.
- 6 In the Outliner, change the name of the asset to All_lightposts_AST.

Assigning an existing template to an asset

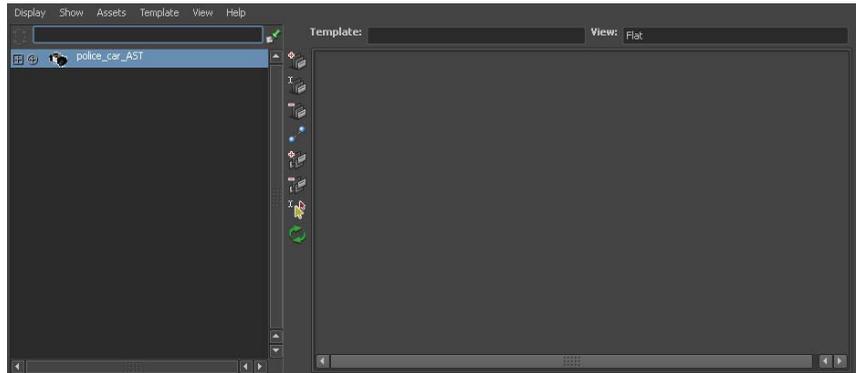
You can use assets to replace the existing police car in the scene with the fire truck model you created in [Lesson 1](#), without losing the existing animation.

First you need to make sure that the published attributes of the police car and fire truck match precisely. To do this, you can reuse the template you created in Lesson 1 from the fire truck to define the police car.

NOTE If you did not start at [Lesson 1](#), both the fire truck model and template can be found in the `Getting Started/Assets/assets` folder.

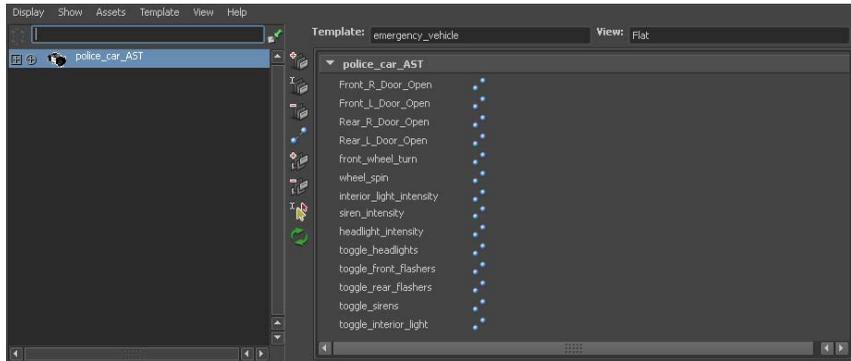
To assign a template to an asset

- 1 Select Assets > Asset Editor.
The Asset Editor appears.
- 2 In the left column of the Asset Editor, select `police_car_AST`.
- 3 Click the Pin Asset icon ().



The police car currently has no published attributes, so the right panel remains empty even when the asset is pinned.

- 4 Select Template > Assign > Assign New Template >  .
The Assign Template Options window appears.
- 5 Turn off Autobind and Create Stand-in Attributes.
- 6 Click Assign Template.
A file browser appears.
- 7 Navigate to the `emergency_vehicle.template` file you created in Lesson 1 and select it.
If you didn't start at [Lesson 1](#), navigate to `Getting Started/Assets/assets/templates` and select the file `emergency_vehicle.template`.
- 8 Click Select.



The published names defined by the template appear in the right panel. However, right now they are unbound to anything in the police car asset because it had no matching published attribute names.

NOTE If the attributes don't appear as shown, you may need to press the

refresh button ().

Autobinding attributes

In order to fulfill the template you've assigned, you need to publish the attributes outlined by the template. Since this template has a binding set which you created earlier (see, [Creating a binding set](#) on page 733), you can choose to autobind this asset. Autobinding an asset automatically publishes the necessary attributes based on the template's binding set.

To autobind an asset

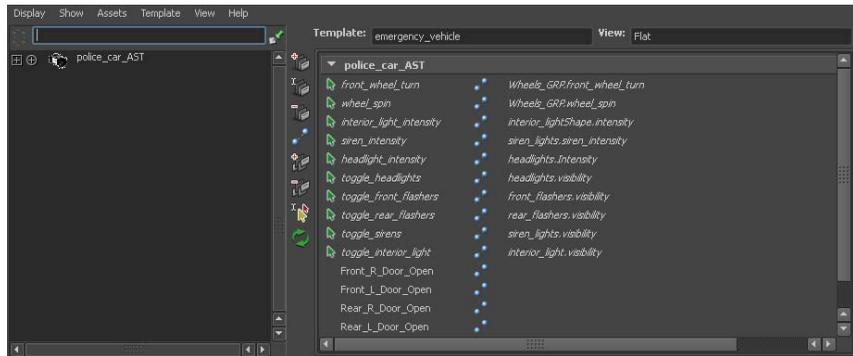
- 1 In the Asset Editor, select **Template > Autobind Asset**.

You may lose control of Maya for a few moments as it processes the template's binding set.

- 2 **View > Mode > Use Template**.

The right panel is split into a number of sections.

The majority of the attribute icons in the police_car_AST section turn green. This signifies that they have been bound to identical published names that were found in the template.



However, there are a set of published names without icons next to them. These represent published names that the template expected, but couldn't find. You need to publish these attributes manually.

Manually binding attributes

In order for your asset to be considered valid, every published name in the `police_car_AST` section must be bound to an attribute of the police car.

Currently, the door attributes are not bound because their internal node names are different (on the fire truck they are called `Cab_door`, while in the police car they are simply called `Door`), so the template is invalid. To fulfill this template, you must bind them manually.

To bind the doors

- 1 In the left panel of the Asset Editor, click the square expand icon () next to `police_car_AST` and again next to `Doors_GRP`.

The contents of the asset appear.

- 2 Click the circular expand icon () next to `Door_f_l_GRP`.

A list of the node's attributes appears.

NOTE If you do not see the circular expand icons (), make sure that `Display > Attributes (Channels)` is turned on in the Asset Editor.

- 3 Scroll down and select the Door Open attribute.

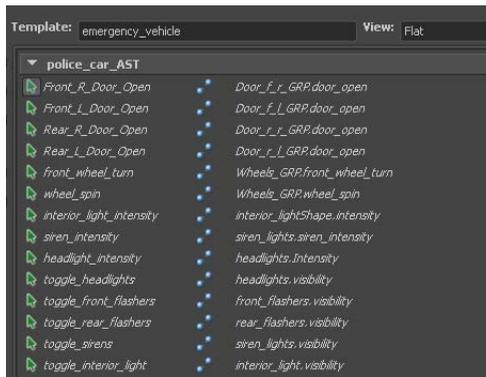
- 4 In the right panel, click the bind icon () next to `Front_L_Door_Open`.

A small green icon appears next to the published name indicating that it is now bound and the bind icon changes to an unbind icon. The name of the attribute it is bound to appears on the right side of the bind icon.

NOTE Binding an attribute automatically publishes it to your asset.

- 5 Repeat steps 2-4 for the other doors, binding them to the appropriate published names in the right panel.

When you are finished all the icons in the police_car_AST section should be green. Your template is now valid.



Swapping assets

Assets with partial or fully identical sets of published names can be swapped while maintaining the values of the matching published attributes. This allows you to quickly re-purpose scenes or swap different assets in and out of your current scene.

To swap the police car for a fire truck

- 1 Select File > Import.
A file browser appears.
- 2 Navigate to the Fire truck file that you saved in [Lesson 1](#).
If you didn't start at Lesson 1, navigate to Getting Started/Assets/assets and select the file firetruck_templated.ma.
Maya imports the fire truck into your scene.



3 In the Outliner, select `police_car_AST` and Ctrl + select `fire_truck_AST`.

4 Select Edit > Transfer Attribute Values > .

The Transfer Attribute Values Options window appears.

5 In the Transfer Attribute Values Options window, select Edit > Reset Settings.

6 Set the following options:

- Turn on Values.
- Turn on In connections and set it to Transfer from source.
- Turn on Out connections.

7 Click Apply and Close.

The fire truck jumps to the police car's position so that they overlap. If you play the animation now, the fire truck follows the police car's previous path and then stops outside the apartment building and opens its doors. The police car remains stationary.



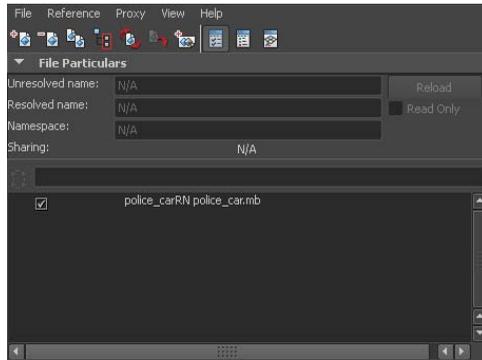
- 8 Delete the police car asset by selecting it from the Outliner and pressing Delete.

Referencing assets in a scene

In the previous section you imported a fire truck asset directly into the scene. However, you can also reference assets from other files. File referencing allows you to represent an object from another file in your scene without actually importing that file.

To reference an asset

- 1 Select File > Reference Editor.
The Reference Editor window appears.
- 2 In the Reference Editor, select File > Create Reference >  .
- 3 Select Edit > Reset Settings.
- 4 Click Reference.
A file browser appears.
- 5 Navigate to `Getting Started/Assets/assets` and select `police_car.mb`.
- 6 Click Reference.



An entry appears in the Reference Editor. This represents the police car. You can turn the reference on and off by clicking the check box next to the reference name.

Now you can move the police car asset to the appropriate spot in your scene. Up until now, you've been using the Outliner to select your containers. However, you can change the selection method so that you can select them in your scene.

To turn on asset centric selection

- 1 Select Window > Settings/Preferences > Preferences.
The Preferences window appears.
- 2 Select the Selection section under Settings.
- 3 Turn on Asset centric selection.
- 4 Click Save.

Now if you click on any part of the police car in the main scene, Maya selects the object's asset.

Now select the Move Tool and move the police car to the edge of one of the streets in the scene. Using the same procedure outlined above, reference the same police car file 3 more times and arrange the police cars around the scene, but not in the path of the fire truck.



Now you need a reason for all the emergency vehicles crowding around.

To reference a fire asset

- 1 In the Reference Editor, select File > Create Reference.
- 2 Click Reference.
A file browser appears.
- 3 Navigate to `Getting Started/Assets/assets` and select `flames.ma`.
- 4 Click Reference.
- 5 Select `Flame_AST` in the Outliner.
- 6 In the Channel Box, set the Translate values for the `Flame_AST` node to `400.581, 84.139, -90.849`.

Now if you play the scene, the apartment burns before the fire truck arrives.



Creating a proxy asset

After adding the dynamic fire effect, you may notice the interactive performance of Maya slow down significantly. You can speed up your scene by using proxy assets.

A proxy is an object that stands in a scene for another, more complex object. Proxy objects allow you to quickly switch between complex and simple representations of an object in your scene. They are useful for temporarily reducing the complexity of the scene as you work with it.

To export a proxy container

- 1 In the scene view, select the fire's fluid asset.
Since asset-centric selection is on, this selects the fire's asset node.
- 2 Select Assets > Export Proxy Asset >  .
The Export Proxy Asset window appears.
- 3 Select Edit > Reset Settings.
- 4 Change the name to Flame_proxy.
- 5 Click Apply and Close.
The File Browser appears.

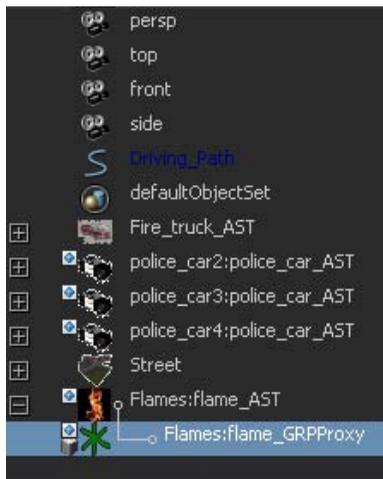
- 6 Navigate to `Getting Started/Assets` and type the name `flame_proxy.ma`.
- 7 Click `Export`.
Maya creates a proxy file.

Now you can load the proxy to represent the asset.

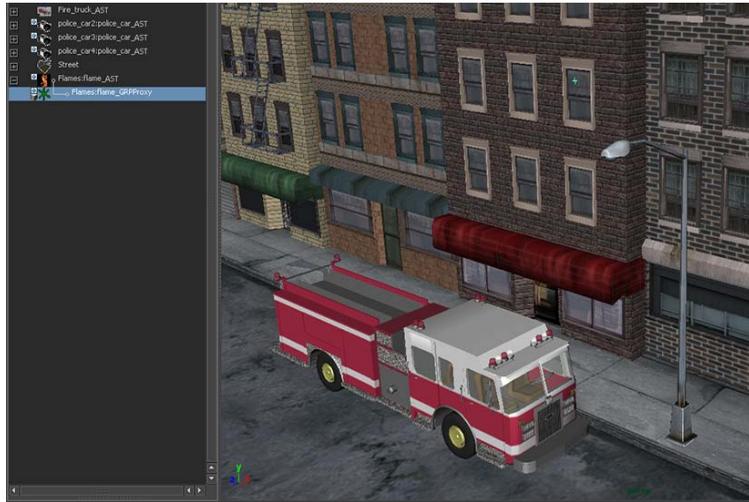
To load a proxy

- 1 In the Reference Editor, select the referenced flame file that appears next to the proxy asset icon ().
- 2 Select `Proxy > Reload Proxy as > proxyAsset (flame_proxy.ma)`.
The fire effect disappears and a locator appears in its place.

The asset for your fire still appears in the Outliner, however if you expand it, only a single locator node appears (instead of the fire's group node).



Likewise, the fire in your scene has been replaced with a locator node. Your interaction with this asset has not changed, you can still assign values to the published attributes and move the locator around the scene. When you reload the proxy asset again, the fire reflects those changes. However, if you play the scene when the proxy asset is active, the scene playback is much faster.



Modifying a proxy asset

Although the locator in the scene gives you the position of your referenced asset, it doesn't accurately convey the size or shape of the object it represents. You can modify the proxy file to give the proxy asset a more useful shape.

To modify the proxy asset

- 1 Save your current scene file.
- 2 Select File > Open Scene.
A file browser appears.
- 3 Navigate to `Getting Started/Assets` and select `flame_proxy.ma`.
- 4 Click Open.

Maya opens the proxy file.

The proxy file consists of a single locator at the same position as the fire in your city scene.

- 5 Select Create > Polygon Primitives > Cube > .

NOTE Make sure Interactive Creation is turned off in the Create > Polygon Primitives menu.

The Create Polygon Cube window appears.

- 6 Select Edit > Reset Settings.
- 7 Set the following attributes:
 - Width: 10.5
 - Height: 16.5
 - Length: 6.5
- 8 Click Create.
A polygon cube appears in the scene.
- 9 Select the cube and open the Channel Box.
- 10 Set Translate to 400.581, 84.139, -90.849.
- 11 Set Rotate X to 15.
- 12 Open the Outliner (Window > Outliner).
- 13 Expand flame_AST.
- 14 Select pCube1 and middle-mouse drag it onto the locator node (Flame_GRPProxy) inside flame_AST.
Performing this step allows you to move the cube and locator together.



You can add vertices and model the cube to shape it more like fire if you choose. You can find polygon modeling tips in [Polygonal Modeling](#) on page 73.

Now save this file and return to your original scene file. The proxy that was once a single locator has now changed to a cube that is roughly the size of the referenced fire effect.



You can switch back to the the fluid effect by right-clicking the proxy and selecting Reload Proxy As > original from the marking menu.

Dressing the rest of the scene with assets

To finish dressing the scene you can import (or reference) a number of other assets located in the Assets directory. For example, you could:

- Import barricade.mb to block roads and give the fire truck a clear path.
- Import traffilight.mb and place a copy over each intersection.
- Import park.mb to add a public park.
- Import trashbag.mb and trashcan_A.ma/trashcan_B.mb to make the city dirtier.
- Import a few more police cars to create a police barricade.
- Import additional instances of flames.mb to make all the windows in the building burn.

Notice that assets can range from very simple objects (trashbags, trashcans), to collections of objects (the park), to complex objects with moving parts (the police car), yet all of them are treated as single black box asset nodes in the

scene each with a single interface. You could take this further by placing all similar objects (all trash bags and trash cans for example) into a single asset.



You can now batch render the scene to see your finished product.

To batch render the scene

- 1 If you haven't done so already, set all proxies to their original representations by right-clicking them and selecting **Reload Proxy As > original** from the marking menu.
- 2 Position the camera in a spot where you can see the majority of the action.
- 3 Click the **Render Settings** button ().
The **Render Settings** window appears.
- 4 Set the following:
 - **Image format:** Maya IFF (iff)
 - **Frame/Animation ext:** name.#.ext
 - **Start frame:** 0
 - **End frame:** 170
- 5 Click **Close**.
- 6 In the **Rendering** menu set, select **Render > Batch Render**.

Maya batch renders the scene. This may take a few minutes. You can monitor the progress in the Status Line.

When the batch render is complete, you can view the sequence by selecting File > View Sequence, and then opening the first file of the sequence (it has an extension of .001). Maya then plays back the sequence of rendered images showing you the final animation.



Beyond the lesson

In this lesson you learned how to incorporate assets into a scene.

- You can use an existing template to define an asset.
- You can autobind attributes to publish them according to the template.
- You can manually bind attributes to an asset.
- You can swap one asset for another.
- You can reference assets into a scene and instance them multiple times.

- You can create proxy assets to stand in for referenced assets.
- You can import additional assets to dress the scene.

Introduction

File referencing allows you to populate a scene with objects originating from external files, without actually importing those files into your scene. This allows you to keep scenes separated into pieces which can be modified independent of the main scene.

File referencing is especially useful when working with large scenes since it allows multiple artists to work on various pieces of the scene simultaneously.

File Referencing also allows you to set up proxies to stand-in for particularly dense or system intensive objects to improve playback performance. You can also export edits you've made to file references so that you can apply them to other references later in the pipeline.

Preparing for the lessons

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

- 1 Select File > New Scene.
- 2 If you have not already done so, copy the Getting Started folder from its installation location to your projects directory. Then, set the Getting Started directory as your Maya project. For more information, see [Copying and setting the Maya project](#) on page 25.
- 3 As you work through these lessons, remember to save your work frequently and increment the name of the file every so often (filename1, filename2, and so on).

In this way you'll have earlier versions of your work to return to should the need arise.

- 4 Set the current project folder to the File Referencing tutorial folder by selecting File > Project > Set. Then navigate to the Getting Started folder, select the File Referencing folder and click OK.

Lesson 1: Using file references

Introduction

In this lesson, you create multiple file references to a shuffling zombie asset for use in your graveyard scene. You then create proxies for these zombies to speed up the scene. You will also use the export edits feature to tweak one zombie's animation and apply it to others, as well as apply an entirely different walk animation obtained from a completely different file.

In this lesson, you'll learn how to:

- Create a reference
- Create a proxy reference
- List reference edits
- Export reference edits
- Apply reference edits

Lesson setup

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

- 1 Make sure you've done the steps in [Preparing for the lessons](#) on page 755.
- 2 (optional) Set your project to the File Referencing folder by selecting File > Project > Set, and selecting the following directory in your Getting Started folder:
`Getting Started/File Referencing`
- 3 Open the scene file named graveyard.mb.

If you set your project in step 2, the file appears in the default folder. If you didn't perform step 2, this file can be found in the Getting Started directory that you set as your Maya project:

Getting Started/File Referencing/graveyard.mb

The scene contains a graveyard.



Creating a file reference

A file reference is a set of objects that appear in a scene, but are not stored in that scene. Instead, the scene refers to external files in which they are stored.

To create a file referenced zombie

- 1 Select File > Create Reference >  .
The Create Reference options window appears.
- 2 In the Shared Reference Options, turn on Shared Nodes, Display layers, and Shading networks.
- 3 Click Reference.
The File Browser appears.
- 4 Select `zombieShuffle.mb` and then click the Reference button.
A new zombie appears in the scene. When you play the animation, the zombie shuffles a short distance.



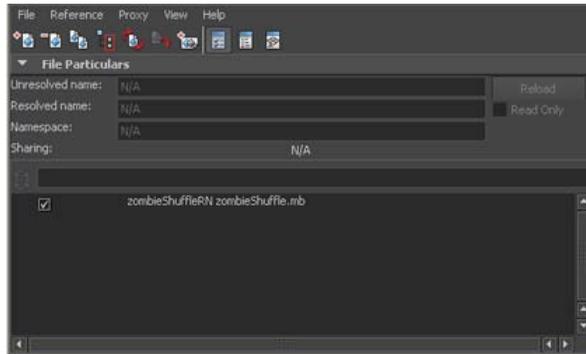
Using the Reference Editor

You can duplicate your referenced zombie repeatedly to dress the scene. To do this, you use the Duplicate Reference option in the Reference Editor.

To duplicate the referenced zombie

- 1 Open the Reference Editor (File > Reference Editor).

The Reference Editor displays all your referenced files in a list. From this editor you can create and remove references, load and unload references, create proxies, and control all other aspects of the referenced files in the scene.



- 2 In the list of referenced files, select `zombieShuffleRN zombieShuffle.mb`.
- 3 Select `Reference > Duplicate Reference`.
A second entry called `zombieShuffleRN1 zombieShuffle.mb` appears in the Reference Editor.

Although another entry appeared in the Reference Editor, there still appears to be one zombie in the scene. This is because the duplicated zombie is overlapping the original.

To reposition the duplicated zombie

- 1 Open the Outliner (`Window > Outliner`).
The last entry in the Outliner is `zombieShuffle1:zombie_AST`. Notice that there is a small icon in the top-left corner of the asset icon, indicating that this is a referenced object. Also note that the prefix `zombieShuffle1:` denotes the namespace for that zombie.
- 2 Select `zombieShuffle1:zombie_AST` in the Outliner and then use the manipulators to reposition the zombie in the scene.



If you play the animation, both zombies perform identical shuffles in different parts of the scene.

Now repeat the above steps 3 or 4 more times to fill the graveyard with a horde of zombies. Leave the original zombie reference in its default place.



Exporting objects in your scene

Just as you can import references from external files into your scene, you can also export objects in your scene to external files and keep them as references.

To export the willow tree

- 1 Open the Outliner (Window > Outliner).
- 2 Select willow1MeshGroup.
- 3 Select File > Export Selection >  .
The Export Selection Options window appears.
- 4 Select Edit > Reset Settings.
- 5 Turn on Keep only a reference.
- 6 Click Export Selection.
The file browser appears.
- 7 Enter the name tree and click Export Selection.
If you check the Reference Editor, you can see a new entry called `treeRN` `tree.ma`. The main scene is now referencing the tree from an external file, just like your zombies.

Creating a proxy for a referenced object

You may notice the interactive performance of your scene is slowing down as it becomes more complex. You can speed the scene up again by replacing some of the more complex objects with simpler proxies.

To create a proxy file for the tree

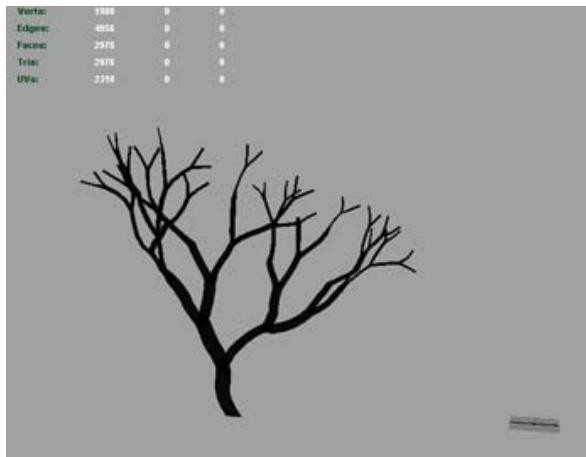
- 1 Select File > Save Scene As and save your current work under the name `graveyard1.ma`.
- 2 Select File > Open Scene and open `tree.ma`.
This is the tree file you exported from the previous step. If you zoom out, you can see the tree sitting in the same spot as in the graveyard scene.
- 3 Select Display > Heads Up Display > Poly Count.

Maya displays information about the scene in the top-left corner of the workspace. You can see that the tree consists of 17136 faces.



- 4 Select the leaves and press delete.

Notice that the number of faces drops to 2976 in the poly count. Since the base of the tree is still the same size and shape, this is makes an ideal proxy.



- 5 Select the tree and open the Attribute Editor.
- 6 In the willow1Main tab, change the transform name to treeProxy.
- 7 Select File > Save Scene As and save the file as treeProxy.ma.

Creating a proxy for a referenced asset

The zombies in your graveyard scene are even more complex than the willow tree. To further improve the performance, you can replace some of the zombies with proxies.

However, unlike the tree, the zombies in your scene are contained in assets. The procedure for creating proxy assets is slightly different from creating a proxy for other objects.

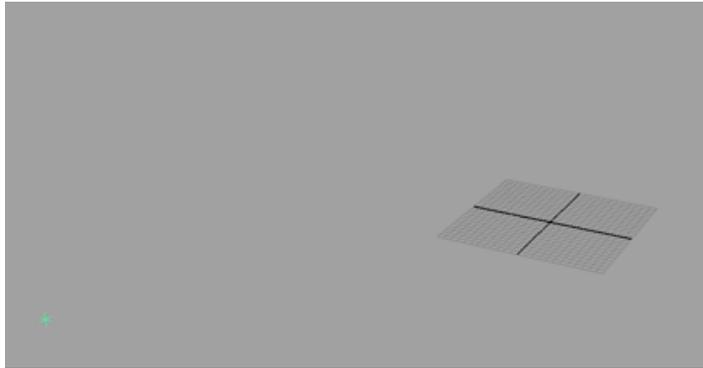
To export an proxy asset

- 1 Re-open your graveyard1.ma scene file (File > Open Scene).
- 2 In the Outliner, select zombieShuffle1:zombie_shuffle_AST.
- 3 Select Assets > Export Proxy Asset >  .
The Export Asset Proxy Options window appears.
- 4 Select Edit > Reset Settings.
- 5 Change the Name to zombieProxy.
- 6 Click Apply and Close.
The File Browser appears.
- 7 Name the proxy zombieProxy1 and click Export.

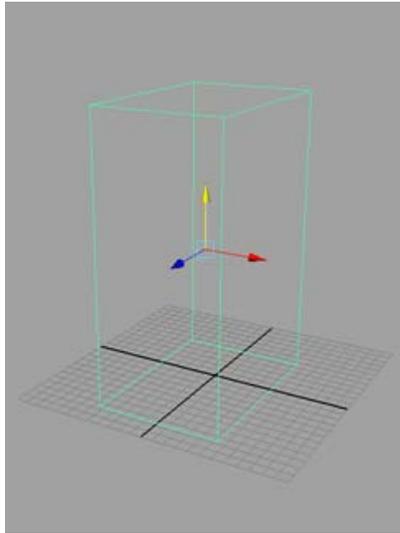
A proxy file is created for the zombie. This file is the base for your proxy.

To create a proxy file for the zombie shuffle

- 1 Save the current scene as graveyard2.ma.
- 2 Select File > Open Scene and open the zombieProxy1 file.
The file contains a single locator node inside an asset named zombie_shuffle_AST. This locator represents the position of the zombie in the graveyard scene.

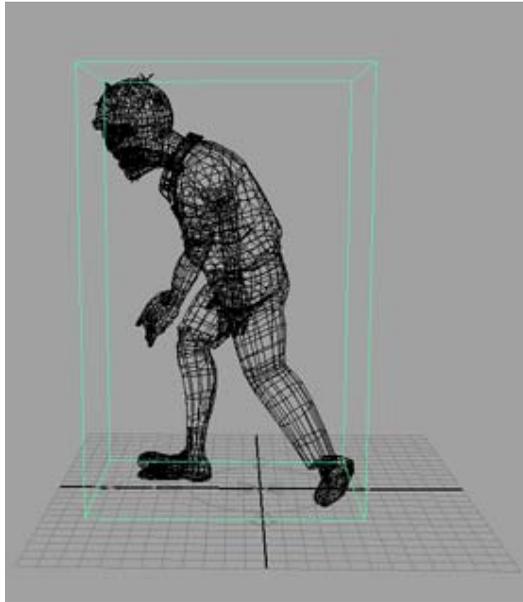


- 3 Since the zombie is far more complex than the tree as well as being animated, you will create a proxy from scratch, as opposed to using the original model as a starting point like the willow tree.
- 4 Select Create > Polygon Primitives and make sure that Interactive Creation is turned off. Then select Cube > .
The Polygon Cube Options window appears.
- 5 Select Edit > Reset Settings.
- 6 Set the cube properties to the following:
 - Width: 10.676
 - Height: 24.042
 - Depth: 15.215
- 7 Click Create.
Maya creates a polygon cube in the scene.
- 8 Open the Channel Box and set the following:
 - Translate Y: 12.044
 - Translate Z: 2.073



- 9 Open the Attribute Editor.
- 10 In the pCube1 tab, change the cube's transform name to `zombie_proxy`.
- 11 In the Pivots section of the Attribute Editor, set the following:
 - World Rotate Pivot: 0.000, 0.000, 2.073
 - World Scale Pivot: 0.000, 0.000, 2.073

This cube will act as a stand-in for zombies in your graveyard scene. Because it is approximately the same size as your zombie, you can use it to accurately predict spacing in your scene.



NOTE The image above is just a comparison. The actual zombie model should not be part of your proxy file.

- 12 Select Modify > Freeze Transformations.

If you want, you can make the proxy more elaborate by modeling it into a shape that is closer to the actual zombie.

Next you need to add animation that approximates the zombie's shuffling animation.

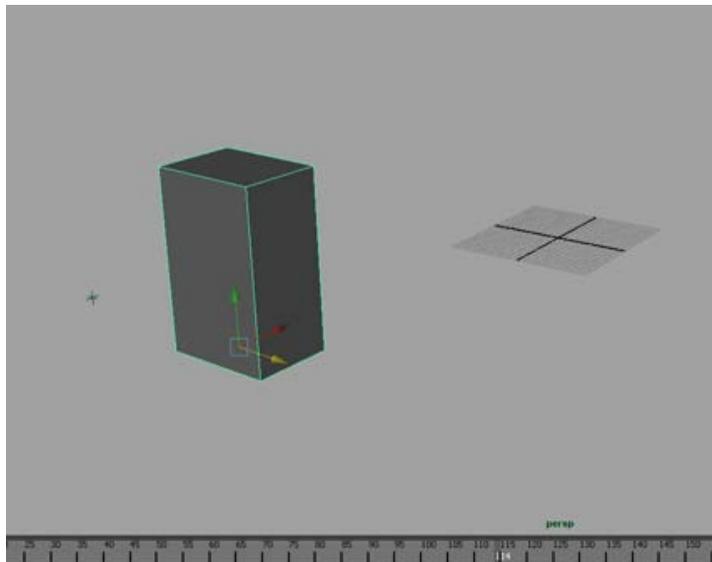
To animate the proxy zombie

- 1 Set the Playback End Time box to 200.



- 2 Make sure the time slider is set to frame 1.
- 3 In the Outliner (Window > Outliner), expand `zombie_shuffle_AST` and select `zombie_controller_GRPProxy`.

- 4 Ctrl + select zombie_proxy and then select Edit > Transfer Attribute Values >  .
The Transfer Attribute Values window appears.
- 5 Select Edit > Reset Settings.
- 6 Click Apply and Close.
The proxy object moves to the same location as the locator. This ensures it will be positioned correctly in the graveyard scene.
- 7 In the Outliner, middle-drag zombie_proxy onto zombie_controller_GRPProxy.
- 8 Select zombie_shuffle_AST press 's'.
Maya sets a keyframe.
- 9 Set the time slider to frame 200.
- 10 In the Attribute Editor, set proxy_zombie's Translate Z to 25.578.
- 11 Right-click the Translate attribute and select Set Key.
Now if you play the scene, the cube moves approximately the same distance and speed as the zombie shuffle animation.



- 12 Select File > Save Scene.

You can repeat these steps for each zombie you want to create a proxy for. Make sure you name the proxies appropriately so you know which zombies to apply them to since they are not interchangeable.

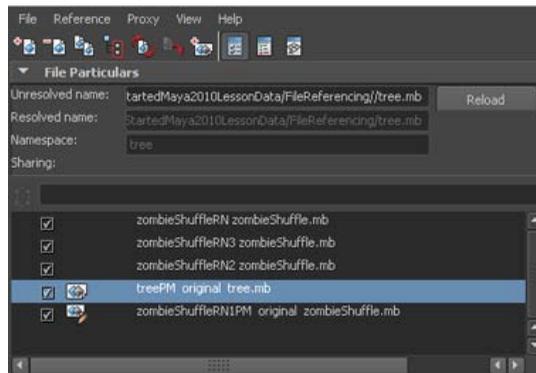
Creating proxy references

Now that you've created proxy files for the willow tree and zombie, you can replace the respective objects in your graveyard scene with those proxies.

To create a reference proxy

- 1 Re-open your graveyard2.ma scene file (File > Open Scene).
- 2 Open the Reference Editor (File > Reference Editor).
- 3 In the Reference Editor, select `treeRN tree.ma`.
- 4 Select Proxy > Add Proxy.
The File Browser appears.
- 5 Select `treeProxy.ma` and click the Proxy button.

A new icon appears next to the reference file and the name changes to `treePM original tree.mb` in the Reference Editor. However, nothing has changed in the scene.



To reload the referenced tree as a proxy

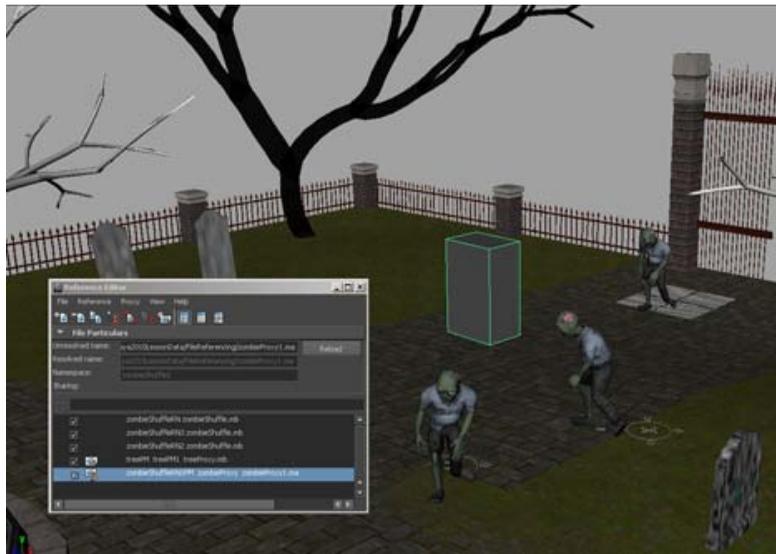
- 1 In the Reference Editor, right-click `treePM original tree.ma`.

- 2 From the pop-up menu, select Proxy > Reload Proxy as > treePM1 (treeProxy.ma).

The willow tree is replaced with the proxy you created. You can change the proxy back to its original form by repeating the above steps and reloading the proxy as original.

Note that zombieShuffleRN1 already has an associated proxy since you created it earlier as part of the proxy asset procedure. You can reload this zombie (as well as any other zombies you made proxies for) as its proxy using the same steps you used to reload the tree as a proxy.

Now if you play the scene, the performance should be greatly improved since there is much less complexity.



If you select any of the proxies in the Outliner and move them around the scene, the changes affect both the proxy and the original referenced file. This allows you to dress your scene and test it with simple proxies rather than with their more complex originals.

Tweaking the zombie's walk cycle

All the zombies in the graveyard scene currently walk exactly the same way. To add a little variety, you can create a tweaked walk cycle based on the original walk animation.

To tweak the zombie shuffle

- 1 Select File > Save Scene As, and save your current work under the name graveyard3.ma.
- 2 Create a new scene (File > New Scene).
- 3 Set the Playback End Time box to 200.



- 4 Select File > Create Reference.
The File Browser appears.
- 5 Select zombieShuffle.mb and click the Reference button.
The zombie with its shuffle animation are referenced into the empty scene.
- 6 Press '6' to view the zombie in shaded mode.
- 7 Select zombie_shuffle_AST in the Outliner (Window > Outliner).
- 8 Open the Channel Box / Layer Editor and select the Anim tab in the Layer Editor and make sure the layer named zombieShuffle:Arm_Edits is selected.
- 9 Open the Attribute Editor and click the zombieShuffle:zombie_shuffle_AST tab. On frame 1 of the animation, set the following attributes:
 - Left Shoulder FK Control Rotate: -8.848, -10.489, 43.284
 - Right Shoulder FK Control Rotate: -13.567, -21.276, -58.994The zombies arms rise into a reaching pose.



10 In the Attribute Editor, right-click the Left Shoulder FK Control Rotate attribute name and select Set Key from the pop-up menu. Do the same for Right Shoulder FK Control Rotate.

11 Repeat steps 8 and 9 at the following frames with the associated values:

Frame	Attribute	Value
18	Left Shoulder FK Control Rotate	10.694, -9.344, 61.577
26	Left Shoulder FK Control Rotate	8.239, 1.676, 65.611
33	Left Shoulder FK Control Rotate	-8.848, -10.489, 43.2845
33	Right Shoulder FK Control Rotate	-13.567, -21.276, -58.994

12 Play the scene.

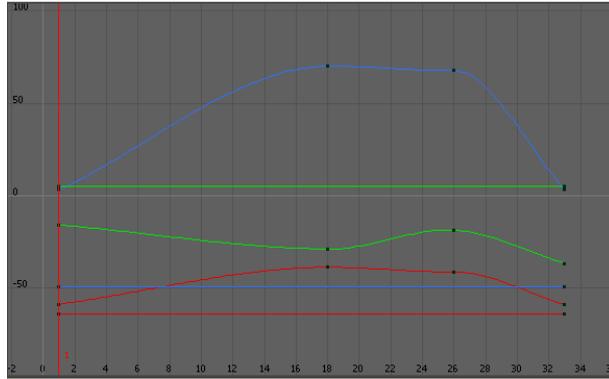
The zombie shuffles with its arms reaching forward. Since you made the change on an animation layer, the raised arm animation is added on top of the existing animation. For more information on animation layers, see *What are animation layers?* in the Animation book.



For the first 33 seconds of the animation, the zombie's arms behave properly. However, after that its left arm starts fanning out widely. This is because the tweaks you made to the animation layer only affect the first 33 seconds. To fix the rest of the animation, you need to loop your changes so that they span the entire walk cycle.

To loop the tweaks to the zombie's walk cycle

- 1 Select the zombieShuffle asset in the Outliner.
- 2 Select Window > Animation Editors > Graph Editor.
- 3 Under zombieShuffle:zombie_shuffle_AST > zombieShuffle:Arm_Edits, select Rotate X and then Shift + select the second Rotate Z.
These entries correspond to the attributes Left Shoulder FK Control Rotate and Right Shoulder FK Control Rotate.
- 4 Select View > Frame All.



- 5 Marquee select all the curves in the graph.
- 6 Select Curves > Post Infinity > Cycle.
- 7 Click an empty spot in the graph to deselect the curves.
- 8 Select Curves > Bake Channel.
If you zoom out of the graph, you can now see that all the curves cycle over the first 200 seconds.
- 9 Close the Graph Editor.
Now if you play the animation, the zombie's arms behave properly for the entire walk cycle. If you look at the time slider, you can see that there is now a keyframe on every frame of the animation.

Exporting reference tweaks

The tweaks you've made to the zombie shuffle have not actually changed the original zombie animation. Instead, the changes reside in the current scene file as reference edits.

To view reference edits

- 1 Open the Reference Editor (File > Reference Editor).
- 2 In the Reference Editor, select the zombieShuffle reference.
- 3 Select File > List Reference Edits.
Maya displays a list of the node and attribute values that have changed since the zombie file was referenced.

```

----- setAttr -----
setAttr FKShoulderCtrl_L.Left_Shoulder_FK_Control_rotate -type "double3" 9.950894 -11.146023 65.418587
setAttr FKShoulderCtrl_L.Left_Shoulder_FK_Control_rotateZ -av
setAttr FKShoulderCtrl_L.Left_Shoulder_FK_Control_rotateY -av
setAttr FKShoulderCtrl_L.Left_Shoulder_FK_Control_rotateX -av
setAttr FKShoulderCtrl_R.Right_Shoulder_FK_Control_rotate -type "double3" -19.027408 -20.144997 -61.089814
setAttr FKShoulderCtrl_R.Right_Shoulder_FK_Control_rotateZ -av
setAttr FKShoulderCtrl_R.Right_Shoulder_FK_Control_rotateY -av
setAttr FKShoulderCtrl_R.Right_Shoulder_FK_Control_rotateX -av
setAttr JKSpine.translate -type "double3" -1.3549 20.387555 4.939279
setAttr JKSpine.rotate -type "double3" 41.77981 14.958219 8.399076
setAttr lyr_controls.visibility 0
setAttr Shuffle_DragLeg.lock 1

----- connectAttr -----
connectAttr "AnimLayerRoot.solo" "BaseAnimation.childrenSolo[0]"
connectAttr "BaseAnimation.childSoloed" "AnimLayerRoot.siblingSolo"
connectAttr "AnimLayerRoot.parentLayer" "BaseAnimation.childrenLayers[0]"
connectAttr "BaseAnimation.foregroundWeight" "AnimLayerRoot.parentWeight"
connectAttr "BaseAnimation.outMute" "AnimLayerRoot.parentMute"
connectAttr "FKShoulderCtrl_R_rotate_Arm_Edits_inputBX.output" "FKShoulderCtrl_R_rotate_Arm_Edits_inputBX"
connectAttr "FKShoulderCtrl_R_rotate_Arm_Edits_inputBY.output" "FKShoulderCtrl_R_rotate_Arm_Edits_inputBY"
connectAttr "FKShoulderCtrl_R_rotate_Arm_Edits_inputBZ.output" "FKShoulderCtrl_R_rotate_Arm_Edits_inputBZ"
connectAttr "FKShoulderCtrl_L_rotate_Arm_Edits_inputBX.output" "FKShoulderCtrl_L_rotate_Arm_Edits_inputBX"
connectAttr "FKShoulderCtrl_L_rotate_Arm_Edits_inputBY.output" "FKShoulderCtrl_L_rotate_Arm_Edits_inputBY"
connectAttr "FKShoulderCtrl_L_rotate_Arm_Edits_inputBZ.output" "FKShoulderCtrl_L_rotate_Arm_Edits_inputBZ"

```

From this you can see that the zombieShuffle file still contains the original, unaltered zombie walk, while the current scene referencing zombieShuffle contains only the tweaks you've made since.

4 Close the List Reference Edits window.

You can export the reference edits in the current scene for use in your graveyard scene.

To export the reference edits

- 1** In the Reference Editor, select the zombie reference.
- 2** Select File > Export to Offline File.
The File Browser window appears.
- 3** In the Options section, set the following:
 - Export nodes: Connected to referenced file for selection
 - Include setAttr edits: On

NOTE This tutorial assumes you are using the default Maya File Browser. If these options do not appear in the File Browser, change back to the default browser by selecting Window > Settings/Preferences > Preferences, and then going to the Files/Projects section and setting the File Dialog option to Maya Default.

- 4 Name the file armTweak1, and then click the Export to Offline File button. Maya creates an edits file containing your animation tweaks.

Now you are ready to apply the tweaked animation to zombies in the graveyard file.

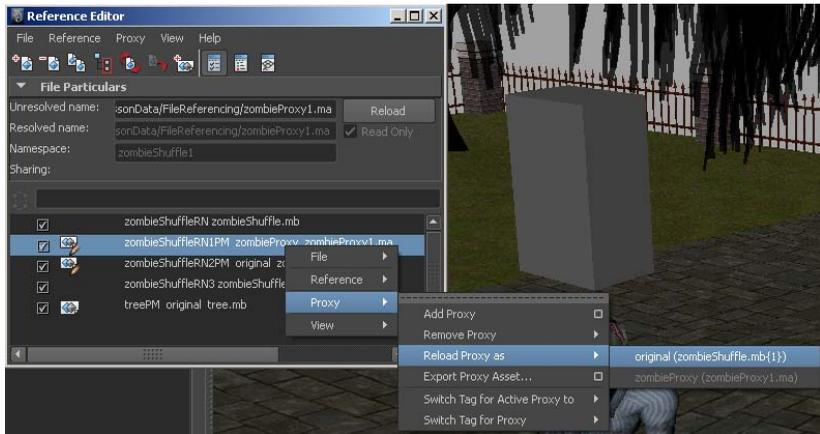
To prepare for the next part of the tutorial, re-open the graveyard3.ma scene (File > Open Scene). You do not need to save the current scene.

Applying reference tweaks

Now that you have a tweaked animation for your zombie, you can apply it to your graveyard scene.

To apply the tweaked zombie shuffle

- 1 Open the Reference Editor (File > Reference Editor).
- 2 Select any of the referenced zombies in your scene.
- 3 If the zombie is currently loaded as its proxy, change it back to its original reference by right-clicking it in the Reference Editor and selecting Proxy > Reload Proxy as > Original.



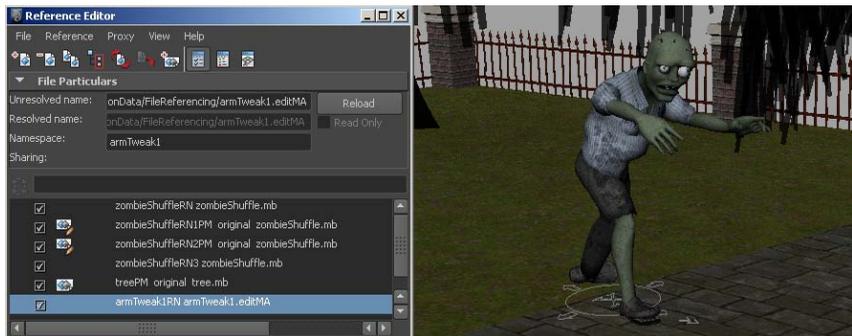
- 4 Select File > Assign Offline File.

The File Browser appears.

- 5 Navigate to the folder where you previously stored the edit file and select armTweak1.ma, then click the Assign Offline File button.

NOTE If you cannot find the file, make sure the Files of type field at the bottom of the file browser is set to Offline edit Ascii.

A new entry appears in the Reference Editor.



- 6 Play the scene.

The zombie now shuffles with the same alterations you made in the previous steps.

You can toggle between the old default animation and your new tweaked animation by clicking the checkbox next to the name `armTweak1RN` `armTweak.mb` in the Reference Editor to load and unload the reference edit. You can also apply the tweaked animation to any or all of the other zombies using the same method above.



If you were so inclined, you could also create additional shuffle variations and apply them all to the same zombie, then use the checkboxes to quickly toggle between them.

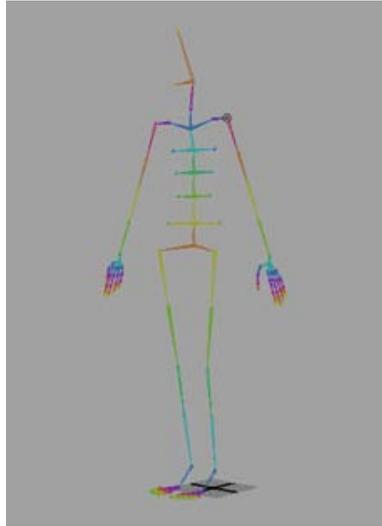
Storing animation as reference edits

In the previous steps you tweaked an existing animation and exported those tweaks so that you could apply it to other references with the same animation. However, you can go one step further than this and export entire animation sequences as reference edit files.

To open the next scene

- 1 Select File > Save Scene As, and save your current work under the name `graveyard4.ma`.
- 2 Create a new scene (File > New Scene).
- 3 Set the Playback End Time box to 200.

- 4 Select File > Create Reference.
The File Browser appears.
- 5 Select model_skeleton_noanim.mb and click the Reference button.
A plain skeleton with no animation is referenced into the scene.

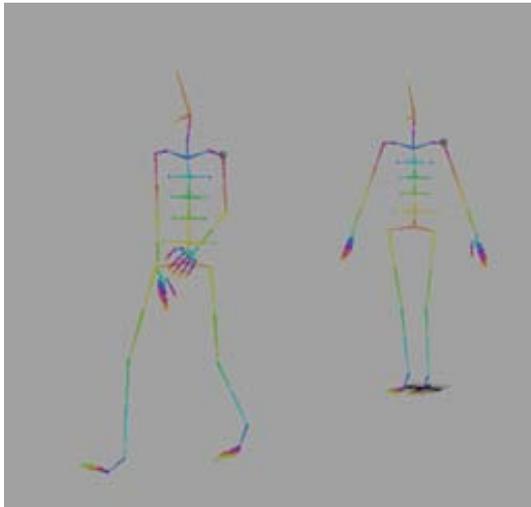


The skeleton you've referenced is very similar in structure to the zombie's skeleton. However, the asset is not exactly the same. Most notably, this skeleton is missing some joints and its rest pose is not exactly the same as the zombie's.

You will now animate this skeleton as a series of reference edits. To save time, we have provided you with an animated skeleton that you can transfer attributes from. This has the same effect as if you had animated the skeleton by hand.

To animate the skeleton

- 1 Select File > Import.
The File Browser appears.
- 2 Select model_skeleton_anim.mb and click the Import button.
- 3 Play the scene.
An identical skeleton performs a runway model-like strut and strikes a pose.



- 4 Open the Outliner (Window > Outliner).
- 5 Select `model_skeleton_anim:model_AST` and then Shift + select `model_skeleton_noanim:walk_rig_AST`.
- 6 Select Edit > Transfer Attribute Values > .
The Transfer Attribute Values Options window appears.
- 7 Select Edit > Reset Settings.
- 8 Set the following:
 - Values: On
 - In Connections: On
 - Transfer from source: On
- 9 Click Apply and Close.
- 10 Play the animation.
Although it looks like nothing changed, notice that `model_AST` is now animated, while `model_anim_AST` remains still.

You are now ready to export the reference edits.

To export the reference edits

- 1 Open the Reference Editor (File > Reference Editor).
- 2 Select the skeleton reference.
- 3 Select File > Export to Offline File.
The File Browser appears.
- 4 Name the file `runwayStrut` and click the Export to Offline File button.
Maya exports the edits.

The file `runwayStrut.exportMA` now contains the edits necessary to turn any skeleton asset with matching attribute names into a runway strut. You can store this file for you or anyone else to use on any skeleton you create in the future.

Now you are ready to use the runway animation in the graveyard file.

To prepare for the next part of the tutorial, re-open the `graveyard4.ma` scene (File > Open Scene). You do not need to save the current scene.

Applying animation as reference edits

To change this scene from a horror-filled graveyard to an undead fashion show, you will apply the runway strut animation to one of the zombies in the scene.

However, all the zombies are animated. The runway strut animation you exported is meant to be used on a non-animated skeleton. Thus, you need to replace one of the zombie references with its non-animated counterpart.

To replace a reference

- 1 Open the Reference Editor (File > Reference Editor).
- 2 Select `zombieShuffleRN` `zombieShuffle.mb`.
- 3 Select Reference > Replace Reference.
The File Browser appears.
- 4 Select `zombiePlain.ma` and click Reference.
The animated zombie is replaced with a non-animated version.



The non-animated zombie model is more appropriate for the strut animation since it more closely matches the reference that the strut was created from. The asset has all the correct published attributes, however the default View hides them from you.

To apply the strut animation

- 1 Select `zombieShuffleRN zombiePlain.mb` in the Reference Editor.
- 2 Select File > Assign Offline File.
The File Browser appears.
- 3 In the `scenes\edits` folder, select `runwayStrut.editMA` and click the Reference button.
The zombie changes positions. If you play the scene, the zombie walks with the same runway strut as the animated skeleton.



Beyond the Lesson

In this lesson you learned how to break up your scenes into pieces and reference those pieces into a single file. You learned how to:

- Create a file reference
- Duplicate a file reference
- Use a proxy reference for objects outside and inside of an asset
- Export animation tweaks as reference edits
- Use reference edits and assets to apply animation to a non-matching skeleton.

For the purposes of this tutorial you did all the steps yourself. However, imagine a multiple artist scenario in which one artist is in charge of the main scene, while another artist is in charge of the character modeling, and another in charge of the character animation. If the scene is referenced properly, each artist can work on their portion of the scene simultaneously in their own files without having to wait for other artists to finish their work.

Introduction



With Maya® Hair™, 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 Effects™ 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 786
- Lesson 2 Creating a dynamic non-hair simulation: [Introduction](#) on page 808

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 Time Slider 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](#) on page 25 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	<p>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.</p> <p>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.</p>
Current Position	<p>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.</p>

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 785.
- 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.

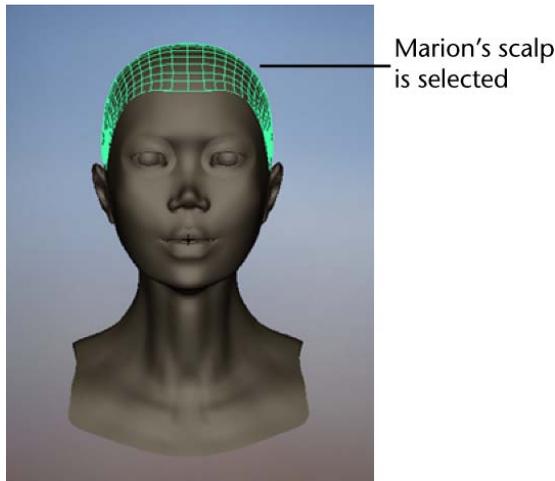
Creating hair on a surface

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.
 - Ensure that Create Rest Curves is turned off.

- Ensure that Grid is turned on and 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.
- Ensure that Edge Bounded is turned off.
- Turn the Equalize option on. This ensures the hair curves are evenly distributed over the surface.
- Ensure that Dynamic is turned on.
- 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.
- 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 0.10. 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 > From Current.
- 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.

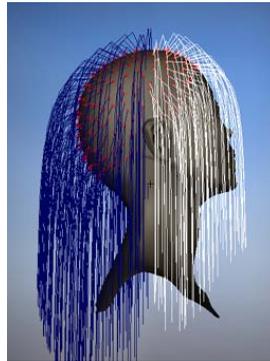
To edit the Start curves to create the basic hair shape

- 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.

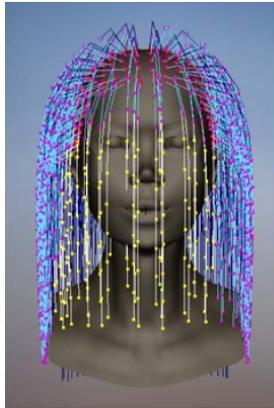


- 4 Now tumble the camera so Marion is again facing you.
- 5 To trim the Start curves in the front, do the following:
 - Select Hair > Modify Curves > Lock Length. This allows you to manipulate the curve while maintaining the curve's hull length.

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



Selected CVs
are curled
under using
Bend

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.

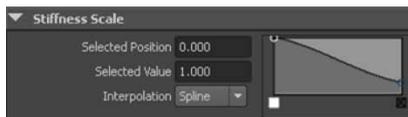
Watch the hair try to achieve the Rest position.

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

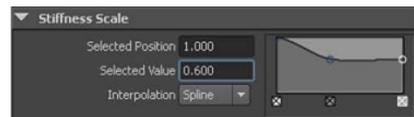
6 As the hair starts to fall due to gravity, increase the Stiffness value to 0.2. If some Rest curves won't relax and stabilize, increase Length Flex and Iterations in the hairSystemShape1 tab.

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



To modify 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

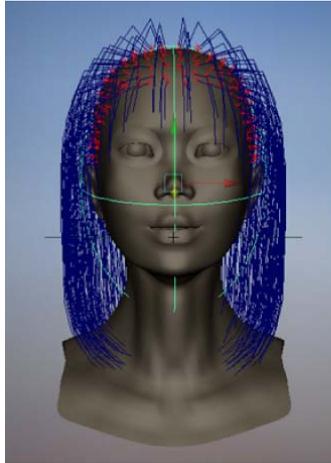
Before trimming Marion's hair you may have noticed that some hairs intersected her face. As well hairs currently intersect the lower part of her neck. To fix this you set up hair collisions. You can set hair to collide with itself, other objects or the ground (grid). In the next steps you use collision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



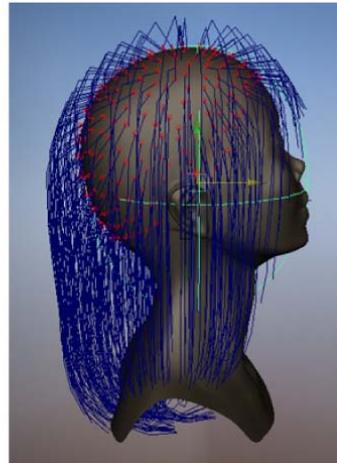
Notice the hairs at the bottom are intersecting with the upper part of Marion's torso

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.

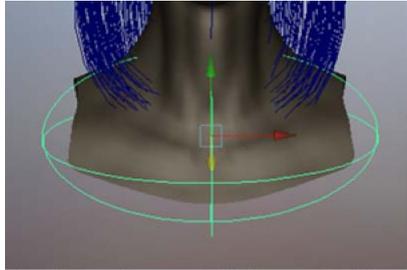


Notice the sphere width from ear to ear and height from the top of Marion's head to below her chin

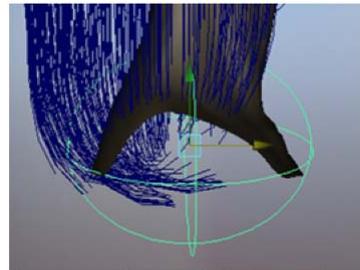


Notice the sphere width from the back of Marion's head to her nose

- 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 and you must render the hair. In the next steps you use the mental ray for Maya renderer to render Marion's hair as it is now. Rendering the hair allows you to see which aspects need work to make it look more realistic. Before rendering, be sure to set the appropriate Render Settings.

Using the mental ray for Maya renderer

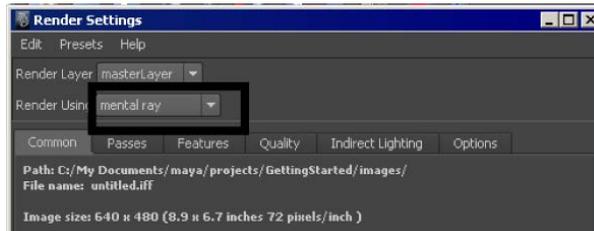
In this tutorial, you will render using the mental ray for Maya renderer.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.



To render the hair using the mental ray for Maya renderer

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

- 2 Click the Render Current Frame  button on the Status Line.

The hair model is rendered using the mental ray for Maya renderer. Your rendered hair model should look something like the image below.



This is how the hair looks in the mental ray 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 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

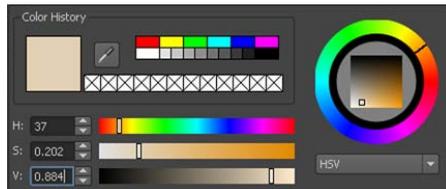
- 1 Select Window > Outliner.
- 2 In the Outliner menu, select Show > Objects > Lights.
- 3 Select one of the three directional lights.
- 4 In the light's Attribute Editor, go to the Shadows section and in the Depth Map Shadow Attributes subsection, turn on Use Depth Map Shadows.

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 specular. 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 (see Automatic 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

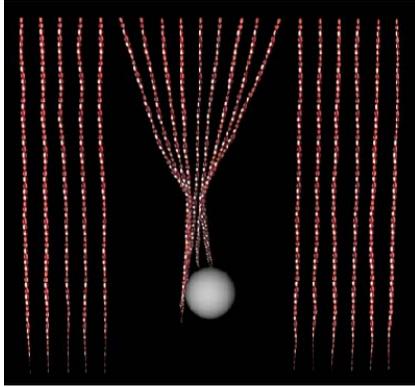
Lesson 2: Creating a dynamic non-hair simulation

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

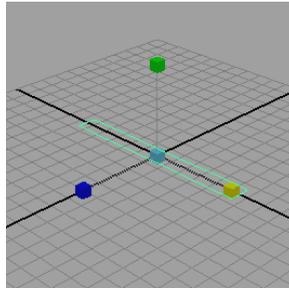
In these steps you model a 2D curtain rod from a plane and attach hair to it. Then you add and keyframe a sphere in the scene to later be used as a collision object.

To create the curtain using Hair

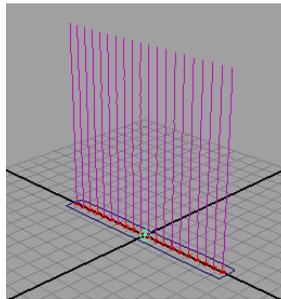
- 1 To create a NURBS plane, select Create > NURBS Primitives > Plane, then click-drag in the scene to create a plane.
(If you have previously shut off the interactive primitive creation option, simply click once in the scene to create the primitive at the origin.)



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

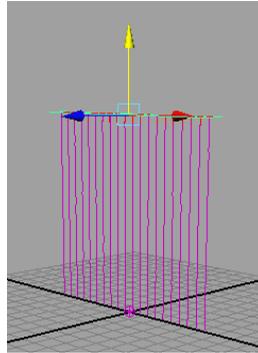


- 3 With the plane selected, select Hair > Create Hair > , set the following options and then click Create Hairs:
- Output to NURBS Curves
 - U Count to 20, V count to 1
 - Passive fill to 0
 - Randomization to 0
 - Points per hair to 12
 - Length to 10
 - Turn off the options Create Rest Curves, Edge Bounded and Equalize
- Hair is created and attached to the plane.

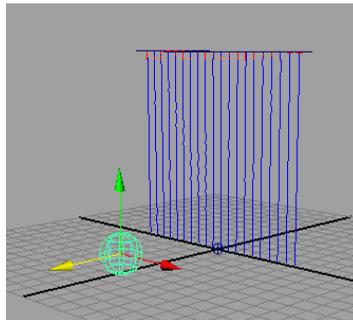


- 4 Select the plane and, using the Rotate Tool , rotate the plane so the hair appears to be on the underside of the plane. Then, using the Move Tool , move the plane up the Y axis so the bottom of the

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

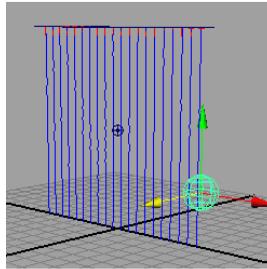


- 5 Create a directional light (Create > Lights > Directional Light) and do the following:
 - Move the light above and to the left of the curtain.
 - Rotate the light so it aims at the curtain.
 - In the light's Attribute Editor (ctrl + a) go to the Directional Light Attributes section and change the 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.

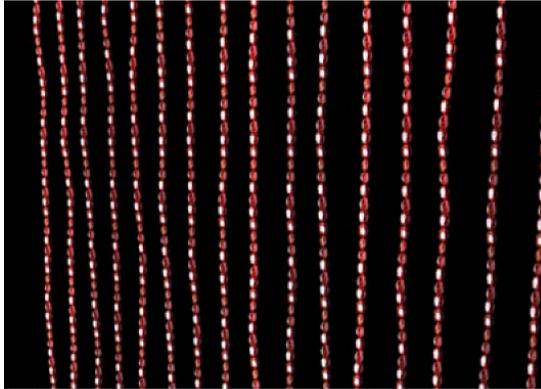
Assigning a Paint Effects brush to the hair

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 mental ray for Maya 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 Shading section; under Hair Color Scale, set Hair Color to white so the brush's bead color appears
 - In the Hair Color Scale 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 Before you can render in mental ray for Maya, you must convert Paint Effects to polygons. Drag around the hair curves to select the hair system and select Modify > Convert > Paint Effects to Polygons.
- 8 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.

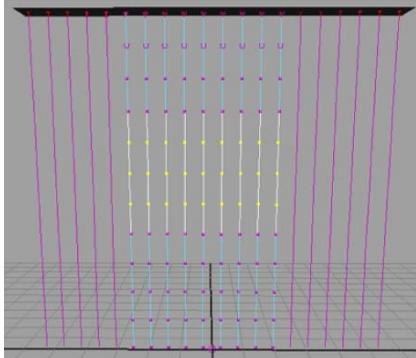
To model the curtain into a tied-back position

- 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.

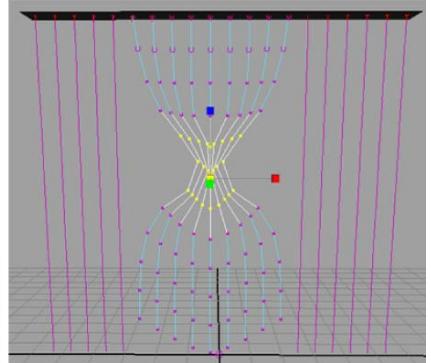
- 3 Select a block of Start curves in the middle of the curtain.
- 4 Select Hair > Modify Curves > Lock Length. This ensures the entire curve is modified when you transform CVs in the next steps.



- 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

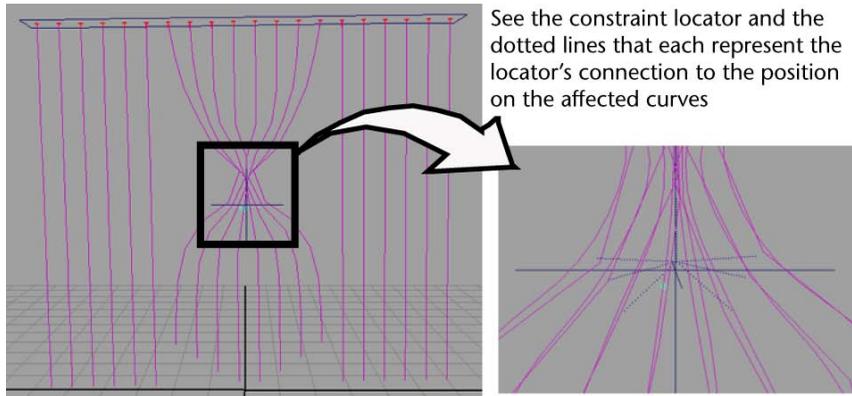


The middle rows of CVs are scaled inwards so the hair curves appear bound

To constrain the curtain in the tied-back position



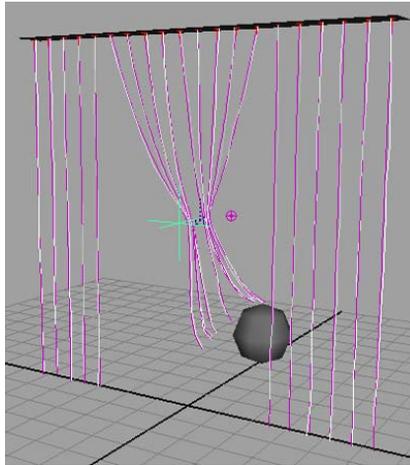
- 1 Click the  button on the Status Line to display the Start curves (not the CVs).
- 2 With the bound hair curves still selected, select Hair > Create Constraint > Hair to Hair.
A constraint locator is created and connected to the selected curves.



- 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

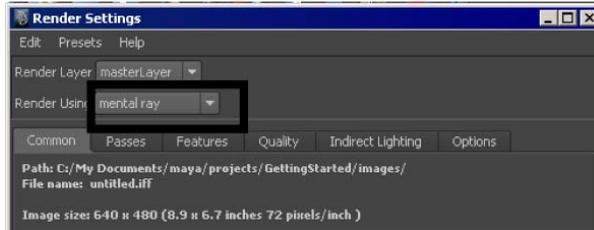
In these next steps you render the curtain, which is Paint Effects hair, using the mental ray for Maya renderer.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.

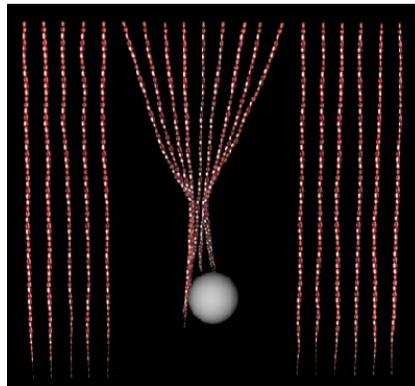


To render the curtain scene

- 1 Before rendering be sure to display the hair system's Current Position (Hair > Display > Current Position).
- 2 Play the simulation and stop at the frame you want to render.

- 3 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.

Introduction



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

Using Maya Fluid Effects™, 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 823
- Lesson 2 Creating a non-dynamic 3D fluid effect: [Introduction](#) on page 833
- Lesson 3 Creating a dynamic 3D effect: [Introduction](#) on page 842
- Lesson 4 Creating an ocean effect: [Introduction](#) on page 852

Preparing for the lessons

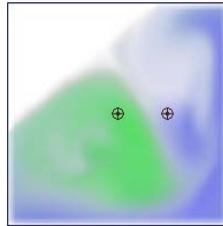
To ensure the lessons work as described, do these steps before beginning:

- 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](#) on page 25 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.

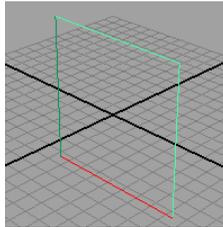
Creating a two-dimensional fluid container

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

- 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

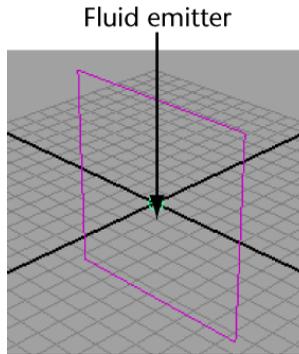
Fluid emitters create fluid property values and emit them into containers as the simulation plays. The emitter controls the position, rate of emission, and turbulence forces applied to the fluid properties when they are created. After fluid property values are created, the attributes of the fluid container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

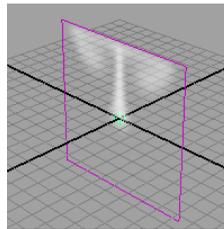
Maya creates a fluid emitter called `fluidEmitter1` and places it at the center of the fluid container.



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

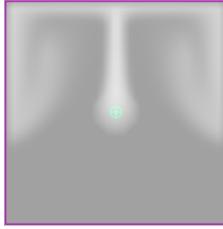
- 2 Play the simulation using the playback controls at the bottom of the Maya window.

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.

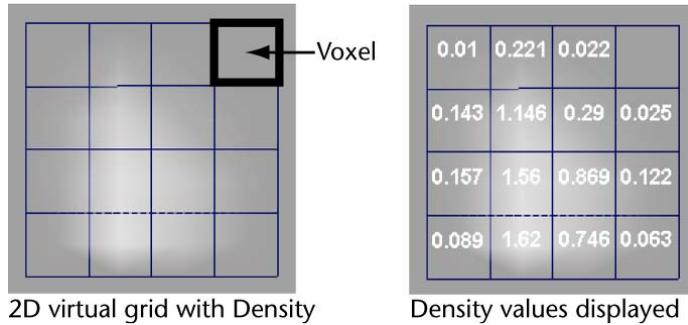
In the next steps you'll look at some of the attributes of the fluid container to gain some understanding into dynamic fluid effects.

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.



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.

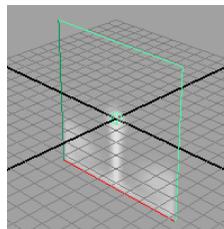
For dynamic fluid simulations, Density must be set to Dynamic Grid. Also, Velocity cannot be turned Off because velocity moves the property values around inside the grid.

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.)

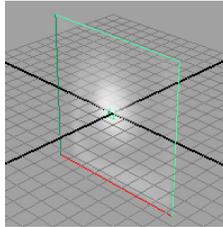
Negative values move the Density down.

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

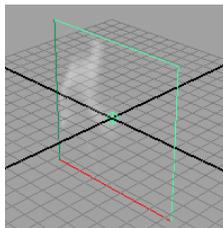


- 6 To add turbulence to the container and give the Density a whirling circular motion.
 - Ensure the fluid container is selected if it is not already.

- 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.

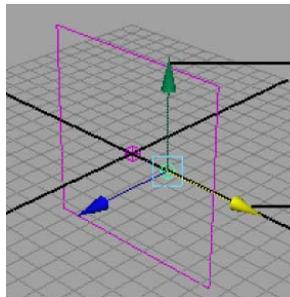
To combine colored fluids in the container

- 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.

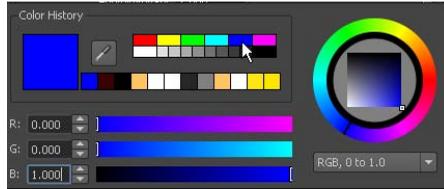


- 3 Open the Attribute Editor in the right panel of the Maya window, and click the `fluidEmitter2` tab.
- 4 In the Fluid Attributes section, turn on Emit Fluid Color.

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

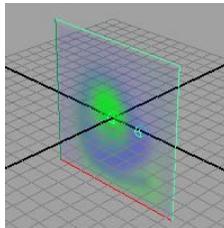
By setting the Color Method to Dynamic Grid, the color values you place in the container grid are recalculated by the fluid dynamics solver and changed during simulation. This means colors can interact during the simulation.

- 5 Click Set to Dynamic.
- 6 Click the Fluid Color color swatch in the Attribute Editor.
The Color Chooser opens.
- 7 In the Select *blue* from the Color Chooser.



Move your mouse off the Color Chooser to close it.

- 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 swatch.
The Color Chooser opens.
- 11 Select *green* from the Color Chooser.
Move your mouse off the Color Chooser to close it.
- 12 Play the simulation using the playback controls at the bottom of the Maya window.



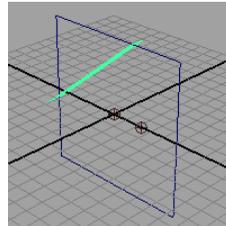
- 13 Stop the playback and go to the start of the playback range.

Colliding a fluid with an object

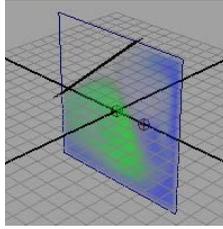
The 2D fluid you created is dynamic, so it can interact with objects. In the next steps, you'll create a plane inside the fluid container and make the fluid collide with it, rather than pass through it.

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 now 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 mental ray 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 822.

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 In the Create 3D Container Options window, set the following:

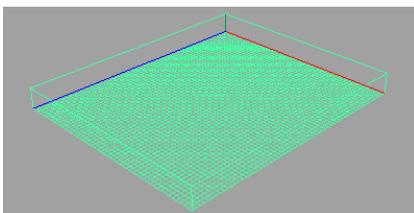
- X resolution: 50
- Y resolution: 5
- Z resolution: 60
- X size: 50.0
- Y size: 5.0
- Z size: 60.0

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.

- 3 Click Apply and Close.

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



Next, you add fluid to the container.

Adding fluid to a container

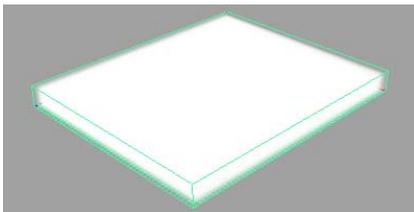
To make this a non-dynamic fluid, you add fluid property values to the container that will stay constant over time. There are two ways to do this: add these values to a static grid, or select a set of values from a list of predefined gradients that come with Maya. For this lesson, you'll use the latter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

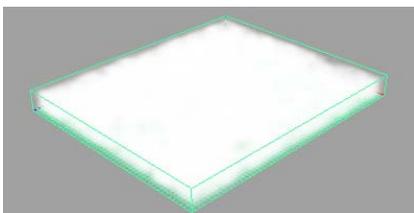
- 1 Open the Shading section in the Attribute Editor.
- 2 In the Color subsection, change Color Input to Density.
The Density values in the container will now take on the colors defined on the color bar. Leave the color white to make the clouds appear white.
- 3 In the Opacity subsection, check that 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.

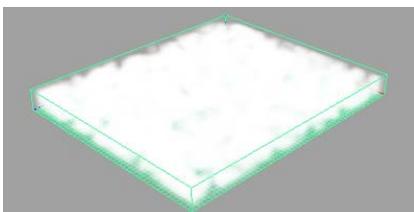
To texture the density of a fluid

- 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.

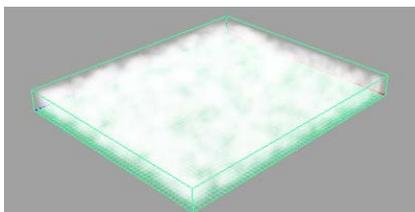


Notice that the Density now has a slightly blotchy look to it, with areas that are more opaque and areas that are more transparent. This texture provides the standard 3D noise used in the 3D Solid Fractal texture included with Maya.

- 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.



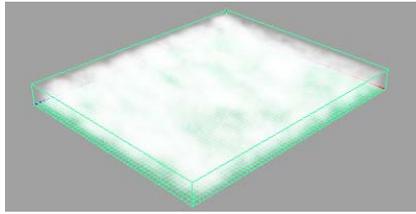
- 5 Change the look of the texture by setting the following texture attributes:
 - Amplitude: 0.5
 - Depth Max: 4



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

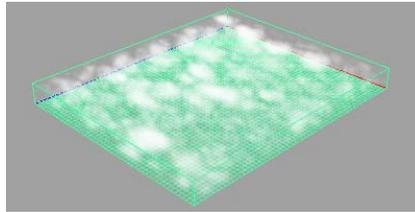
Increasing Depth Max adds detail. Increasing it will also increase render time.

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



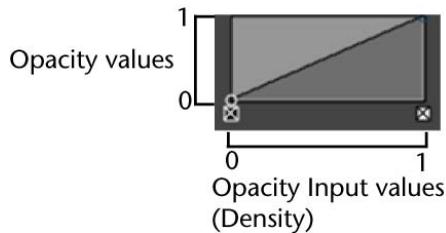
7 Change the following Billow texture attributes to make the “billows” less dense, more spotty, and with randomly different sizes.

- 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).

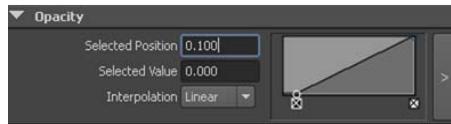
So for the linear graph shown above, where Density values are 0, Opacity values are 0, making the Density totally transparent, where Density values are 0.5, Opacity values are 0.5, making the Density partially opaque, and where Density values are 1, Opacity values are 1, making the Density totally opaque.

- 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.

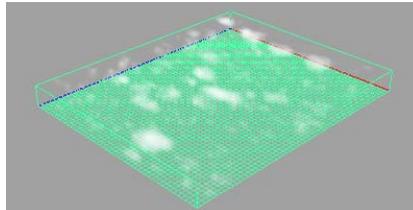


Click this dot to select the position marker.

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



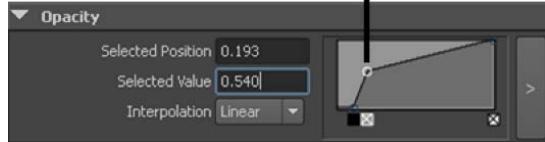
The position marker moves to the right. Now, for Density values between 0 and 0.10, the Opacity values will be 0. This means that Density that was previously partially transparent will be completely transparent.



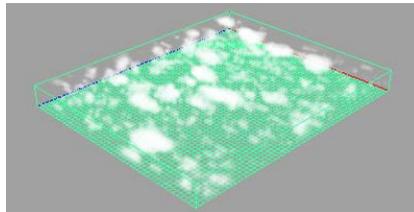
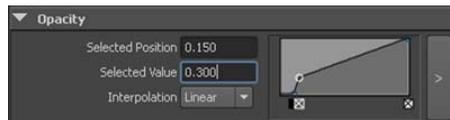
The more transparent areas of cloud disappeared, but now the solid areas of cloud are less opaque.

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

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



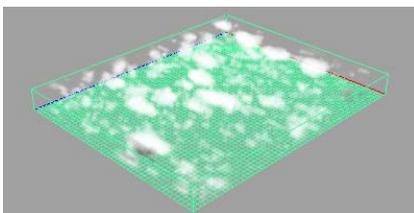
Density values that are greater than 0.15 are now more opaque, and the transition between areas of total transparency (Opacity 0), and areas where the Density becomes more visible (Density 0.15) is less gradual.

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.

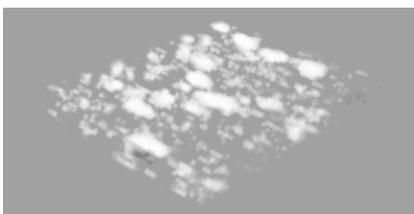
To add self shadowing to the texture density

- 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

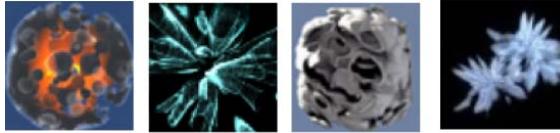
This lesson showed you a basic technique for creating non-dynamic fluids. While creating the effect, you learned how to:

- 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.

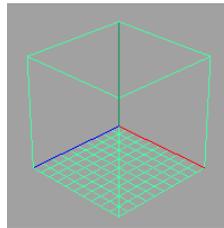
In this lesson, you create a dynamic explosion effect by filling the bottom of a container with Fuel and Density, then “igniting” it with Temperature. You learn how to:

- 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 822.

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

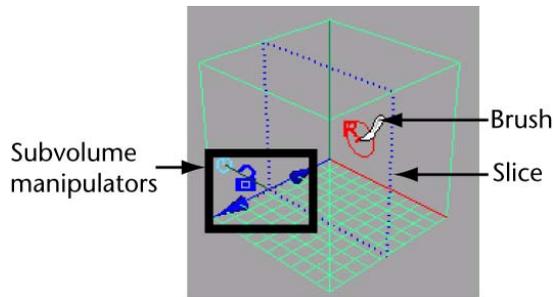
In Fluid Effects, Fuel defines the state of a reaction (unreacted, completely reacted, and states in between). Density represents the substance that is reacting, while Temperature causes the reaction. Combining Density with Fuel defines a situation in which you can see a reaction. You add Temperature 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 > .

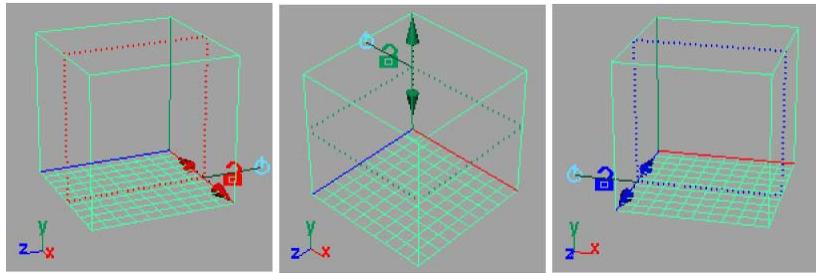
The Tool Settings window opens and a slice displays at the origin of the fluid container. The slice is represented by a plane with dotted edges and fluid subvolume manipulators at one corner. When you move the pointer over the slice the pointer changes to a brush indicating that you can paint.



- 2 At the top of the Tool Settings window, click Reset Tool to set the Paint Fluids Tool settings to the default values.
- 3 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.
- 4 Click Set to Dynamic.

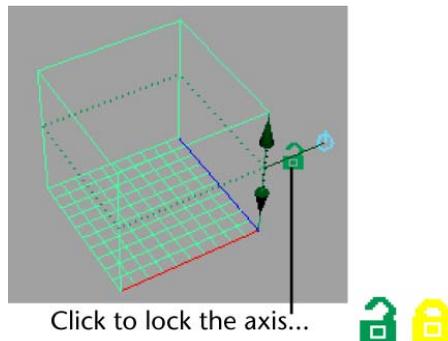
- 5 Tumble the scene and look at the slice and manipulators.



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.

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

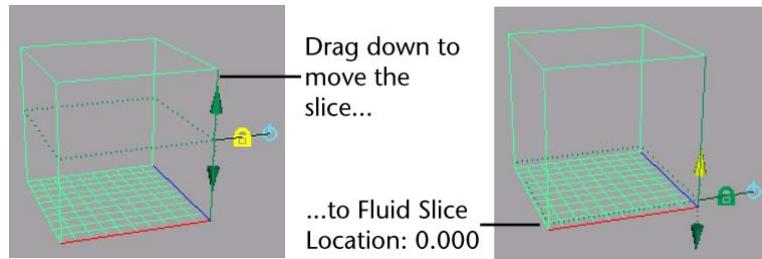


The lock closes. Now when you tumble, the slice remains perpendicular to Y. It does not switch to the other axes. This allows you to change the view in any way and still be able to paint on the same slice.

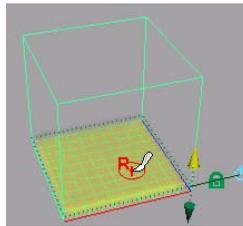
- 7 Drag the top move arrow down along the axis to the first slice at the bottom of the container. The number of slices along each axis corresponds with the resolution of the fluid container. The numbering starts at 0 at the intersection of the axes. For this container the resolution is 10 10 10, so the slices are numbered from 0 to 9.

The help line displays the location of the selected slice.

Fluid Slice Location: 0.000



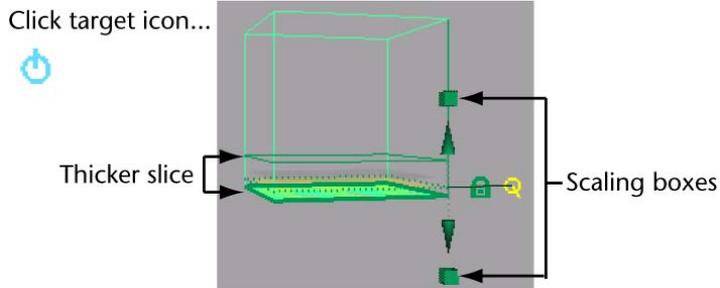
- 8 In the Paint Attributes section of the Tool Settings window, ensure that Value is set to 1. A Fuel value of 1 represents a totally unreacted state while a Fuel value of 0 represents a completely reacted state. A Density value of 1 is totally opaque while a Density value of 0 is completely transparent.
- 9 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.

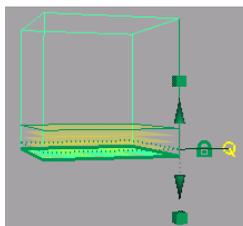
- 10 Resize the paint slice, making it thicker:
 - Click the target icon once to switch to the subvolume scaling mode.



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

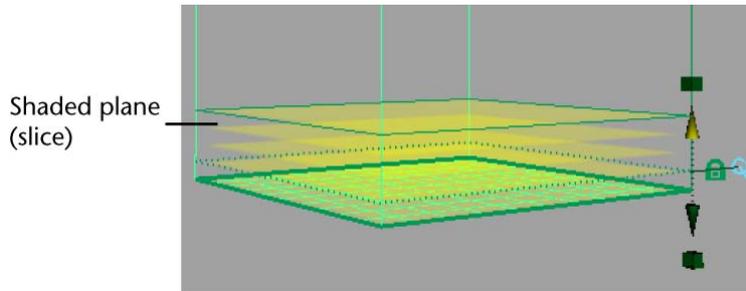
- Click the top scaling box and read the help line to determine that the Fluid Slice Thickness is 3.00. You could drag the scaling boxes to change the thickness of the slice, but the slice is thick enough for the lesson.

11 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.

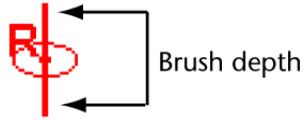
Painting Temperature values into a container

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

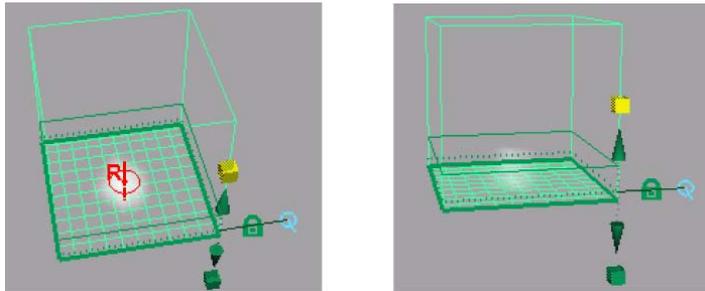
- 1 In the Tool Settings window, beside Paintable Attributes, select Temperature. A message appears prompting you to set the fluid's Temperature method to Dynamic Grid or Static Grid.
By setting the Temperature method to Dynamic Grid, the temperature values you place in the container grid are recalculated 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.

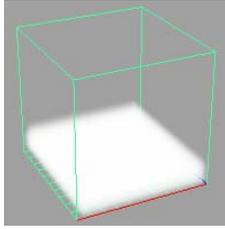
The next steps are to view all the property values together, and color the properties to differentiate them.

Adding color to Density and Temperature

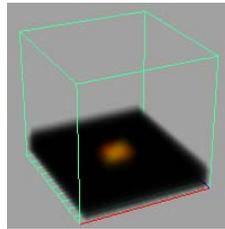
Use the built-in shader to modify the color of Density and Temperature in the container.

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



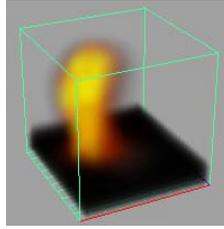
- 2 With the fluid container selected, open the Attribute Editor.
- 3 Change the color of the fluid as follows:
 - In the Shading section of the Attribute Editor, expand the Color section, and change Color Input to Density.
 - The Density values in the container will now take on the colors defined on the color bar.
 - Click the Selected Color swatch. The Color Chooser opens.
 - Select black in the Color Chooser.
 - Move your mouse off the Color Chooser to close it.



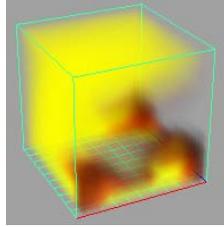
The Density turns black.

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.

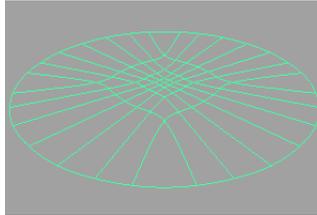
Before beginning the lesson, do the steps in [Preparing for the lessons](#) on page 822.

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.

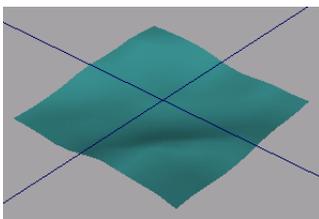
Because interactive shading would slow down playback, the shading is turned off for the surface.

Adding a preview plane to an ocean

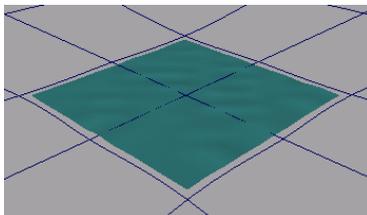
Because shading is turned off for the ocean plane, you need a preview plane to see displacement and shading in the hardware render. Displacement changes the surface of the ocean plane so you can see the different wave heights.

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

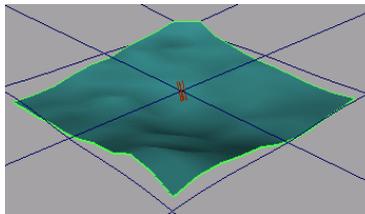
- 1 Render the current frame to see what the ocean looks like before you modify the ocean shader attributes. On the Status line, click the render current frame icon.



- 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.

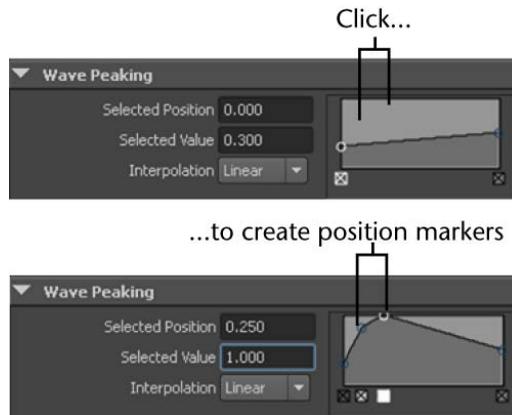
- 6 Render the current frame.



- 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.

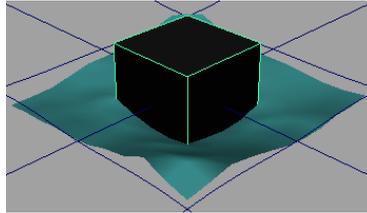


Floating objects

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

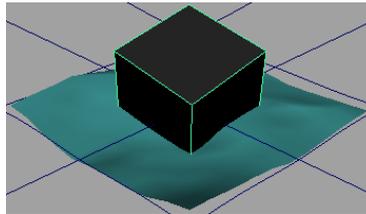


- 3 Select both the cube and the ocean plane (not the preview plane) at the same time and select Fluid Effects > Ocean > Make Boats.

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.

Introduction



With the Maya® Fur™ 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, 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 862

- Lesson 2 Rendering fur: [Introduction](#) on page 887

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](#) on page 25 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 Paint Effects menu.

Lesson 1: Assigning a fur description

Introduction

Maya 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

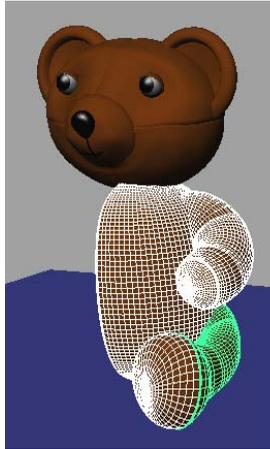
To ensure the lesson works as described, do these steps before beginning:

- 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

- ▶ 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
<code>leftArm1</code>	<code>rightArm</code>
<code>leftArmpaw1</code>	<code>rightArmpaw</code>

Old name	New name
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.

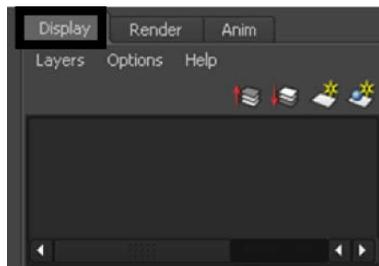
Layers are overlapping views of your scene that can have objects assigned to them. You can assign any objects within a scene to *layers*. One useful characteristic of layers is that they let you simplify your scene so you can work on specific objects without disturbing other objects. You can choose which 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



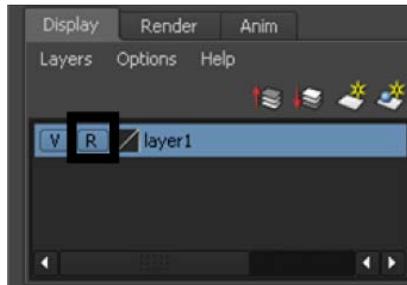
- 2 Ensure the Layer Editor is set to Display Layers by clicking the Display radio button.



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 be 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.

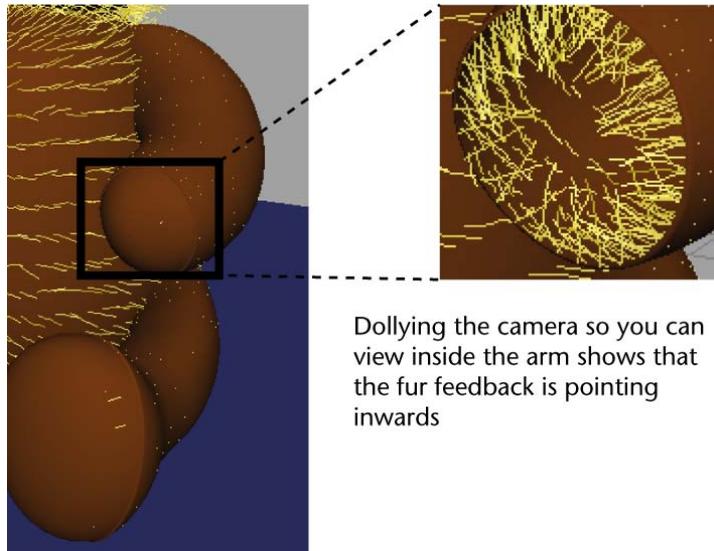
Fur feedback shows various fur attributes such as color, density, length, direction, and scraggle. By looking at the fur feedback you can immediately see how the changes you make to the fur attributes affect the appearance of the fur, without having to render the scene.

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.

- 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.

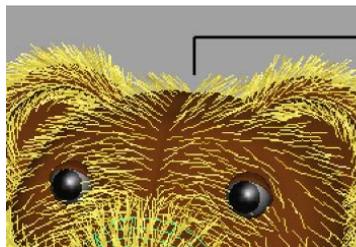
- 3 Select Fur > Offset Fur Direction and click on the double lines at the top of the menu that appears so you can tear off the Offset Fur Direction menu.



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.

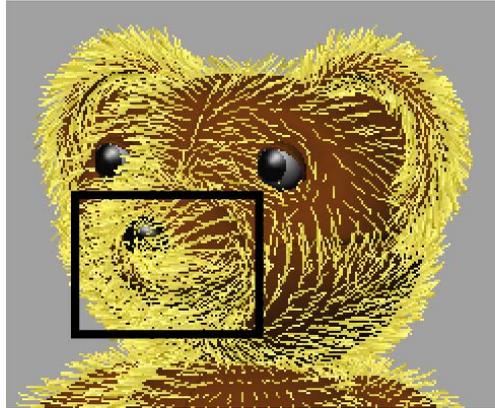


Set the Fur Direction Offset so the fur appears to part along the center line of the head.

- 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:

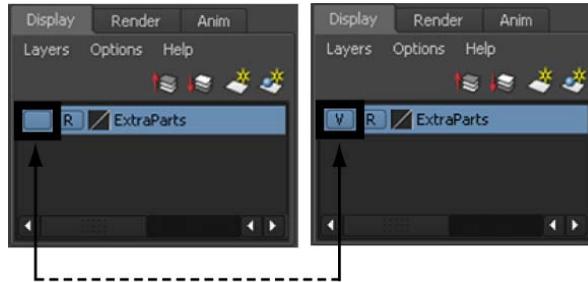
- reduce the fur length in a localized area on a surface. This is useful when you want regions with shorter or longer fur on a surface.
- create bald regions on a surface. This is useful for creating localized fur effects such as a moustache or goatee on a character, or whiskers on an animal.

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.

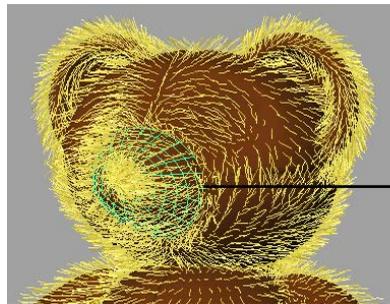
To hide the referenced display layer

- 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.

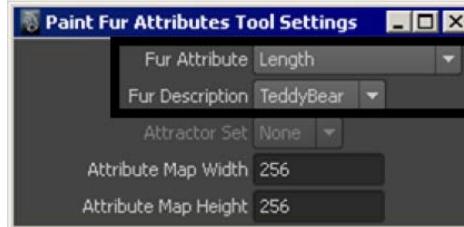


Select the snout surface before selecting the Paint Fur Attributes Tool.

- 2 Select Fur > Paint Fur Attributes Tool > .

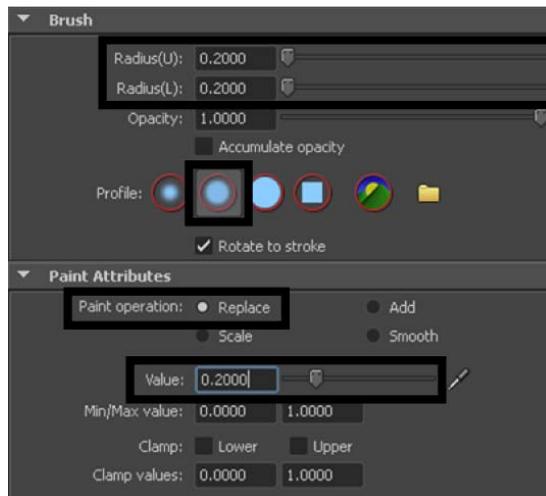
The cursor updates to display as a paint brush indicating that the Paint Fur Attribute Tool is active. The Paint Fur Attribute Tool Settings window and the Paint Tool Settings editor appears. These windows let you set and control which attributes get painted for an attribute map.

- 3 In the Paint Fur Attributes Tools Settings window, ensure the following options are set:
 - Fur Attribute: Length
 - Fur Description: TeddyBear

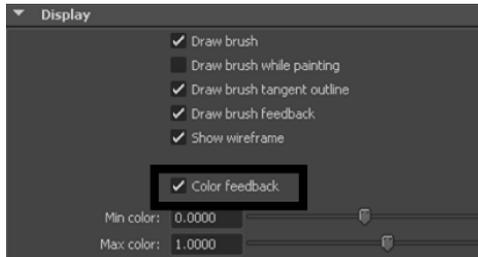


- 4 In the Paint Tool Settings editor, set the following:
 - Brush - Radius (U): 0.2
 - Brush - Radius (L): 0.2
 - Brush - Profile: Soft
 - Paint Attributes - Paint Operation: Replace
 - Paint Attributes - Value: 0.2

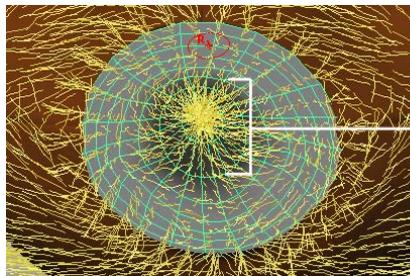
These settings set the brush to paint a relatively small, soft edged region, with each stroke of the tool as you replace the existing fur with shorter fur.



- 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.



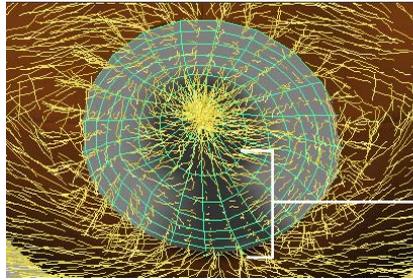
The snout surface updates to display the fur length attribute map you are currently painting. The black region at the end of the snout indicates the area you have just painted. This region of the attribute map indicates where the fur will be shorter in length when it is rendered. The lighter gray region 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.

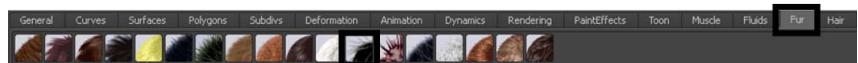
- 8 In the Paint Tool Settings editor, set the Paint Operation to Smooth, and then click Flood to apply a smooth to the entire fur length attribute map. You can click Flood more than once to apply multiple smooth operations.

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

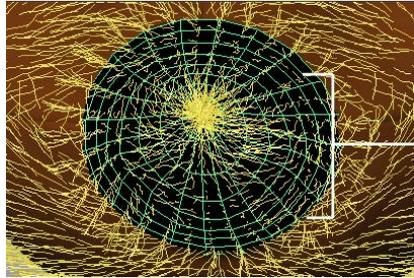
A second fur description is assigned to the bear's snout. The snout appears cluttered at this point but you'll rectify it in the following steps.

- 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.

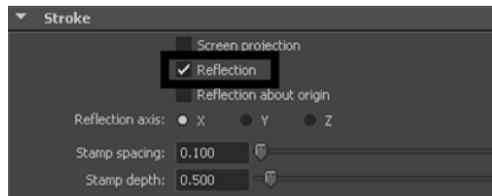
This creates a baldness attribute map for the Porcupine fur description that is black in color making the Porcupine fur feedback disappear temporarily.



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

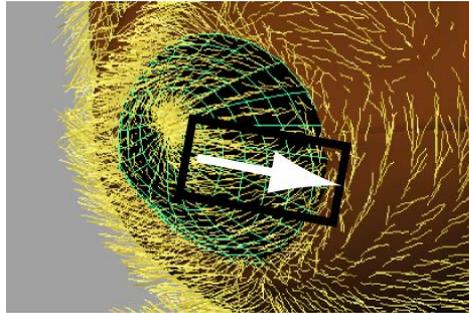
- 5 In the Paint Tool Settings editor, set the following:

- 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

- Select Fur > Edit Fur Description > Porcupine.
The Attribute Editor appears and displays the attributes for the Porcupine fur description.

To rename the fur description

- 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.

The bear's snout
with shorter fur
and whiskers



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

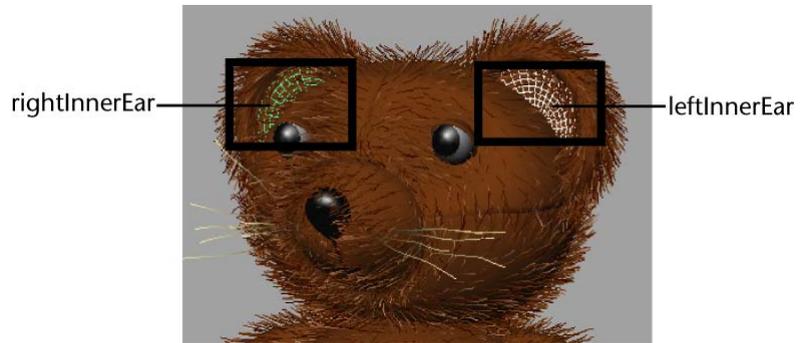
- 1 Display the Channel Box/Display Layer Editor.
- 2 In the Display Layer Editor, click the box that displays the letter R until the box is empty.

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 it's 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

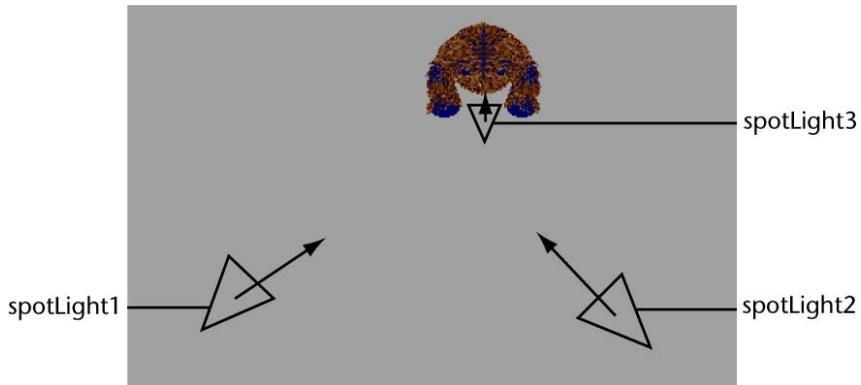
To ensure the lesson works as described, complete the steps in lesson 1, or do the following before beginning:

- ▶ Open the scene named `FurRender.ma` that is located in the `GettingStarted/Fur` 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: 0.5A 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 mental ray shadow maps 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 add shadowing for the spotlights.

In this tutorial, you will add shadows to your scene by generating shadow maps and rendering using the mental ray for Maya renderer. For more information about shadow maps, see mental ray shadow maps in the **Rendering** guide.



The scene rendered without shadowing defined.

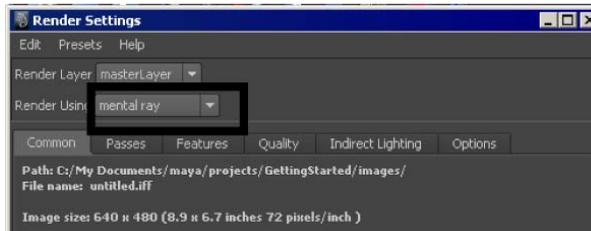
The lack of fur shadows makes the fur appear flat and unrealistic.

To turn on the mental ray for Maya renderer

- 1 In the main menu, select Window > Rendering Editors > Render Settings (or click the Render Settings icon in the Render View window).



- 2 In the Render Settings window, set Render Using setting to mental ray.

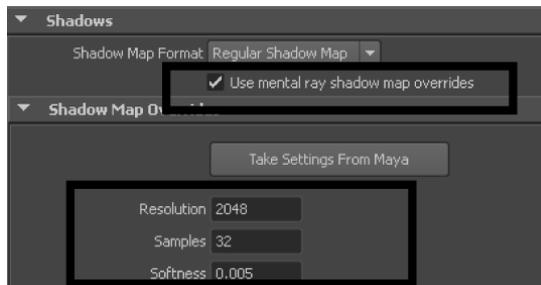


To add fur shadowing attributes to spotlights

- 1 Select Window > Outliner to display the Outliner.

For complex scenes, the Outliner is 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.

- 2 In the Outliner, click spotLight1 to select it in the scene.
- 3 Open the Attribute Editor by clicking the Show/Hide Attribute Editor icon on the Status Line.
- 4 Select the spotLightShape tab, and under the mental ray section, expand the Shadows section.
- 5 Enable Use mental ray shadow map overrides to customize the shadow maps for the spotlight.
- 6 Under the Shadow Map Overrides section, set the following attributes:
 - Resolution: 2048
 - Samples: 32
 - Softness: 0.005



- 7 Enable Use mental ray shadow map overrides for spotLight2 and spotLight3 and set the Shadow Map Overrides as follows:
 - Resolution: 1024
 - Samples: 16
 - Softness: 0.05

Rendering the scene

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

To render the scene

- To render the fur, do one of the following:
 - Select Render > Render Current Frame.

- In the Status Line, click the Render Current Frame button  .

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



Note: Final results may vary.

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

Beyond the lesson

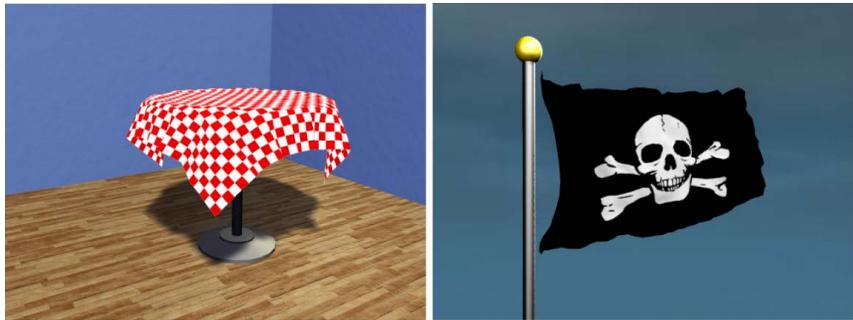
In this lesson you learned how to:

- Add spotlights to the scene—A three light setup is a good starting point for many renderings.

- Add mental ray shadow maps to lights— Generate mental ray shadow maps 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.

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 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:

- [Lesson 1: Creating nCloth collisions](#) on page 897
- [Lesson 2: Creating nCloth constraints](#) on page 912
- [Lesson 3: Creating nCloth Clothing](#) on page 926

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](#) on page 25 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:

- 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

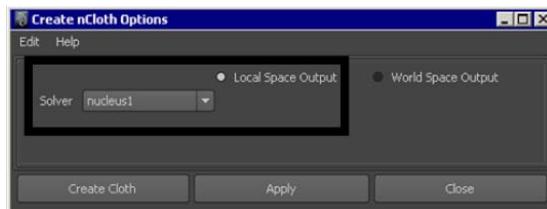


The first step in the nCloth creation process is to convert the polygon object you want to behave like cloth to an nCloth object.

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.

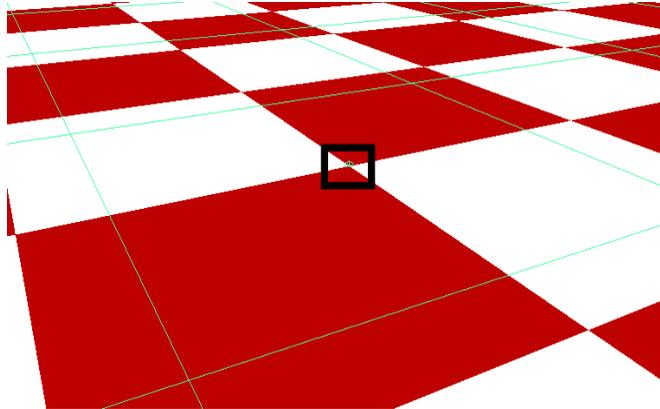
Selecting a solver determines which Maya Nucleus system the nCloth belongs to.

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.

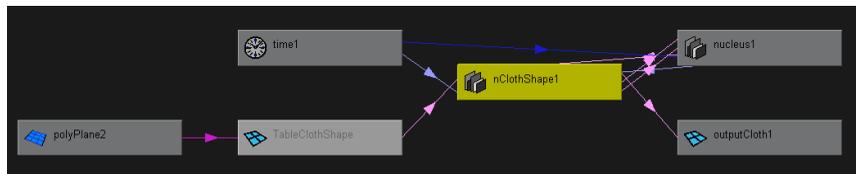
The static quad polygon table cloth is automatically tessellated and converted to a dynamic nCloth object, and an nCloth handle appears on your table cloth in the scene view. The nCloth handle resembles a small wireframe sphere, and it is located at the center of table cloth. You can use the table cloth's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

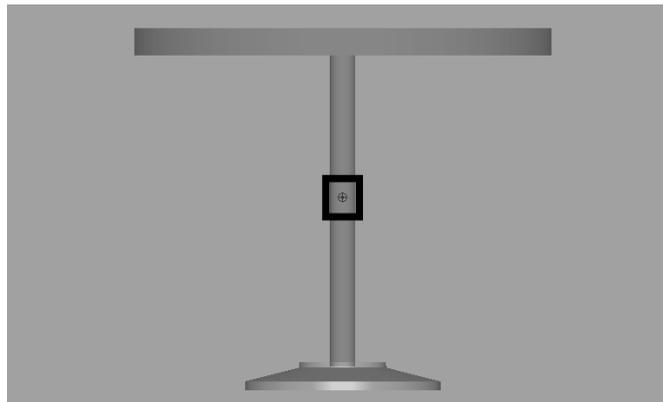
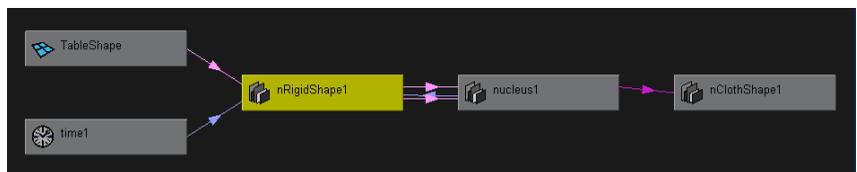


Table as viewed in X-Ray Shading mode.

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.



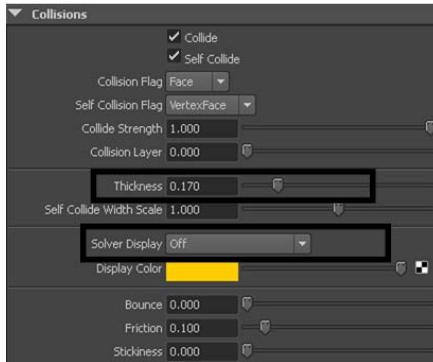
Table cloth's simulation with passive object collisions.

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.

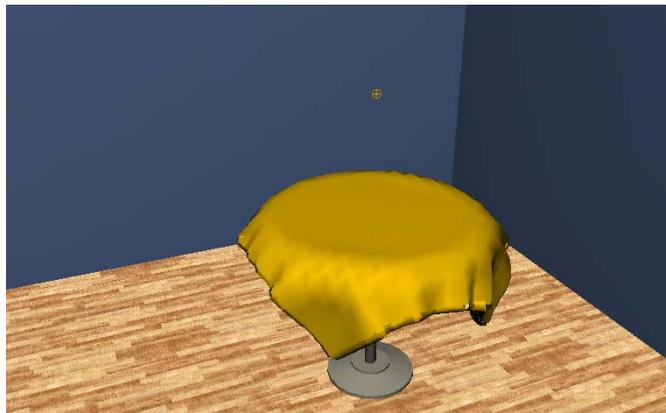


Table cloth Solver Display: Collision Thickness

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.

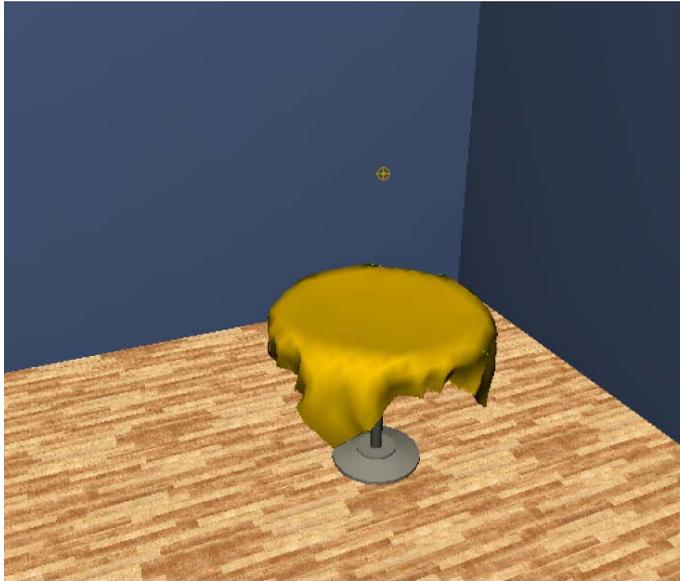


Table cloth Solver Display: Collision Thickness

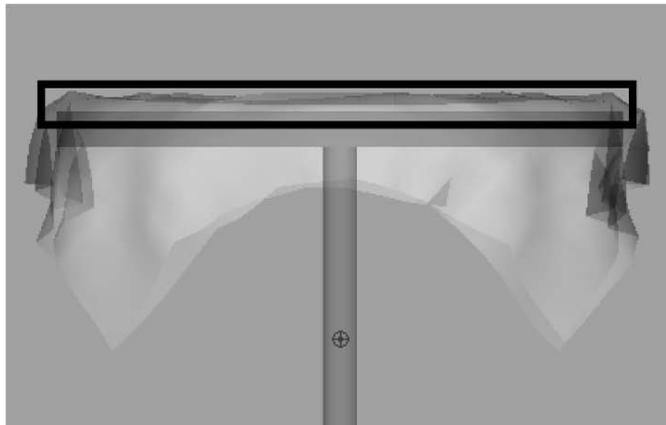


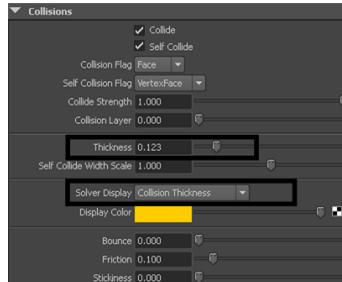
Table cloth and table as viewed in X-Ray Shaded mode.

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.

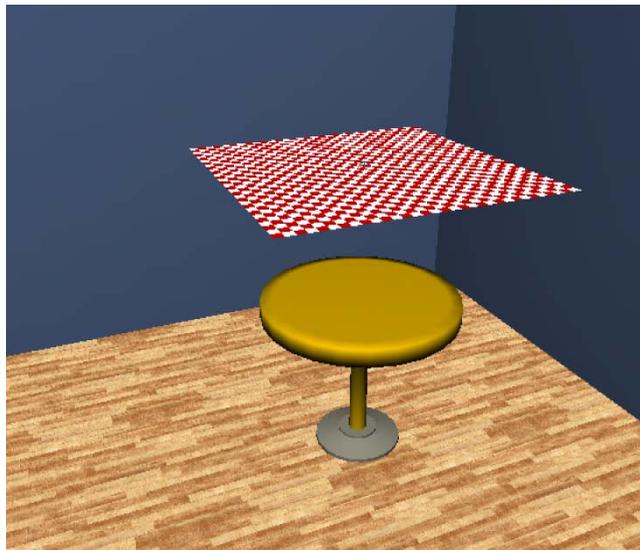


Table Solver Display:
Collision Thickness

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

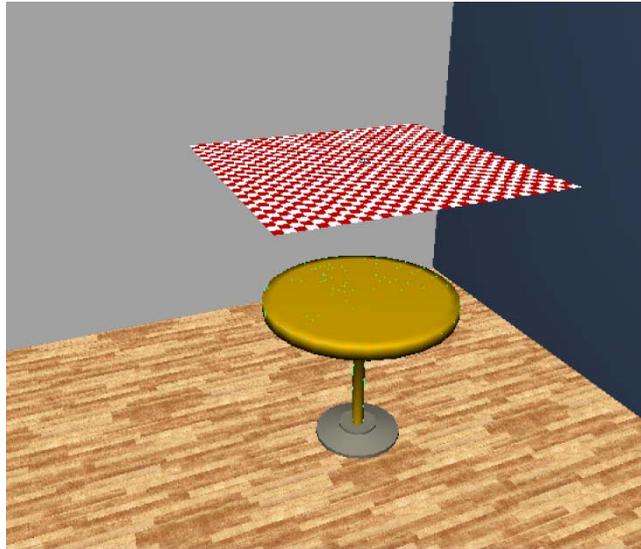


Table Solver Display:
reduced Collision
Thickness

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.



Table cloth's improved collisions.

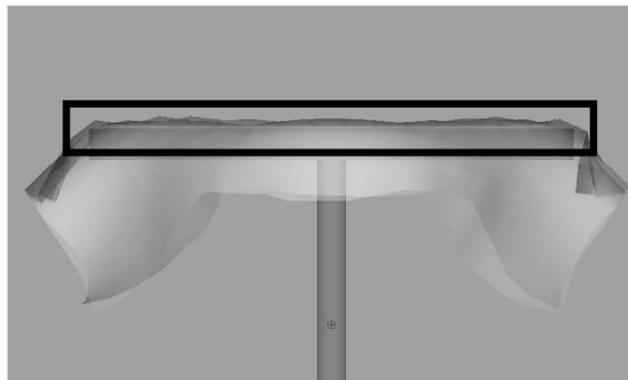


Table cloth and table as viewed in X-Ray Shaded mode.

nCloth table cloth and table passive object improved collisions.

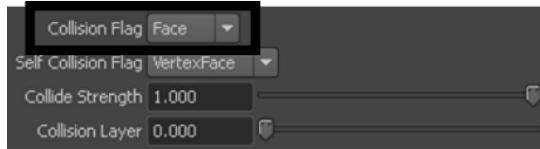
Adjusting the Thickness of the table cloth nCloth and the table passive object dramatically improved the accuracy of the table cloth's collisions.

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.

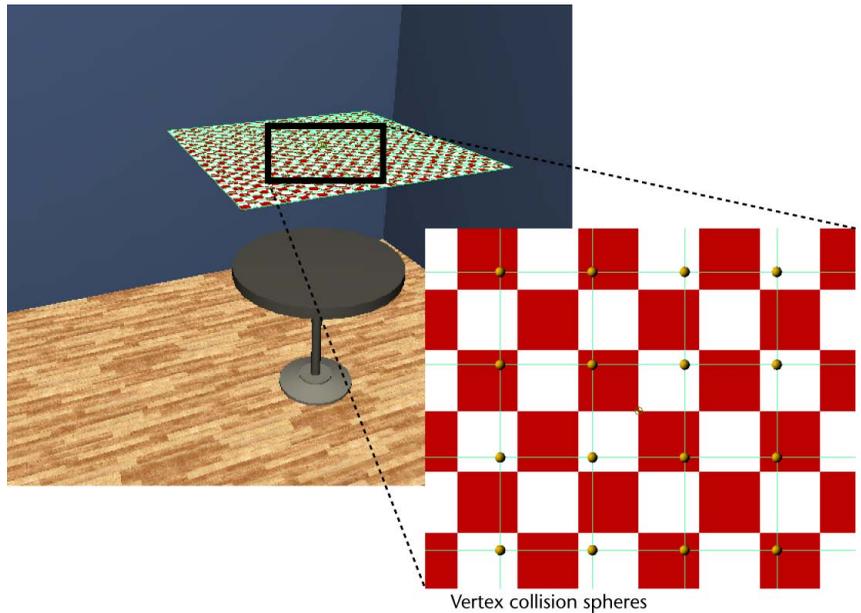
To set which nCloth components participate in collisions

- 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.

Parts of the table cloth are not properly colliding with the table. 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:

- 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.

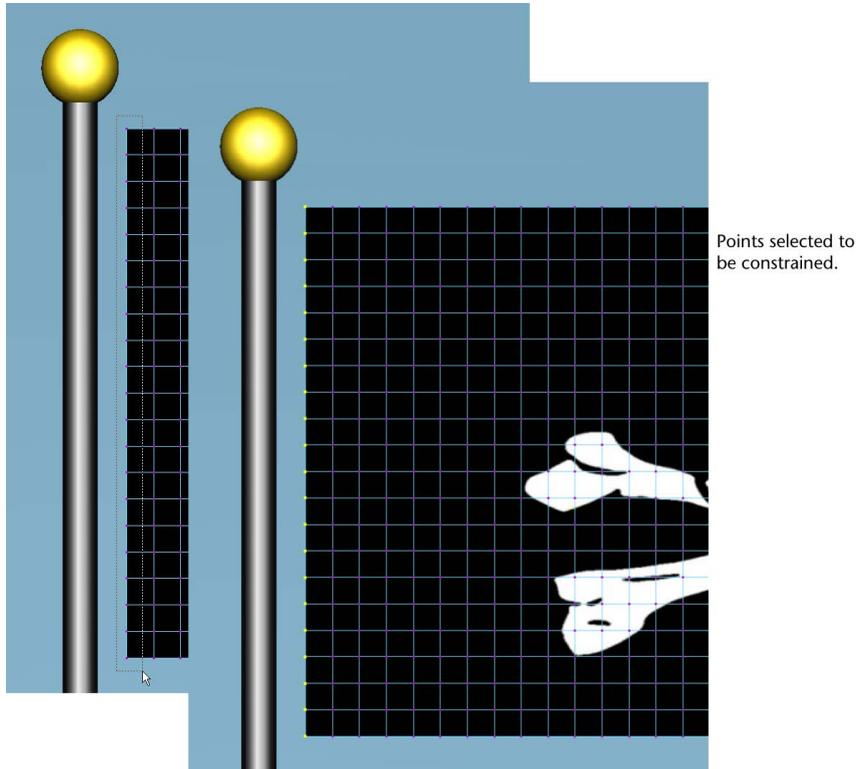


The flag falls downward through the air. For the flag to retain its position beside the flag pole in XYZ space, it needs to be constrained to the static flag pole passive object.

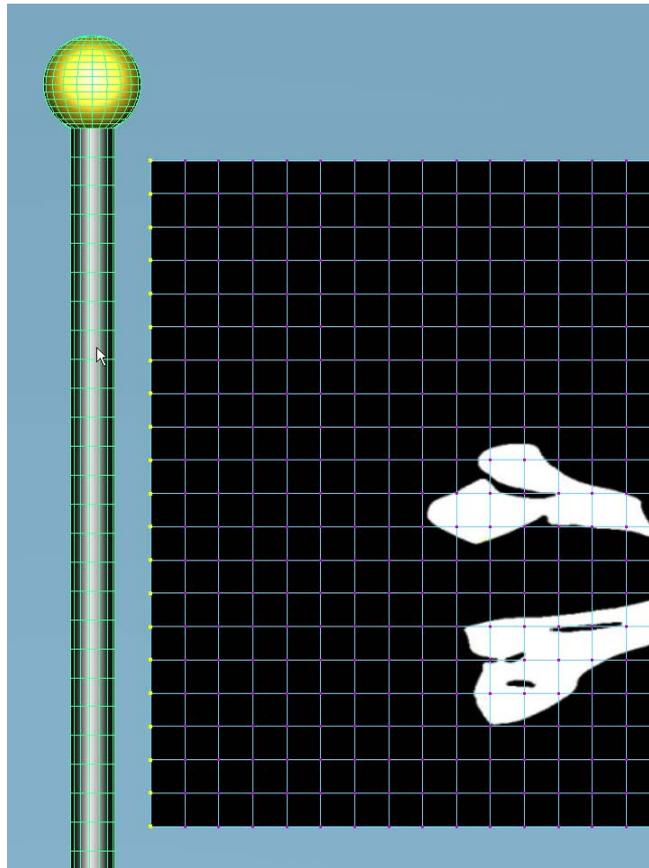
NOTE You may need to increase the total number of frames in the timeline.

- 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.

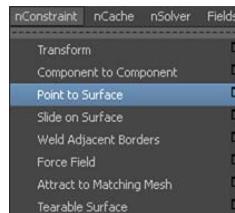


- 7 Shift-select the flag pole passive object to include it in your selection.

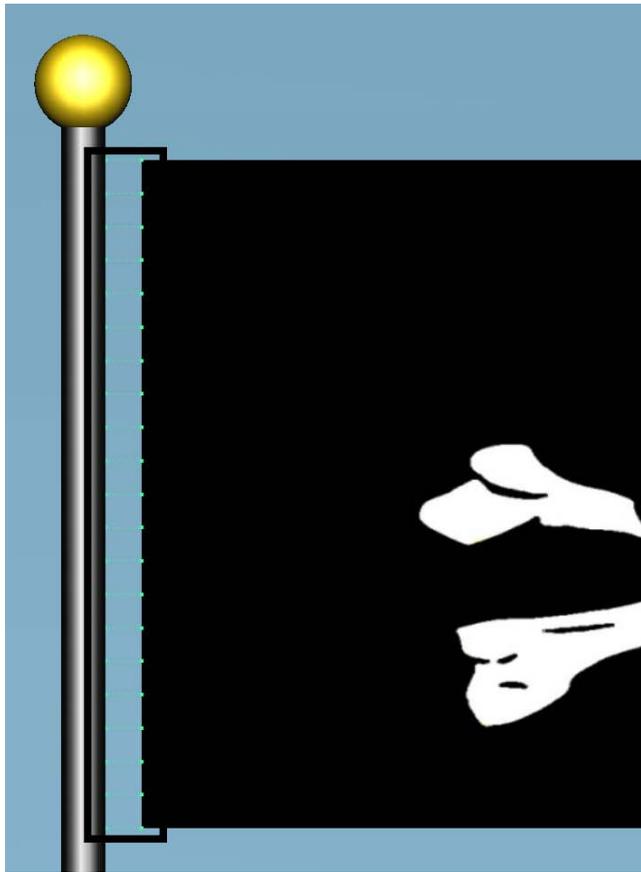


Object selected for the flag points to be constrained to.

8 Select nConstraint > Point to Surface.



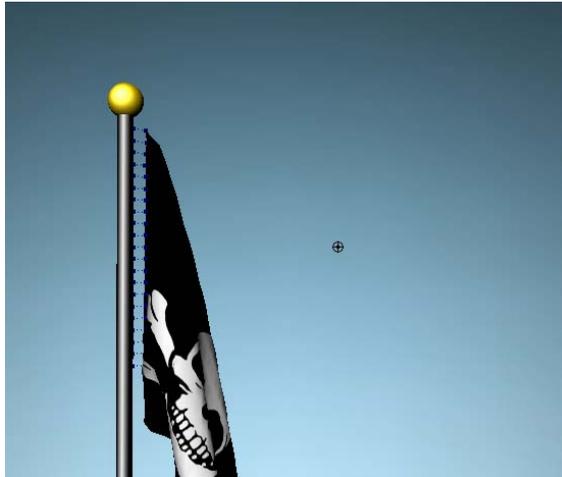
A Point to Surface constraint is created, and constraint links appear between the vertices you selected on the nCloth flag, and the surface of the flag pole passive object.



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.



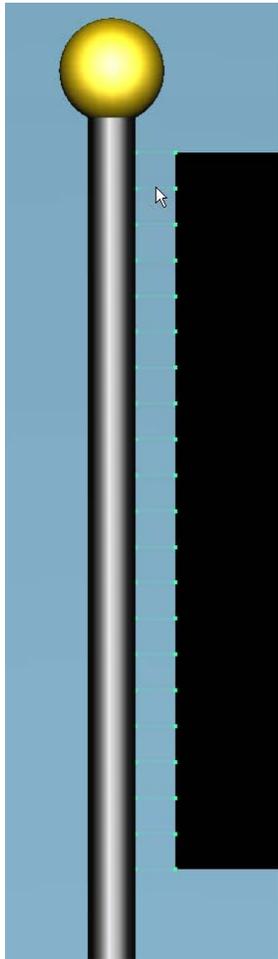
- 10 Go to the beginning of the playback range.

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.

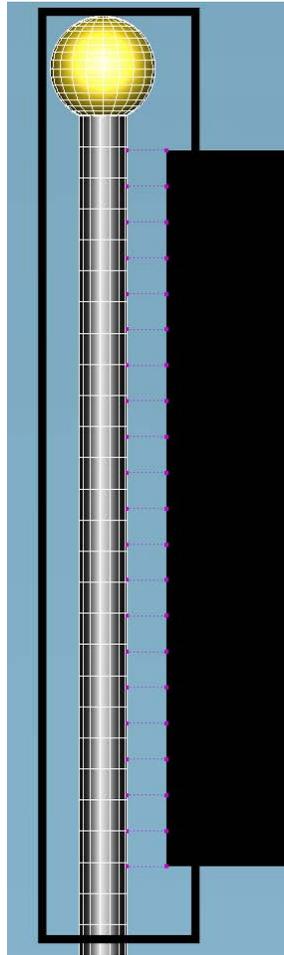


Selected Point to
Surface constraint links.

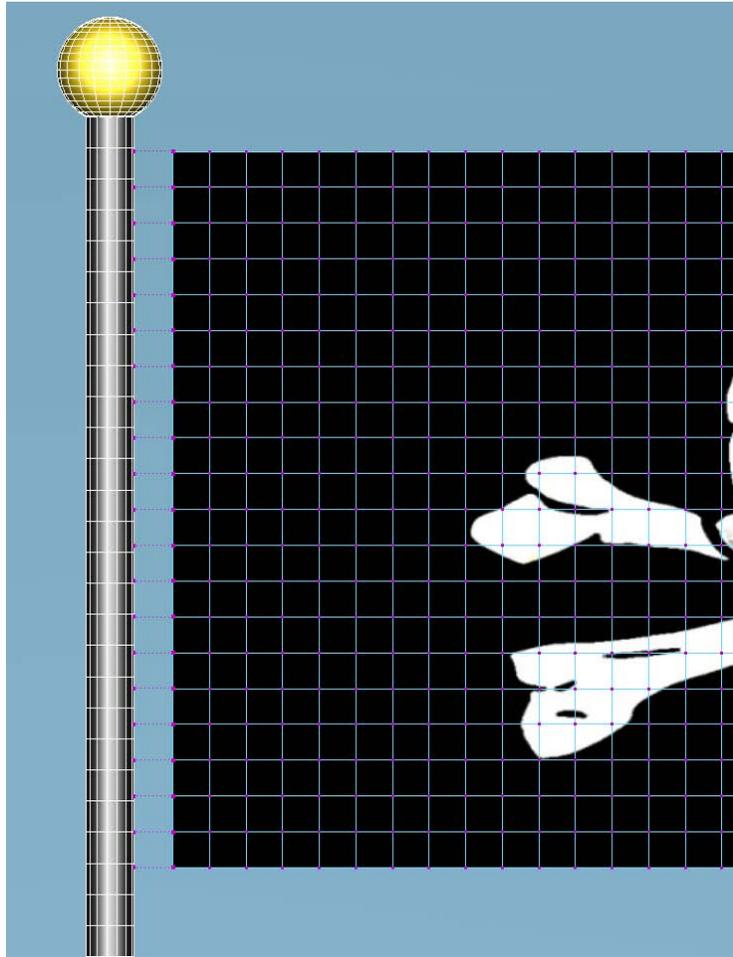
- 3 Select nConstraint > Select Members.



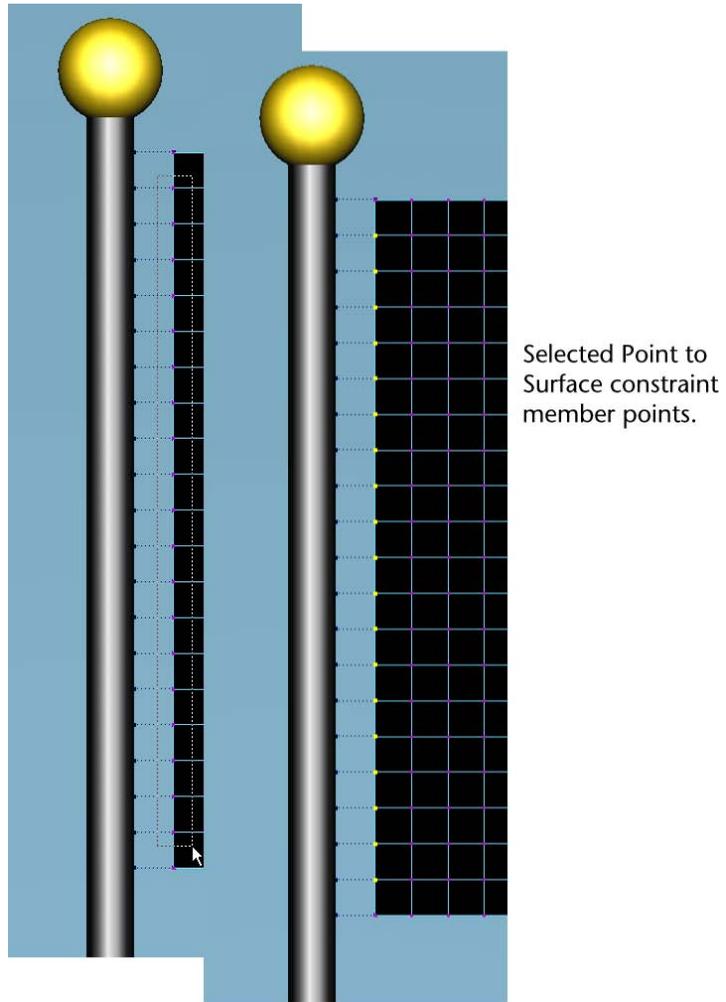
The flag *points* and the flag pole *surface* that are members of the selected Point to Surface constraint highlight in the scene view.



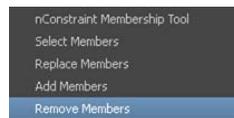
- 4 Right-click the flag and select Vertex from the context-sensitive menu that appears.
The flag switches to vertex display mode.



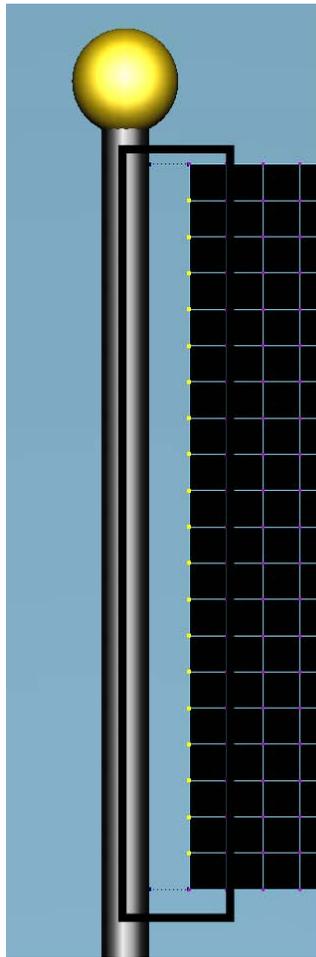
- 5 Select all the flag points, except the top most and bottom most points.



6 Select nConstraint > Remove Members.



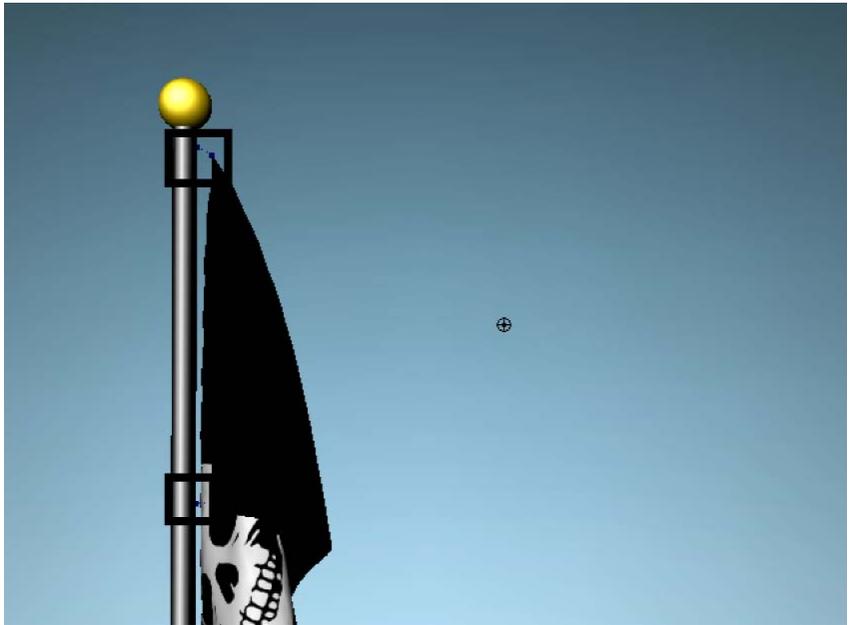
The selected constrained flag points are removed from the constraint and their constraint links disappear from the scene view.



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

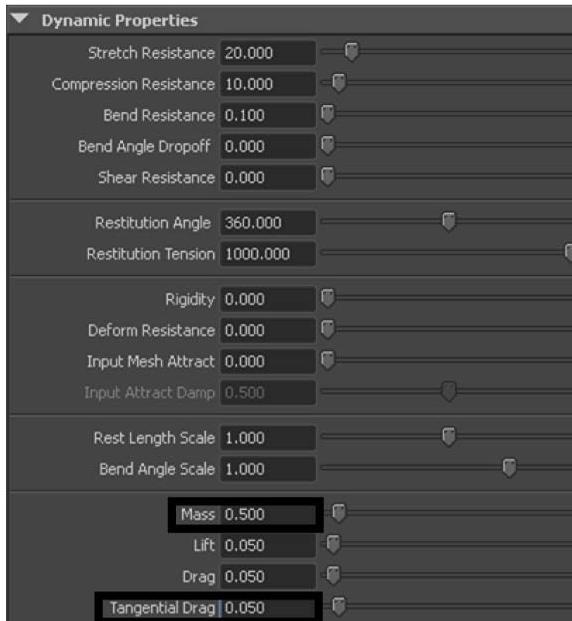
- 1 Select the flag.
- 2 Open the Attribute Editor and select the nClothShape tab.
- 3 In the Dynamic Properties section, decrease Mass and increase Tangential Drag.



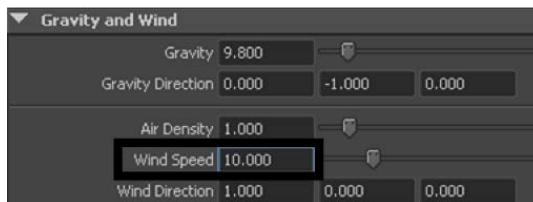
nCloth flag simulation
with:
Mass = 0.5
Tangential Drag = 0.05
Wind Speed = 10

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:

- 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.

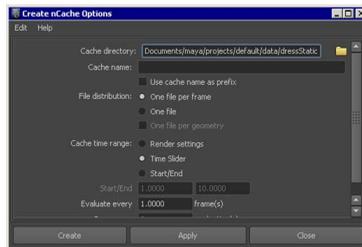


Caching nCloth to speed up playback

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 nCache Options window appears.



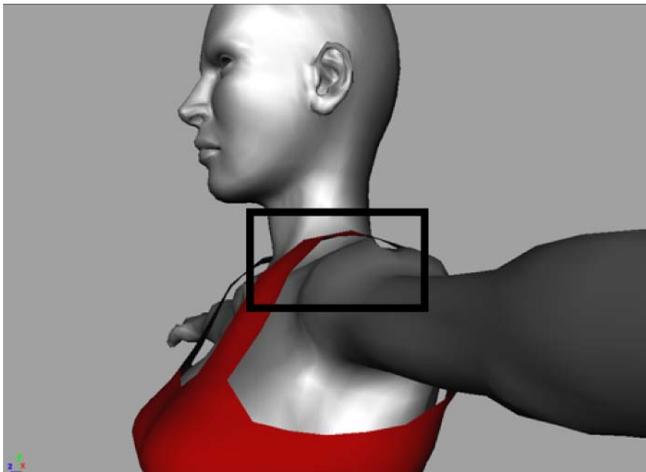
- 3 Set the Cache directory to the folder to which you want your caches saved.
- 4 Set the Cache name to DressCache.
- 5 Click Create.

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.

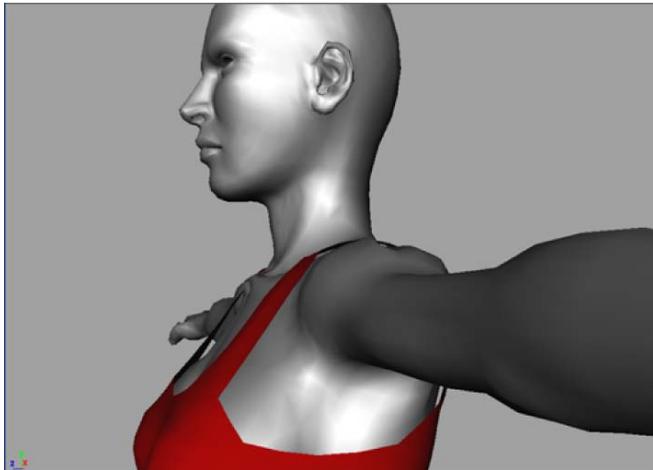
To adjust the collision thickness of the dress

- 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 in the nDynamics guide.
- 4 Select the nClothShape1 tab and in the Dynamic Properties section set Mass to 0.7.
By lowering the mass of the cloth, Maya will simulate it as a lighter fabric such as silk.
- 5 In the Dynamic Properties section set Stretch Resistance to 140 and Compression Resistance to 120.
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.

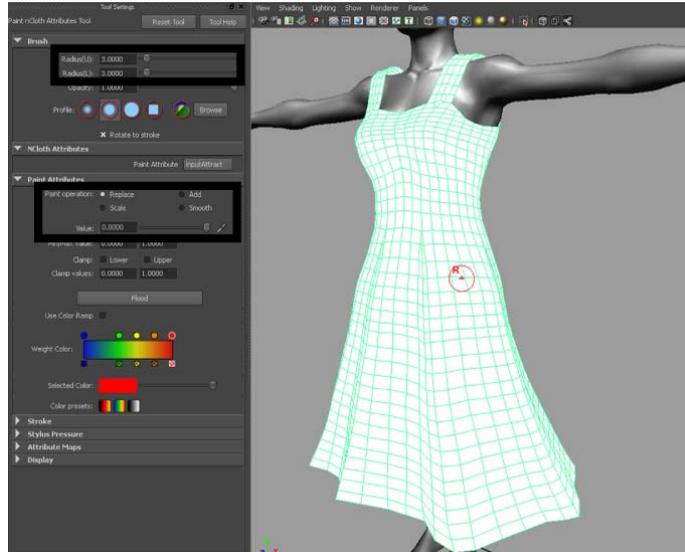
To paint Input Attract

- 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.

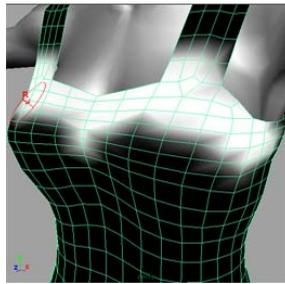
- 11 Click the Flood button.

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.

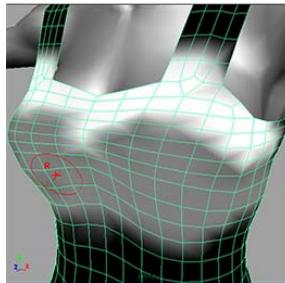
- 13 Using the Tool brush, paint around the top of the bodice.

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.



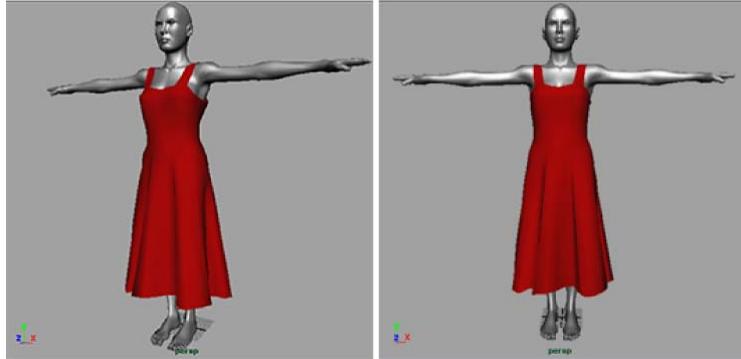
- 16 Dolly and tumble to ensure that you have painted the entire bodice.

- 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:

- 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.



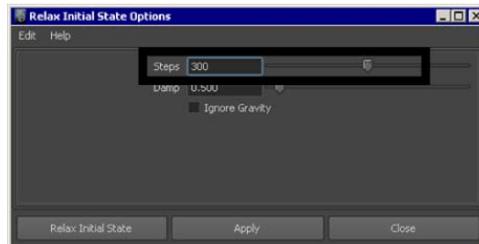
- 2 Go to the beginning of the playback range.

- 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.

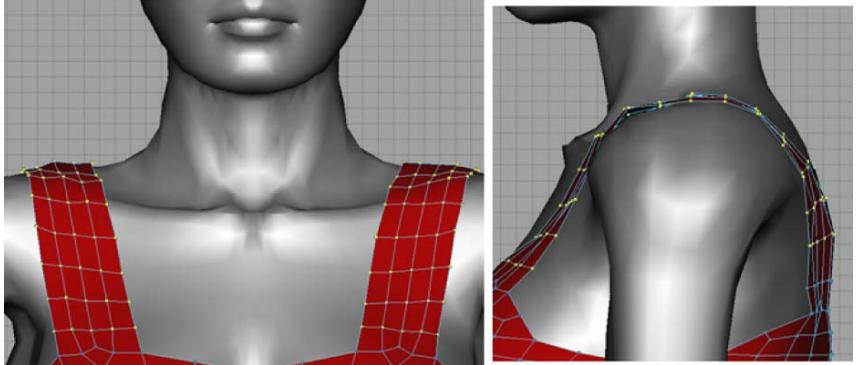
Constraining nCloth clothing



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.

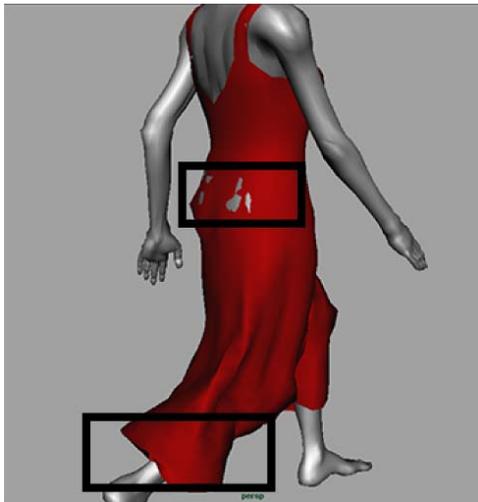
NOTE If you de-selected the constraint and have trouble re-selecting it, you can select Window > Outliner to open the Maya Outliner. You can then select the constraint by left-clicking dynamicConstraint1 in the Outliner.

- 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.



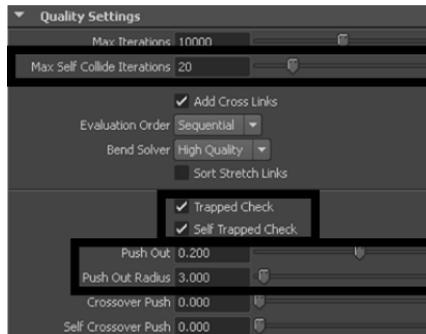
The mesh of the character shows through the dress.

The dress bunches undesirably.

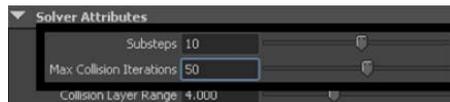
To improve the quality of the nCloth collisions

- 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, do the following:
 - Set Max Self Collide Iterations to 20.
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.
 - Turn on Trapped Check.
 - Turn on Self Trapped Check.
Turning on Trapped Check and 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.
 - Set Push Out to 0.200.
Setting Push Out applies a force that pushes out objects that are intersecting or interpenetrating, to the nearest point on the current nCloth object's surface.
 - Set Push Out Radius to 3.0.

Setting Push Out Radius specifies the maximum distance from the surface of the current nCloth object that the Push Out attribute affects.



- 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 look 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.

Introduction



nParticles is a particle generation system that uses Maya® Nucleus™, the same dynamic simulation framework that generates nCloth simulations. An nParticle object natively collides and interacts with other nParticle objects, as well as with nCloth and passive collision objects. nParticles can also self-collide, meaning that particles from the same nParticle object can collide with each other.

Nucleus dynamics allows you to create particle effects and dynamic simulations that cannot be achieved with Maya classic particles. For example, you can use Liquid Simulation attributes to create particle systems that behave like liquids. You can also create effects in which particles interact and drive nCloth animations and deformations.

This chapter includes these tutorials:

- [Lesson 1: Creating nParticles](#) on page 947
- [Lesson 2: Creating a smoke simulation with nParticles](#) on page 959
- [Lesson 3: Creating a liquid simulation with nParticles](#) on page 981

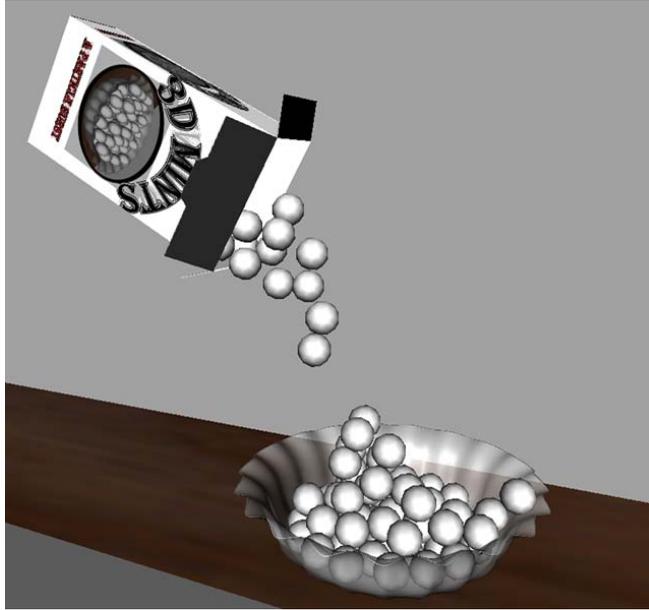
Preparing for the tutorials

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](#) on page 25.
- 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 nParticles

Introduction



In this lesson, you will be introduced to the basic concepts of Nucleus dynamics, as well as learn how to create an nParticle and emitter object. You will then adjust nParticle object attributes to create a simple nParticle effect. In this lesson, you also learn how to:

- Create passive collision objects for your nParticles to collide with.
- Set nParticle size.
- Adjust Nucleus node attributes.
- Adjust attributes to control the behavior of your nParticles.

Lesson setup

To ensure this lesson works as described, open the scene file named `nParticle_Candy.mb`. This file is in the `GettingStarted` directory that you set as your Maya project:

```
GettingStarted/nParticles/nParticle_Candy.mb
```

This scene includes textured polygon models including a desk, bowl, and candy package.

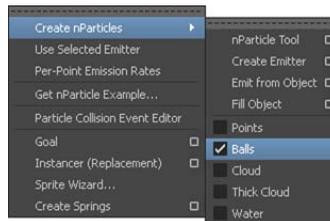
Creating an nParticle system

Before creating an nParticle system, you must decide which nParticle style and creation method suits your nParticle effect. In this lesson, nParticles are used to simulate candies that are poured from a package into a bowl.

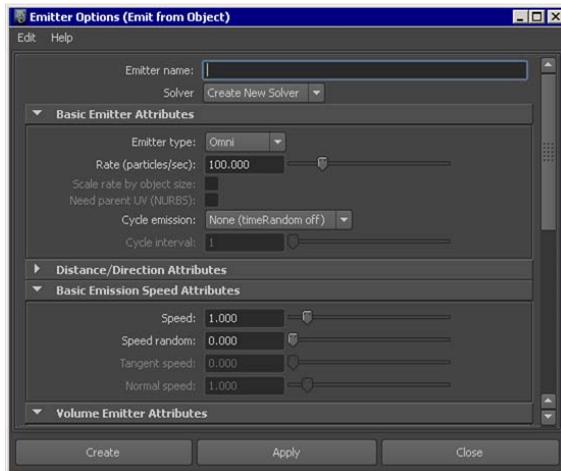
To achieve the appropriate look and behavior, you will use Ball style nParticles. nParticle styles are like templates that provide preset attribute settings for the nParticle object's appearance, size, render type, and collision properties. To create the nParticle system, you will emit nParticles from the surface of a polygon plane that is located inside the candy package.

To create an nParticle system

- 1 Select `nParticles > Create nParticles > Balls`.



- 2 In the Outliner, select `candy_emitter_surface`, then select `nParticles > Create nParticles > Emit from Object >` .



The Emitter Options (Emit from Object) window appears.

- 3 In the Emitter Options (Emit from Object) window, select Edit > Reset Settings.
- 4 For the Emitter name, type *Emitter_Candy*.
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.
This specifies that your nParticle object belongs to a new Nucleus solver.
- 6 From the Emitter type list, select Surface.
A Surface emitter will randomly distribute nParticles from positions on or near the surface of the polygon plane that is inside the candy package.
- 7 Set Rate (particles/sec) to 50.
This determines the number of particles that are emitted.
- 8 In the Basic Emission Speed Attributes section, set Speed to 5.0.
- 9 Click Create.
- 10 From the perspective window menu, select Shading > Smooth Shade All.
- 11 Play your nParticle simulation.
nParticles are emitted, but they fall through the side of the candy package and disappear. This is because the nParticles cannot yet collide with any of the other objects in your scene. For your nParticles to collide with the

candy package and bowl, these meshes need to be passive collision objects and members of the same Maya Nucleus solver system as the nParticles.

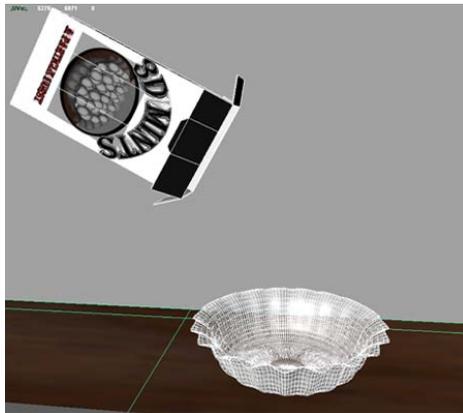
Making nParticles collide with their environment

In the second step of creating an nParticle effect, you convert the meshes you want the nParticles to collide with to passive collision objects. When nParticles collide with passive collision objects, their behavior, speed, and direction change. Passive collision objects are not affected by collisions with other objects.

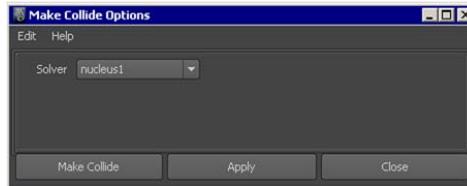
In this lesson, you convert the candy package, bowl, and the desk surface geometry to passive collision objects. The nParticles can then collide with these objects.

To convert geometry to passive collision objects

- 1 Rewind the simulation to the start frame.
- 2 In the scene view, Shift-select the package and bowl meshes as well as the surface of the desk.



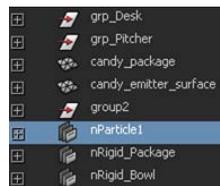
- 3 Select nMesh > Create Passive Collider >  .
The Make Collide Options window appears.



- 4 From the Solver list, select nucleus1.
- 5 Click Make Collide.

The candy package, bowl, and desk surface meshes are converted to passive collision objects. nRigid handles appear at the center of the meshes in the scene view. You can use the nRigid handles to quickly select nRigidShape nodes in the Hypergraph or their tabs in the Attribute Editor. These objects are now members of the same Nucleus system (nucleus1) as the nParticles.

- 6 To rename the nRigid objects, in the Outliner, double-click them one at a time and type *nRigid_Package*, *nRigid_Bowl*, and *nRigid_Desk* respectively. Press Enter after you type each object name.



- 7 Play the simulation.

When you play the simulation, the nParticles now fall from the end of the package and are collected in the bowl.

Notice that the nParticles are not falling from the candy package as quickly as you might expect. Instead, they fall slowly, as if the simulation was playing back in slow motion. This is because the Nucleus solver properties need to be adjusted to suit the simulation.

In the next section of the lesson, you set Nucleus solver properties so that the particles fall realistically.

nParticle and Nucleus nodes

You have now created all of the objects for your Nucleus dynamics system. The following section describes each Nucleus system node.

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_Candy1 is the nParticle emitter node, which carries all the particle emission attributes such as particle emission rate and emitter type.

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

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

Setting Nucleus Space Scale

Adjusting the Nucleus solver properties is an important step of creating an nParticle effect. The Nucleus solver properties control internal forces that affect all of the nodes that are members of a solver system. One of the most significant Nucleus solver attributes is Space Scale.

The Nucleus solver applies Gravity and Wind forces to objects as if the objects were scaled in meters, regardless of the units in which your scene was created. To maintain the effect of real world gravity, you need to adjust the Nucleus Space Scale to suit your simulation.

To set Space Scale

- 1 In the Outliner, select *nParticle1*.
- 2 In the Attribute Editor, click the nucleus1 tab.
- 3 In the Scale Attributes section, set the Space Scale to 0.01.



When you reduce the Space Scale value, Maya evaluates the objects in the simulation as if they were much smaller in size. This results in a visibly increased gravitational affect on the nParticles.

- 4 Play your simulation.
When you play the simulation, notice that the nParticles are now falling from the candy package at a realistic speed.

Adjusting nParticle size and color

You can adjust nParticleShape node attributes to specify how the particles look, move, and collide with other Nucleus objects. Adjusting the emitter attributes also affects the overall characteristics of your nParticle simulation.

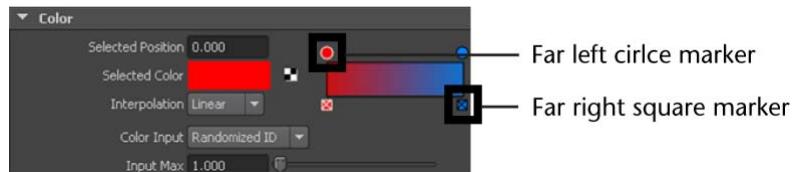
To make the nParticles look and behave like candy, you will increase their size, change their color, and make their surfaces sticky.

To increase nParticle size

- 1 In the Outliner, select *nParticle1*.
- 2 In the Attribute Editor, click the nParticleShape tab.
- 3 In the Particle Size section, set Radius to 1.8.

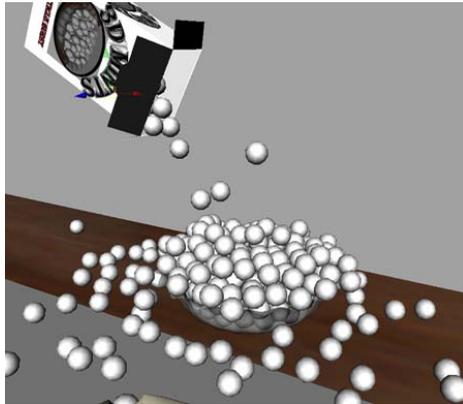
To change the color of the nParticles

- 1 In the Shading section, go to Color.
- 2 Click the far left circle marker (at Selected Position 0) of the ramp.



- 3 Click the color swatch beside Selected Color.
The Color Chooser appears.
- 4 In the Color Chooser, click the white. swatch.
- 5 Drag your mouse off the Color Chooser to close it.
- 6 Click the far right square marker of the Color ramp to delete it.
For this simulation, the color of the nParticles will not change so you only need one ramp marker.
- 7 From the Color Input list, select Constant.
- 8 Play the simulation.
The nParticles now look more like candy, but too many nParticles are being emitted, which is overflowing the bowl.

To solve this problem, you can reduce the number of nParticles by adjusting the emitter attributes. Like many Maya attributes, you can set a key to change the emission rate at a specified time in the simulation.



To key nParticle emission rate

- 1 Go back to the beginning of the playback range.
- 2 Play the simulation and stop at frame 50.
- 3 In the Outliner, expand *candy_emitter_surface* and select *Emitter_Candy1*.
- 4 In the Attribute Editor, click the *Emitter_Candy* tab.
- 5 In the Basic Emitter Attributes section, right-click Rate (Particles/Sec) and select Set Key from the shortcut menu.
A key is created with the nParticle emission rate set at 50.
- 6 Play the simulation for five more frames stopping at frame 55.
- 7 Set Rate (Particles/Sec) to 0.
- 8 Right-click Rate (Particles/Second) and select Set Key from the shortcut menu.
A key is created with the nParticle emission rate set at 0.
- 9 Go back to the beginning of the playback range and play the simulation.
nParticles are now emitted until frame 55, which is just enough to fill the bowl.

Adjusting nParticle collision attributes

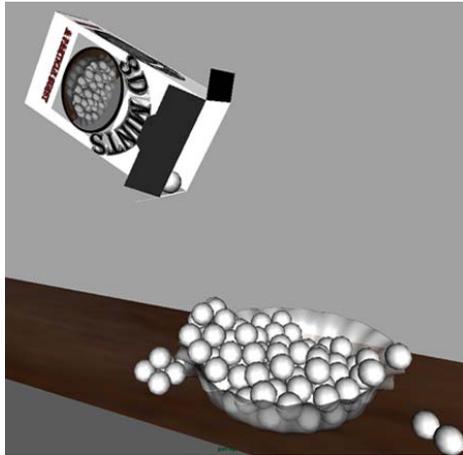
nParticle Collisions attributes determine the behavior of nParticles when they collide each other and with other Nucleus objects. In this section of the lesson, you make the nParticles sticky like candy by adjusting the Friction and Stickiness attributes.

To adjust nParticle Collisions attributes

- 1 In the Outliner, select *nParticle1*.
- 2 In the Attribute Editor, click the nParticleShape tab.
- 3 In the Collisions section, set the following:
 - Bounce: 0
 - Friction: 0.2
 - Stickiness: 0.5
- 4 Play your simulation.
- 5 Notice that the nParticles are now sticking each other and piling up at the edge of the bowl.

Friction and Stickiness are similar attributes in that Friction is a force acting in the tangent direction, while Stickiness is an adhesion force in the normal direction. When Nucleus objects collide, the affect of their Friction and Stickiness values are additive. For example, if the Stickiness value of the nParticles and the bowl are 0.5, the overall Stickiness acting on both objects is equal to a value of 1.0.
- 6 To set Friction and Stickiness on the bowl, in the Outliner select *nRigid_Bowl*.
- 7 Click the nRigid_BowlShape tab, then in the Collisions section, set Friction to 0.2 and Stickiness to 0.5.
- 8 Play your simulation.

Notice that the nParticles have greater tendency to stick to the surface of the bowl. This is because of the additive effect of the Friction and Stickiness attributes.



For more information see Friction and Stickiness in the nDynamics guide.

- 9 Set the nParticle Stickiness 0.2.

The particles now stick to one another and to the bowl like candies.

Adjusting nParticle self collisions

When you play the simulation, the nParticles are piling up as they fill the bowl. This is because the nParticles are self-colliding with each other. For some nParticle effects, such as liquid simulations, lava flows, and splatters, you do not want the nParticles to self-collide. You can turn nParticle self-collisions on and off by setting the Self Collide attribute.

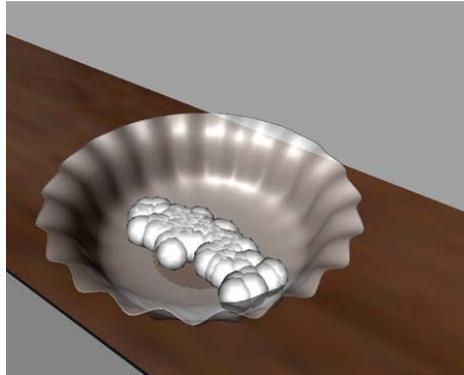
To turn off nParticle self-collisions

- 1 In the Collisions section of the nParticle Attribute Editor, turn off Self Collide.
- 2 Play your simulation.

Notice that the nParticles no longer pile up, but blend and overlap with each other.

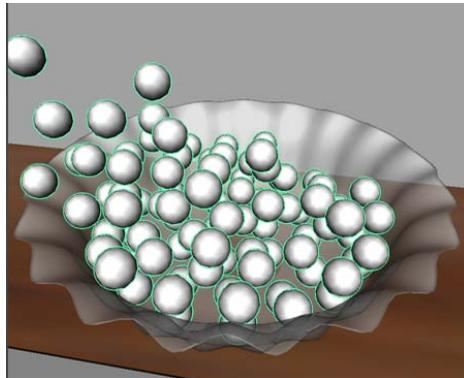
Each nParticle has an outer boundary that determines the distance between the nParticle and an object at collision. The Collide Width Scale attribute sets the collision distance between the nParticles and other Nucleus objects. The Self Collide Width Scale attribute sets the collision distance between self-colliding nParticles.

By default, Collide Width Scale and Self Collide Width Scale are set to 1 for normal collisions. Turning off Self Collide is like setting the Self Collide Width Scale to 0.



- 3 Turn on Self Collide, and set Self Collide Width Scale to 1.5.
- 4 Play the simulation.

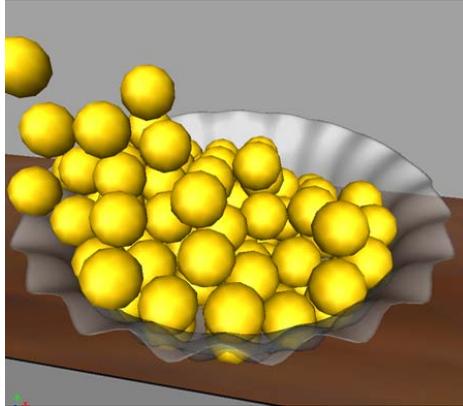
Notice that the nParticles appear to collide with each other when they fill the bowl, but they don't touch one another. The gaps between the nParticles are due to the increased Self Collide Width Scale.



In the next step, you will display the current Self Collision Width Scale by turning on the Solver Display.

- 5 From the Solver Display list, select Self Collision Thickness.

The nParticles now appear with their Self Collision Thickness displayed in yellow. You can make adjustments to the Self Collide Width Scale and see how the thickness affects the space between colliding nParticles.



- 6 Set Self Collide Width Scale back to 1.0, and the Solver Display to Off.
- 7 Play the simulation.

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 by emitting them from a polygon object using a Surface type particle emitter.
- Set the size of the nParticles using the Radius attribute.
- Set the color of your nParticles.
- Set nParticle Collisions attributes to control the behavior of nParticles.
- Turn off nParticle self-collisions.
- Set the collision distance between self-colliding nParticles by setting Self Collide Width Scale.

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

Lesson 2: Creating a smoke simulation with nParticles

Introduction



For the smoke simulation in this lesson, you create an nParticle system composed of an nParticle object, an emitter object, and a nucleus node. You will add a Volume Axis field to your effect to create turbulence and swirling in the nParticle smoke. This field helps create the realistic behavior of smoke as it cools, rises and dissipates. To finish your smoke simulation, you will also add a Nucleus Wind force to make the smoke drift as it disappears.

Lesson setup

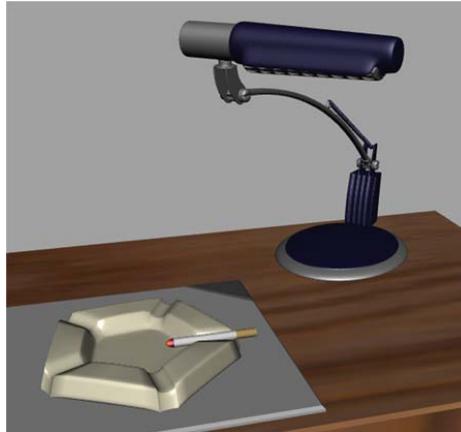
To ensure this lesson works as described, open the scene file named `Smoke_Simulation_1.mb`. This file is in the `GettingStarted` directory that you set as your Maya project:

```
GettingStarted/nParticles/Smoke_Simulation_1.mb
```

This scene includes textured polygon models including a desk, ashtray, and cigarette.

Creating an nParticle system

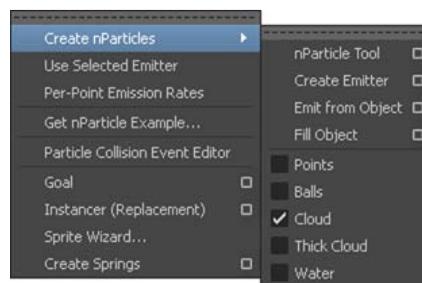
In this part of this lesson, you create an nParticle object and its emitter object. An emitter controls the position, direction, quantity and initial velocity of the emitted nParticles when they are born (emitted into the scene).



Creating an nParticle and emitter object

To create an nParticle and emitter objects

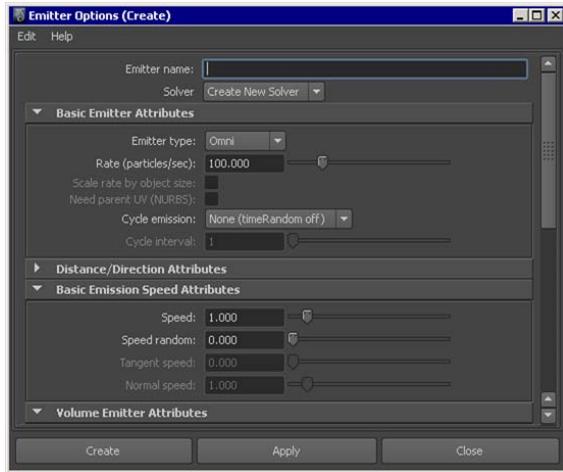
- 1 Select nParticles > Create nParticles > Cloud.



Selecting Cloud presets some nParticleShape attributes that are suitable for simulations such as smoke, dust, or fog.

- To create an nParticle emitter object, select nParticles > Create nParticles > Create Emitter >  .

The Emitter Options (Create) window appears.



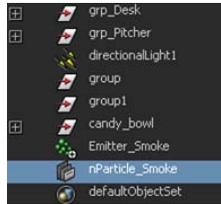
- In the Emitter Options (Create) window, select Edit > Reset Settings.
- In the Emitter name field, type *Emitter_Smoke*.
Naming your emitter object makes it easier to identify in the Outliner when you want to select it and make attribute adjustments.
- From the Solver list, select Create New Solver.
This specifies that your nParticle object belongs to a new Nucleus solver.
- In the Basic Emitter Attributes section, set the following:
 - Emitter type: Directional.
 - Rate (particle/sec): 120
This sets the number of particles that are emitted into the scene. The higher the emission rate the thicker the smoke appears.
- In the Basic Emission Speed Attributes section, set the following:
 - Speed: 2.5.
 - Speed random: 5.0.
When Speed random is set to a positive value, the emitter generates random speeds for each nParticle.

- 8 Click Create.

An *Emitter_Smoke1*, *nParticleShape1*, and a *nucleus1* node appear in the Attribute Editor.

- 9 To rename the nParticle object, in the Outliner, double-click *nParticle1* and type *nParticle_Smoke* then press Enter.

Renaming your nParticle object makes it easier to identify when you have more than one in your scene.



Moving the nParticle emitter

When you create an nParticle and emitter objects, they are placed at the scene origin. For this lesson, you need to move the emitter so that the particles are emitted near the tip of the cigarette.

To move the nParticle emitter object

- 1 In the Outliner, select the *Emitter_Smoke1* object.

- 2 Display the Channel Box by clicking  .

- 3 In the Channel Box enter the following values:

- Translate X: 49.3
- Translate Y: 121
- Translate Z: -27.3

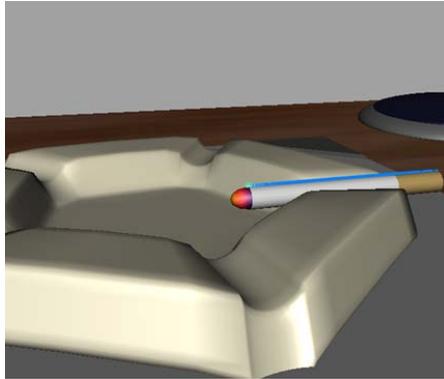
- 4 In the scene view, dolly and tumble so that you get a close-up view of the cigarette.

The nParticle emitter is now positioned near the tip of the cigarette.

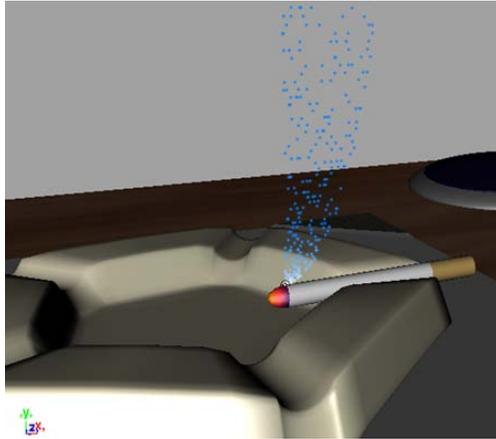
- 5 Play the simulation.

nParticles are emitted from the end of the cigarette, however, they moving along the X axis and away from the cigarette. This is because the force of the Nucleus Gravity is not yet affecting the nParticles. By default, the

Cloud type nParticles are created with Ignore Solver Gravity on, so you need to turn this option off.



- 6 In the Outliner, select *nParticle_Smoke*.
- 7 Open the Attribute Editor by clicking  .
- 8 In the Attribute Editor, click the *nParticle_SmokeShape* tab.
- 9 In the Dynamic Properties section, turn off Ignore Solver Gravity.
- 10 Play your simulation.
nParticles are emitted from the end of the cigarette, however, they are falling down and out of the scene. This is because the force of Nucleus Gravity is negative, so it pulls the nParticles down. You must change the direction of gravity to force the nParticles up.
- 11 In the Attribute Editor, click the *nucleus1* tab.
- 12 In the Gravity and Wind section, change the Gravity Direction for Y from -1.0 to 1.0.
- 13 Play your simulation.
The smoke now rises.



Editing nParticle Lifespan and Radius

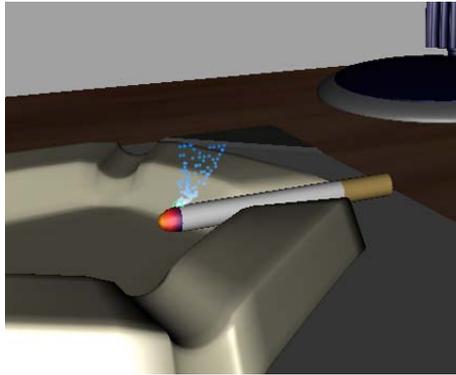
When you play the simulation, notice that the nParticles are not dissipating into the air like smoke does. This is because the nParticles lifespan is not yet defined.

In this lesson, you make the nParticles disappear from the scene after they reach a specified age by using the nParticle Lifespan attributes. You can set the nParticle object's size (Radius) according to its lifespan so that nParticles decrease or increase in size as they age.

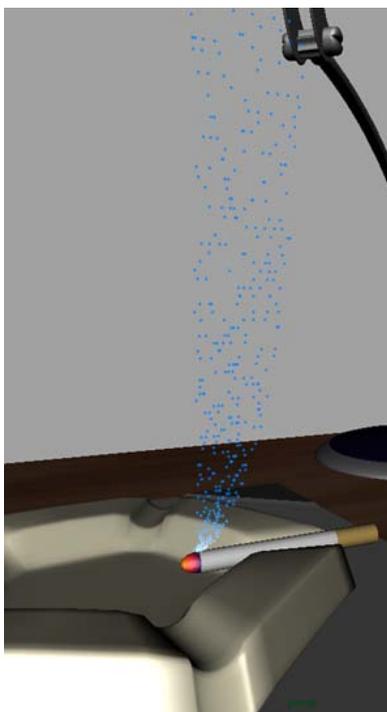
To set nParticle Lifespan

- 1 In the Outliner, select *nParticle_Smoke*.
- 2 In the Attribute Editor, click the *nParticle_ShapeSmoke* tab.
- 3 In the Lifespan section, select *Random range* from the Lifespan Mode list.
- 4 Play the simulation.

The nParticles now die (disappear from the scene) shortly after they are emitted. nParticle Lifespan is measured in seconds, meaning that at a Lifespan of 1.0, the nParticles die after one second.



- 5 Set Lifespan to 5.
- 6 To add some randomization to the nParticle's Lifespan, set Lifespan Random to 5.
- 7 Rewind the simulation and play it back.
The nParticles die off more randomly, but the nParticle object is still not dense enough to look like smoke. You can make your nParticle object denser by increasing the nParticle Radius.



For a realistic 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 nParticles as they age, which creates the effect of smoke dispersing until it gradually disappears.

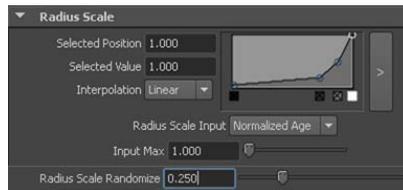
To set nParticle size

- 1 In the Particle Size section, set Radius to 5.0.
- 2 In the Radius Scale section, select the first marker in the ramp, set Selected Position to 0.016, and set Selected Value to 0.04.

Creating a new marker allows you to set two Selected Value and Selected Position values. 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.

- Click in the ramp to create additional markers and set the Selected Position and Selected Value for the new markers to the following values:

Marker	Selected Position	Selected Value
2	0.732	0.180
3	0.882	0.480
4	1.0	1.0



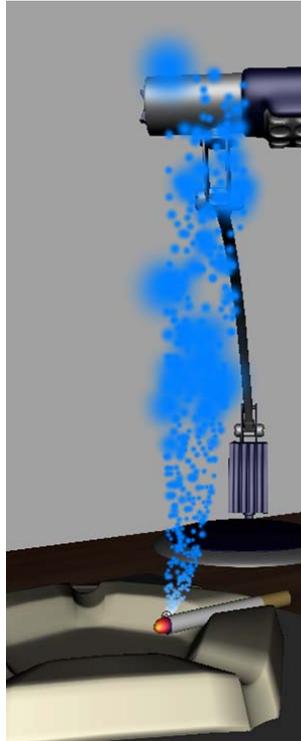
For each point in the ramp, leave Interpolation at its default setting of Linear.

- To map nParticle 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.

- To add randomization to the per-particle radius, set Radius Scale Randomize to 0.25.
- Play the simulation.

You can continue to adjust the per-particle radius by adding more markers to the ramp, or optimize the ramp values as the simulation plays back.



Open the second scene for the lesson

To ensure this lesson works as described, open the scene file named `Smoke_Simulation_2.mb`. This file is in the `GettingStarted` directory that you set as your Maya project:

```
GettingStarted/nParticles/Smoke_Simulation_2.mb
```

Adding a Volume Axis field

You can use Maya Fields to animate the motion of nParticles and create the effect of nParticles flowing around obstacles, as well as creating pluming smoke, swirling tornadoes, and rocket exhaust. For a smoke simulation, you can add a Volume Axis field to the scene to create turbulence and swirling effects in the smoke as it rises through the air.

Maya Fields are not part of an nParticle object's Nucleus system and therefore you may need to compensate for the force of Nucleus Gravity and Wind by increasing or decreasing the field's magnitude. Depending on the effect you are creating, you can also turn on Ignore Solver Gravity or Ignore Solver Wind on the nParticleShape node so that your nParticle object is only affected by Maya Fields. For more information about Maya fields see the Dynamics guide.

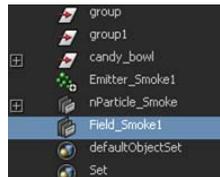
To create a Volume Axis field

- 1 In the Outliner, select *nParticle_Smoke*, then add a Volume Axis field by selecting Fields > Volume Axis >  .

The Volume Axis Options window appears.

- 2 In the Volume Axis Options window, select Edit > Reset Settings.
- 3 In Volume axis field name, type *Field_Smoke*.
- 4 Set the following Volume Axis field settings:
 - Magnitude: 90
 - Attenuation: 2.0
 - Volume Shape: Cube
 - Away From Center: 4.546
 - Directional Speed: 1.834
 - Direction: X to 0 and Y to 1.0.
 - Turbulence: 0.661
 - Turbulence Speed: 0.157
- 5 Click Create.

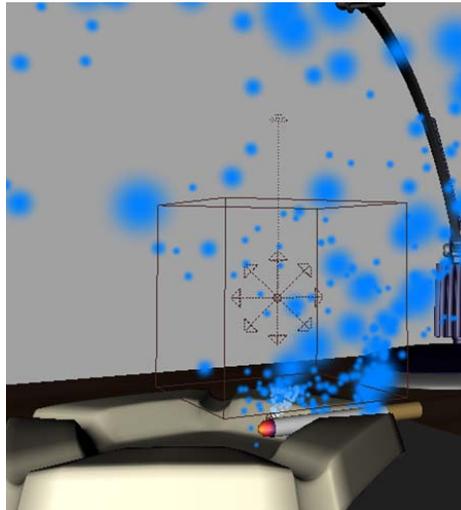
A *Field_Smoke1* node appears in the Attribute Editor and in the Outliner.



The Volume Axis field is placed at the scene origin by default. You must reposition the field so that the emitted nParticles travel through the volume.

To position the field in the path of the emitted nParticles

- 1 In the Outliner, select *Field_Smoke1* and open the Channel Box by clicking  .
- 2 In the Channel Box under *Field_Smoke1*, set the following:
 - TranslateX: 49.57
 - TranslateY: 133.8
 - TranslateZ: -26.49
 - ScaleX: 10
 - ScaleY: 10
 - ScaleZ: 10
- 3 Play the simulation.



The field creates turbulence in the nParticle smoke as it rises.

However, to create the effect of the smoke curling and dissipating, you need to adjust the field and nParticle attributes further. For example, the emitted nParticles have too much velocity for the field to influence their behavior. You will make these adjustments in the next section.

Adjusting nParticle velocity

When using Maya Fields with emitted nParticles, some particles move too quickly through the field to produce the desired effect. You can control how much of an nParticle object's velocity is retained from frame to frame by adjusting the Conserve attribute. You can further control the velocity of the particles as they enter the Volume Axis field by increasing the nParticle object's Drag. Drag adds friction between the air and nParticles, which slows the particles down.

To adjust nParticle velocity for the field

1 In the Outliner, select *nParticle_Smoke*.

2 Open the Attribute Editor by clicking  .

3 In the Dynamic Properties section of the *nParticle_SmokeShape* Attribute Editor, set Conserve to 0.05 and Drag to 0.149.

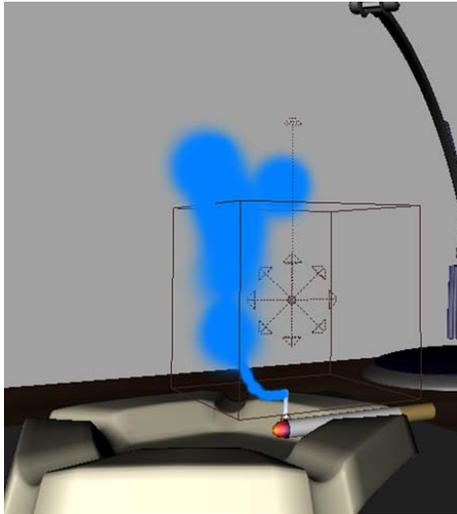
4 Play your simulation.

The particle smoke is still not rising as expected because now it is not forced up enough by the Nucleus Gravity. Maya Nucleus solver applies Gravity to objects as if the objects were scaled in meters, regardless of the units in which your scene was created. Therefore, to maintain a real world force of gravity acting on the objects in your simulation, you may need to adjust the Nucleus node Space Scale. Decreasing Space Scale results in a visibly increased gravitational affect on the nParticles and makes them rise properly.

5 In the Attribute Editor, click the nucleus1 tab.

6 In the Scale Attributes section, set Space Scale to 0.05.

7 Play the simulation.



The nParticles now rise through the field and behaving more like smoke. In the next section, you use the nParticle Shading attributes to make the nParticle object look more like cigarette smoke.

Adjusting nParticle Shading attributes

By adjusting nParticle Opacity, you can mimic the way smoke disappears into the air as the nParticles age. You can then set the Color ramp to add shades of white, gray, and blue to the smoke.

Like the nParticle Radius attribute, Opacity and Color can be set on a per nParticle object or per-particle basis. Using the nParticle object's Shading ramps, you can map per-particle Opacity and Color to nParticle properties such as particle Age, Particle ID, and Radius.

To set nParticle Shading attributes

- 1 To make it easier to see the changes you make to the nParticle object's Shading attributes, you can hide the Volume Axis field shape by selecting *Field_Smoke1* in the Outliner then selecting **Display > Hide > Hide Selection**.

The Volume Axis field still affects your nParticles, but you no longer see the volume shape in the scene view.

- 2 In the Outliner, select *nParticle1_Smoke*.

3 In the Attribute Editor, click the nParticle_SmokeShape1 tab.

4 In the Shading section, set Opacity to 0.1.

5 Play the simulation.

Notice that the emitted nParticle cloud appears thinner and more like cigarette smoke than before.

To further improve the appearance of the smoke as it rises and dissipates in the air, you can use the Opacity Scale ramp to adjust the opacity of individual nParticles as they age.

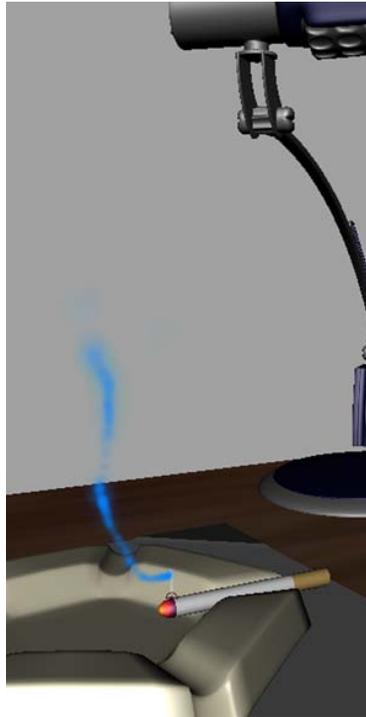
6 In the Opacity Scale section, click in the ramp to create additional markers then set an Opacity ramp by setting the following marker settings:

Marker	Selected Position	Selected Value	Interpolation
1	0.299	0.980	Spline
2	0.560	0.915	Spline
3	0.709	0.580	Spline
4	0.984	0.0	Spline

7 Set Opacity Scale Input to Normalized Age.

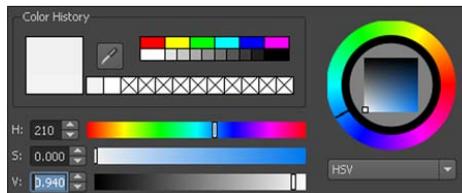
As each nParticle ages from birth (emission), its opacity changes from the opacity values specified by the left marker to the one specified by the right marker.

8 Set Opacity Scale Randomize to 0.116.



- 9 To set the color of the nParticles when they are first emitted, click the far left circle marker of the ramp (at Selected Position 0) then click the color swatch beside Selected Color.

The Color Chooser appears.



- 10 In the Numeric Input section, click the HSV tab.
- 11 Either use the sliders to select a white color with a little grey, or type the following color values into the HSV (Hue-Saturation-Value) fields:
 - H:210
 - S: 0

- V: 0.95
- 12 Drag your mouse off the Color Chooser to close it.
 - 13 To set the color of the nParticles as they continue to age, click inside the ramp to create a marker, set the Selected Position to 0.834, then click the color swatch beside Selected Color.
 - 14 In the Color Chooser, either use the sliders to select a shade of grey that is slightly darker than the color you selected for the first marker, or type the following color values into the HSV (Hue-Saturation-Value) fields:
 - H:210
 - S: 0
 - V: 0.85
 - 15 Select the far right marker of the ramp (at Selected Position 1.0) and set the color to the following color values into the HSV (Hue-Saturation-Value) fields:
 - H:191
 - S: 0.084
 - V: 0.85
 - 16 Set Color Input to Normalized Age.
As each nParticle ages, its color changes from the color specified by the left marker to the one specified by the right marker.
 - 17 To improve the way the nParticles colors blend as they age, do the following:
 - Set Interpolation to Smooth for each marker.
 - Set Color Randomize to 0.5.

When you play the simulation, the nParticles look more like cigarette smoke.

Adding more volume to the smoke would further improve the simulation. Instead of reducing the nParticle Opacity values, you can emit more particles and increase the nParticle Radius to create more smoke volume.



To add volume to the nParticle smoke

- 1 In the Outliner, select *nParticle_Smoke*.
- 2 In the Attribute Editor, click the *nParticle_SmokeShape* tab.
- 3 In the Lifespan section, set Lifespan to 10.
- 4 In the Particle Size section, set Radius to 6.5.
- 5 In the Attribute Editor, click the *Emitter_Smoke1* tab.
- 6 In the Basic Emitter Attributes section, set Rate (Particles/Sec) to 128.4.
- 7 Play the simulation.

You can continue adjusting the Opacity Scale and Color ramps to create the exact dissipation effect and color you want for the smoke.

Open the third scene for the lesson

To ensure this lesson works as described, open the scene file named *Smoke_Simulation_3.mb*. This file is in the *GettingStarted* directory that you set as your Maya project:

`GettingStarted/nParticles/Smoke_Simulation_3.mb`

Fine tuning your smoke effect

With nParticles, you can use dynamic wind generated by the Maya Nucleus solver to affect the motion of your particles. Gravity and Wind forces generated by the Nucleus system are different from the Maya dynamic forces that you create with Maya Fields. Nucleus forces are internal, meaning that they only affect Nucleus objects. Nucleus objects that are assigned to the same Nucleus solver are affected by the same intensity of the solver's wind and gravity.

In this section of this lesson, you create an Axial Magnitude ramp to control the field's magnitude at various points in the field volume. You specify that the field's magnitude gradually increases from the bottom of the volume to its top, so that as the nParticles travel through the field volume, the field's influence on the nParticles increases. To finish the smoke simulation, you use the Nucleus Wind to make the smoke drift as it rises.

NOTE If you had other Nucleus objects (nCloth objects or other nParticle objects) assigned to the same Nucleus solver, they would also be affected by the force of the wind.

To create an Axial Magnitude ramp

- 1 In the Outliner, select *Field_Smoke1*.
- 2 In the Attribute Editor, expand the Volume Control Attributes section, and go to the Axial Magnitude section.
- 3 Click in the Axial Magnitude ramp to create two new markers, and create a ramp by setting the following:

Marker	Selected Position	Selected Value	Interpolation
1	0.055	0.520	Linear
2	0.315	0.900	Linear
3	0.961	1.0	Linear

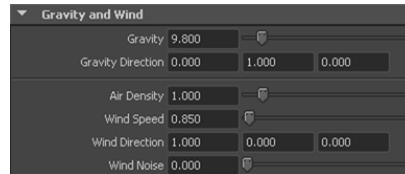


- 4 Play your simulation.

To add Nucleus Wind in your simulation.

- 1 In the Outliner, select *nParticle_Smoke*.
- 2 In the Attribute Editor, click the nucleus1 tab.
- 3 In the Gravity and Wind section, set Wind Speed to 0.85.

Wind Speed determines the force and intensity of the wind. A higher value means a faster wind speed, which has more of an effect on the smoke.



- 4 To set Wind Direction, do the following:
 - Set wind direction to 0 in the X-axis.
 - Set wind direction to -1 in the Y-axis.
 - Set wind direction to -1 in the Z-axis.
- 5 Set Wind Noise to 0.686.

This adds some randomization to the wind direction.
- 6 Play the simulation.



To make the smoke simulation more realistic, you can set up your effect so that the cigarette smoke is rising in the first frame of the simulation. You do this by setting the nParticle object's initial state.

To set the nParticle Object's initial state

- 1 Play the simulation and stop the playback at or around frame 200.
- 2 In the Outliner, select *nParticle_Smoke*.
- 3 Select nSolver > Initial State > Set From Current.
- 4 Rewind the simulation to the start frame and play it back.

Beyond the lesson

In this lesson you created an nParticle system and used Maya Fields to influence the behavior of the particles to create a smoke simulation effect. In addition, you learned how to:

- Use a volume emitter with an nParticle system.
- Use a Volume Axis field to create a disturbance in the path of your nParticles.

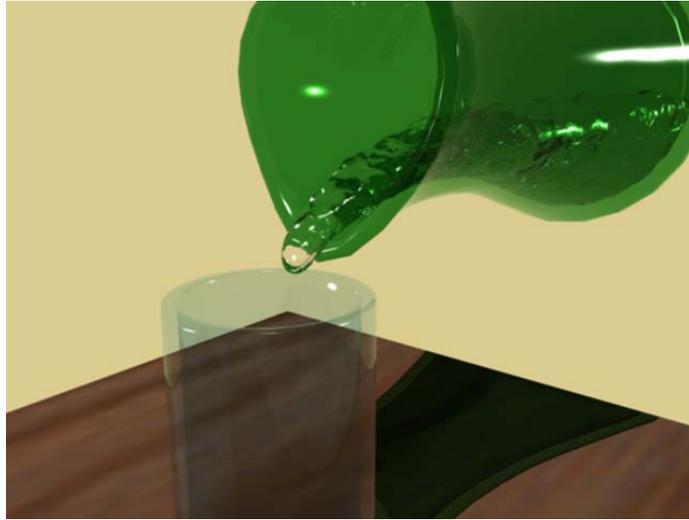
You can also add other Maya fields, such as Turbulence, Radial, or Vortex fields to affect the behavior of nParticles.

- 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 Opacity and Color 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 Opacity Scale and Color ramps.
- Create an Axial Magnitude ramp to control how a field's Magnitude affects nParticles as they travel through the field's volume.
You can fine tune your smoke simulation by adjusting the markers on the curve to produce the desired effect of turbulence as the smoke rises.
- 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.

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

Lesson 3: Creating a liquid simulation with nParticles

Introduction



A common use of particle systems is to simulate liquids such as flowing lava, water sprays, and mud splatters. Using nParticle Liquid Simulation attributes, you can add properties to an nParticle object so that it behaves like a liquid. For example, you can add viscosity and incompressibility properties so that the nParticles fill a volume uniformly and flow like a continuous substance.

When creating a liquid simulation, it is easiest 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 number of particle in the system and edit the attributes that give the nParticles the appearance of a liquid.

In this lesson you create a small-scale liquid simulation that involves pouring Water type nParticles from a pitcher into a glass. You can then use the techniques described in this tutorial to create larger liquid simulations.

In this lesson you learn how to:

- Use Fill Object to create a Water style nParticle system.

- Adjust nParticle Liquid Simulation attributes to create particles that look and behave like water.
- Cache your nParticle simulation.
- Convert a liquid simulation to an nParticle output mesh.
- Assign mia_material_x mental ray shaders to your nParticle output mesh.
- Render a single frame of your nParticle simulation using mental ray for Maya renderer.

Lesson setup

To ensure this lesson works as described, open the scene file named `Liquid_Simulation_1.mb`. This file is in the `GettingStarted` directory that you set as your Maya project:

```
GettingStarted/nParticles/Liquid_Simulation_1.mb
```

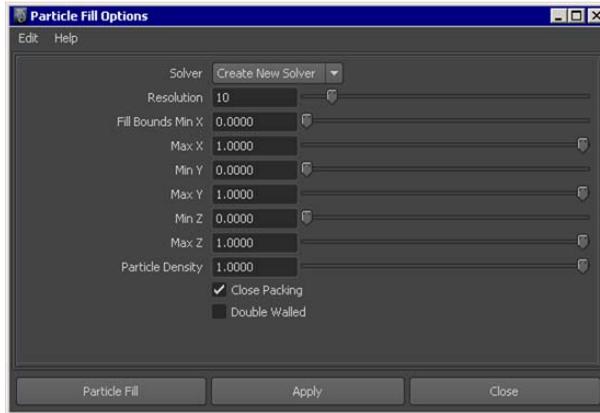
This scene includes textured polygon models including a desk, water pitcher, and water glass. The water pitcher mesh is animated to lift off the desk, tip to pour water into the glass, then set back down on the desktop.

Creating a Water style nParticle object

In the first step of creating your liquid simulation, you fill a polygon water pitcher with Water style nParticles. To do this, you use the Fill Object option.

To create a Water style nParticle object

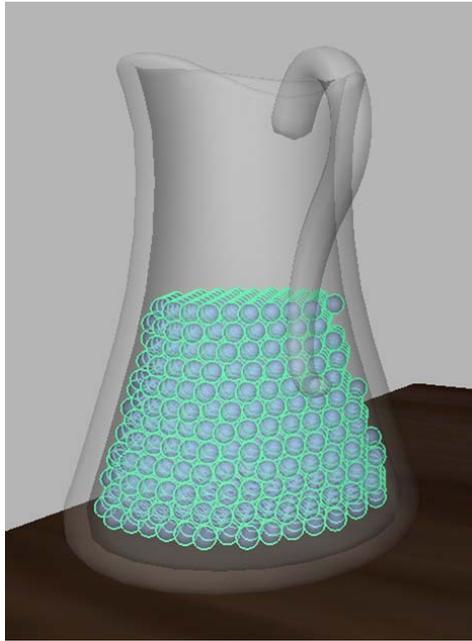
- 1 Select `nParticles > Create nParticles > Water`.
Selecting a Water style nParticle object presets some nParticleShape attributes including turning on Liquid Simulation attributes, turning off nParticle Self Collisions, and setting the Particle Render Type to Bloppy Surface.
- 2 In the Outliner, select `geo_pitcher`, which is grouped in the `grp_Pitcher` object.
This is the geometry that you will fill with Water style nParticles.
- 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 Set the following Particle Fill Options:
 - From the Solver list, select Create New Solver.
 - Set Resolution to 20.
Resolution specifies how many nParticles are evenly placed along the geometry's longest axis.
 - Set Max Y to 0.5.
When set to 0.5, nParticles half-fill the pitcher.
 - Ensure that Close Packing is turned on.
 - Turn on Double Walled.
Turning on Double Walled ensures that the thickness of the pitcher geometry is taken into account when it is filled with nParticles. The particles now only fill the pitcher's empty volume, rather than the space between the polygon walls.

For information about Particle Fill Options, see nParticles > Create nParticles in the nDynamics guide.

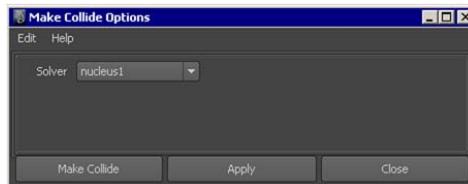
- 6 Click Particle Fill.
Maya creates the nParticle object inside the water pitcher geometry.



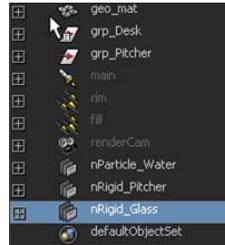
- 7 In the Outliner, double-click *nParticle1* and rename it to *nParticle_Water*. Renaming your nParticle object makes it easier to identify when you have more than one particle system in your scene.
In this lesson, both the pitcher and the glass mesh will hold nParticles, so you must convert meshes to passive collision objects and assign them to the same Nucleus system as the nParticles.

To convert the pitcher and glass meshes to passive collision objects

- 1 In the scene view, Shift-select the pitcher and glass meshes.
- 2 Select nMesh > Create Passive Collider >  .
The Make Collide window appears.



- 3 From the Solver list, select nucleus1.
- 4 Click Make Collide.
- 5 To rename the pitcher and glass nRigid objects, in the Outliner, double-click *nRigid1* and *nRigid2*, and type *nRigid_Pitcher* and *nRigid_Glass* respectively.



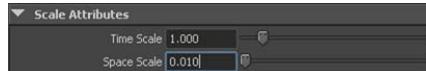
- 6 Play the simulation.

When you play the simulation, notice that the nParticles do not behave as you might expect. Rather than immediately falling to the bottom of the pitcher like water, they fall slowly, as if the simulation is playing in slow motion. To make the nParticles behave realistically, you can set the Nucleus Space Scale for the simulation.

For more information about Space Scale, see [Setting Nucleus Space Scale](#) on page 952.

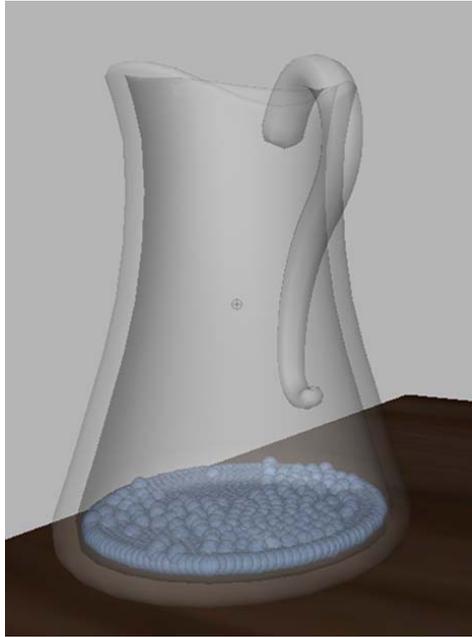
To set Space Scale

- 1 In the Attribute Editor, click the nucleus1 tab.
- 2 In the Scale Attributes section, set the Space Scale to 0.01.



When you reduce the Space Scale value, Maya evaluates the nParticle system as if the particles are much smaller in size. This results in a visibly increased gravitational effect on the nParticles.

- 3 Rewind the simulation to the start frame, and then play it back.



When you play the simulation, notice that the nParticles are still 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 more like a liquid.

Adjusting Liquid Simulation attributes

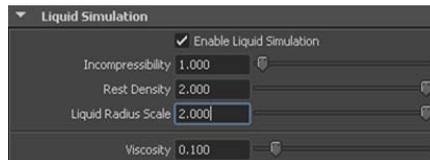
One of the main characteristics of water is that its volume uniformly fills the area inside a holding vessel or container. To give nParticles this volume characteristic, there needs to be space between individual particles. Be aware that having too much space can hinder the fluidity of the particles, which is another important characteristic of a liquid simulation.

Using the Liquid Radius Scale attribute, you can control the amount of space between nParticles.

Adjusting the liquid's volume

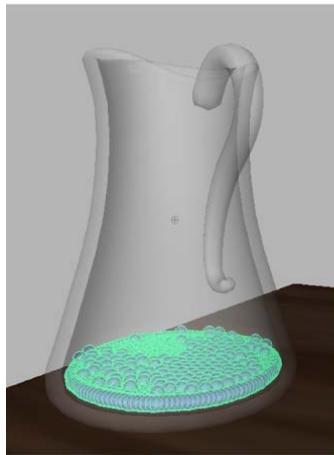
To set Liquid Radius Scale

- 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 on.
- 4 Set Liquid Radius Scale to 2.0.

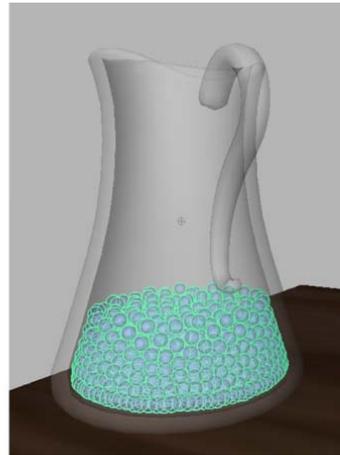


- 5 Play the simulation.

The nParticles are now stacked on top of each other, creating more volume in the pitcher.



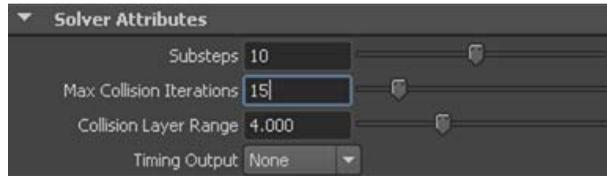
Liquid Radius Scale: 1.0



Liquid Radius Scale: 2.0

For this liquid simulation, you want the nParticles to fill one-third of the pitcher. You can add more volume to the nParticles by adjusting the Nucleus solver Substeps and Max Collision Iterations.

- 6 Set Substeps and Max Collision Iterations by doing the following:
 - In the Attribute Editor, click the nucleus1 tab.
 - In the Solver Attributes settings, set Substeps to 10.
 - Set Max Collision Iterations to 15.



- 7 Play the simulation.

The nParticles now fill the water pitcher, and when the pitcher tilts to pour, the nParticles have enough volume to exit and fill the glass.

More about Liquid Radius Scale

Be aware that 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 affects how nParticles overlap in the liquid simulation.

Adjusting the Nucleus solver Substeps and Max Collision Iterations also affects the volume that is generated by the Liquid Radius Scale value. You can experiment by changing the Substeps and Max Collision Iterations values and observe how your liquid simulation is affected.

For more information about how nParticle attributes affect liquid simulations, see Liquid Simulations in the nDynamics guide.

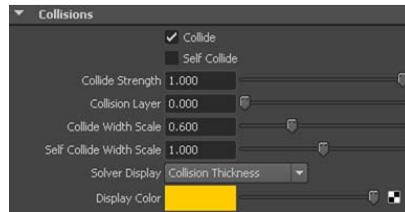
In the next section of this lesson, you adjust nParticle Collide Scale Width and Incompressibility attributes to add fluidity to your simulation.

Adding fluidity to the nParticles

To make it easier to see how the nParticles interact with the pitcher and glass objects, you can turn on the Solver Display attribute 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 collisions between Nucleus objects.

To edit Collide Width Scale and add fluidity

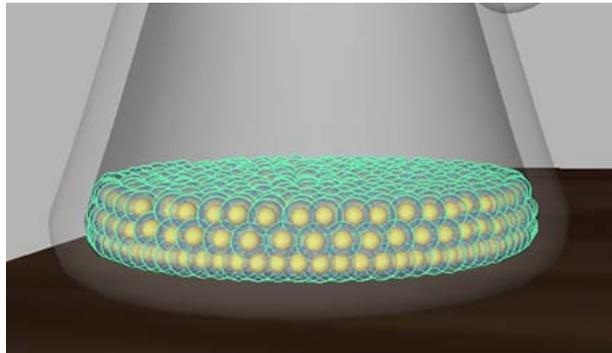
- 1 In the Outliner, select *nParticle_Water*.
- 2 In the Attribute Editor, click the *nParticle_WaterShape* tab.
- 3 In the Collisions section, select Collision Thickness from the Solver Display list.
- 4 Set Collide Width Scale to 0.6.



For 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.
- 6 In the scene view, dolly and tumble so that you get a close-up view of the nParticles at the bottom of the pitcher.

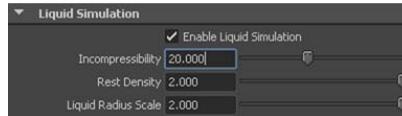
The nParticles occupy the pitcher in three distinct layers one on top of the other. In the bottom layer, the nParticles are evenly spaced and level, while in the upper layer, there are gaps between particles. Since the top layer of nParticles represent the surface of the liquid, the gaps between particles create a surface that is uneven and unlike the surface of water.



Increasing Incompressibility decreases the tendency of the particles to be pushed together. This forces the nParticles into a more uniform distribution throughout the volume, which creates a smoother 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:

- Rest Density: 2.0.
- Viscosity: 0.1.
To experiment with Viscosity, set it to 10 and play the simulation. Notice how long it takes for the nParticles to move from the pitcher to the glass due to the increased viscosity. Ensure you set Viscosity to its default values before continuing the tutorial.

Open the second scene for the lesson

To ensure this lesson works as described, open the scene file named `Liquid_Simulation_2.mb`. This file is in the `GettingStarted` directory that you set as your Maya project:

```
GettingStarted/nParticles/Liquid_Simulation_2.mb
```

Convert nParticles to a polygon mesh

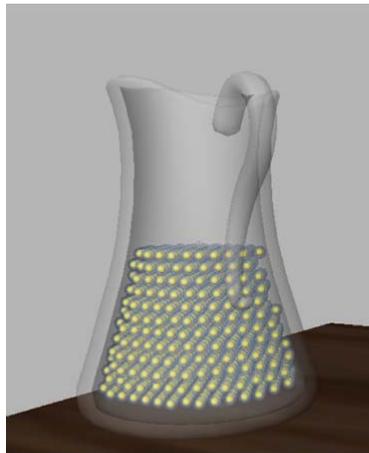
To create a smooth surface for your liquid simulation, you can convert the nParticles to a polygon mesh. Converting nParticles to a mesh has many advantages, because you can edit it like any other polygon object.

For example, you can improve the quality and overall look of your liquid simulation by performing polygon smooth operations on the output mesh. Polygon smoothing increases the number of polygons in your mesh and gives it a smoother appearance. For your liquid simulation, this helps create a smooth, continuous surface, like the surface of water. See nParticle output meshes in the nDynamics guide.

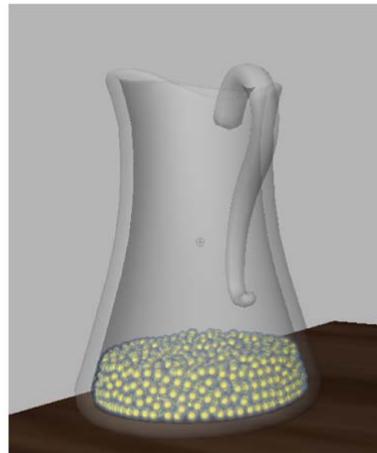
Before converting the nParticles to an output mesh, you can set the initial state of your simulation to an advanced frame. Doing this ensures that the nParticles have settled down to the bottom of the pitcher before its animation starts. From frames 1 to 30, Nucleus gravity and the initial collisions between the nParticles and the inside wall of the pitcher cause the particles to move. For a realistic effect, you want the water to be motionless at the bottom of the pitcher at the start of the simulation.

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, select *nParticle_Water*.
- 3 Select nSolver > Initial State > Set From Current.



Previous initial state



New initial state

- 4 Rewind to the start frame and play the simulation.

If you are not satisfied with how the particles behave in the new initial state, for example they may be still moving in the pitcher, play the simulation for 20 to 30 frames and set the initial state again. Continue setting the initial state until you are satisfied with the results.

To convert your nParticle object to a polygon mesh

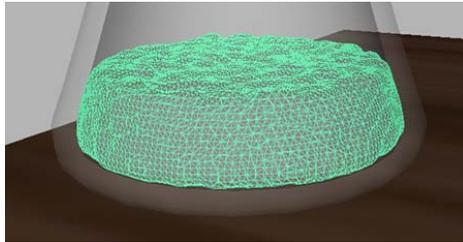
- 1 In the Outliner, select *nParticle_Water*.
- 2 Select Modify > Convert > nParticle to Polygons.

After your nParticle object is converted to a polygon object, Maya no longer displays the particles in the scene view. This reduces simulation time and makes it easier to see how your Output Mesh attribute adjustments affect the mesh.

The *PolySurface1* object appears in the Outliner, and a new *polySurfaceShape* node appears in the Attribute Editor.

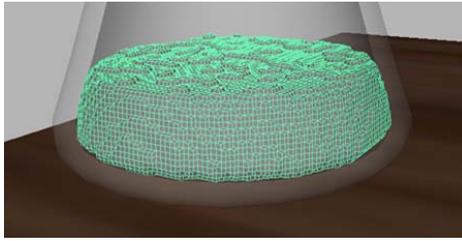
NOTE To make the nParticle object visible in the scene, in the Object Display section of the nParticleShape node Attribute Editor, turn off Intermediate Object.

- 3 In the Outliner, rename the new *PolySurface1* polygon object to *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 appears.

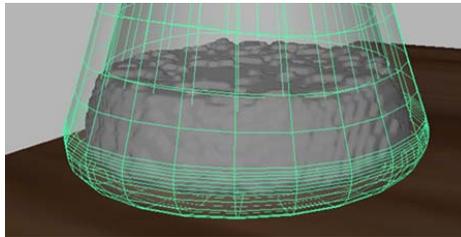
- 6 In the Output Mesh section, select Quad Mesh from the Mesh Method list.
- Selecting Quad Mesh creates your converted mesh using quad-based polygons, while Tetrahedra, Triangle Mesh, and Acute Tetrahedra create meshes using triangulated polygons. Output meshes that are quad-based respond better to polygon smoothing (Mesh > Smooth) than triangle-based output meshes.



- 7 Dolly and tumble the scene so that you can closely observe the nParticle output mesh, and experiment with the following Threshold values.

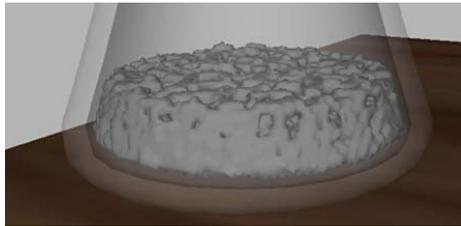
- Set Threshold to 0.2.

At this value, the nParticle output mesh extends beyond the nParticle object's collision width and penetrates the inner wall of the pitcher mesh.



- Set Threshold to 1.5.

At this value, holes appear in the Particle output mesh because some nParticles are no longer overlapping.

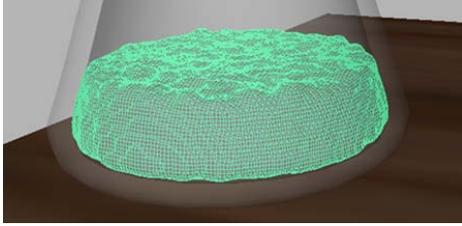


- Set Threshold 0.6 and continue.

- 8 Set Mesh Triangle Size to 0.2.

This setting determines the size of the triangles that are used to create the mesh. Small values produce higher resolution meshes with smoother surfaces. Be aware that higher resolution meshes take more time to simulate.

- 9 To apply smoothing to the corners of the mesh's quad, set Mesh Smoothing Iterations to 3.



The output mesh is noticeably smoother.

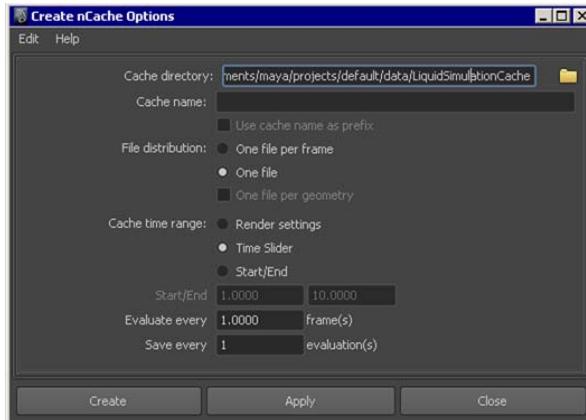
- 10 Play the simulation.

Cache your nParticle simulation

When you play the simulation, notice that it plays back much slower than before. This is due to the additional calculations required for the nParticle object's output mesh. You can use Maya's nCache feature to cache the simulation and play the scene at a faster speed. See nCaching overview in the nDynamics guide.

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.
Hiding the output mesh speeds up the nCaching process.
- 2 In the Outliner, select *Water_Mesh*.
- 3 Select nCache > Create New Cache > .



The Create nCache Options window appears.

- 4 In the Create nCache Options window, do the following:
 - Set the Cache directory to the folder where you want your caches saved.
 - Set the Cache name to *LiquidSimulationCache*.
 - For File distribution, select One file.
 - Click Create.

Maya plays the scene automatically and stores the frames in the directory you specified.

- 5 After the simulation is cached, show the *Water_Mesh* by selecting it in the Outliner and selecting Display > Show > Show Selection.
- 6 Play the simulation.

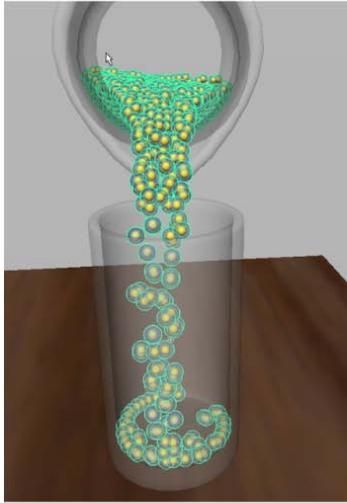
Notice that the scene now plays back faster. You can also scrub slowly through the timeline and view your simulation frame-by-frame. If you change any of the nParticle attribute settings, create a new cache to see your changes.

In the next section of this lesson, you use the cached simulation to help optimize specific areas of the simulation.

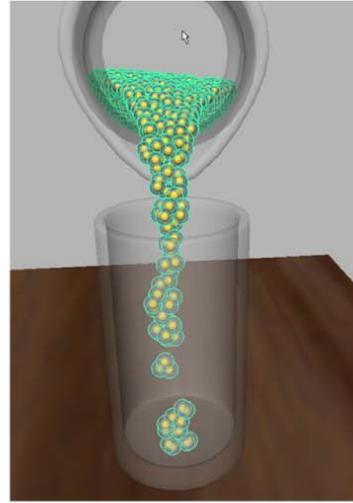
Adding surface tension

An important characteristic of liquids is surface tension. Surface tension is the attractive force between the molecules of liquids, and is most noticeable in phenomena such as the formation of water beads or liquid droplets on surfaces. You can use the Surface Tension attribute to add realistic surface tension to your liquid simulation.

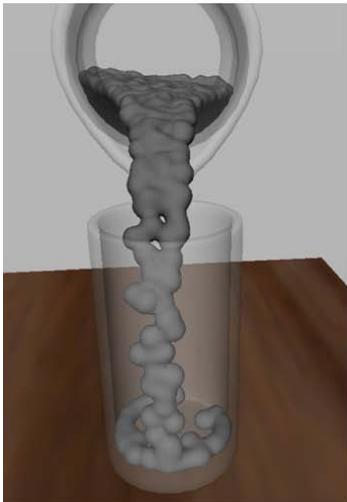
The Surface Tension attribute adds an attractive force between nParticles which keeps them together. The effect of Surface Tension is most visible in liquid simulations after the nParticles object is converted to an output mesh.



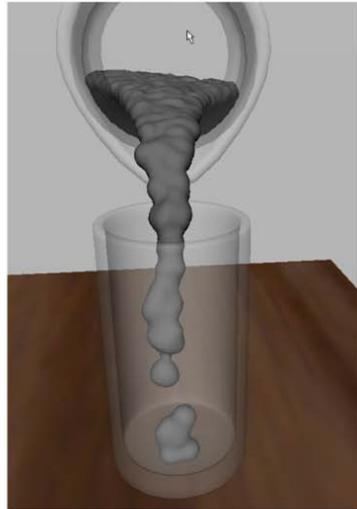
nParticles without Surface Tension



nParticles with Surface Tension



Output mesh without Surface Tension

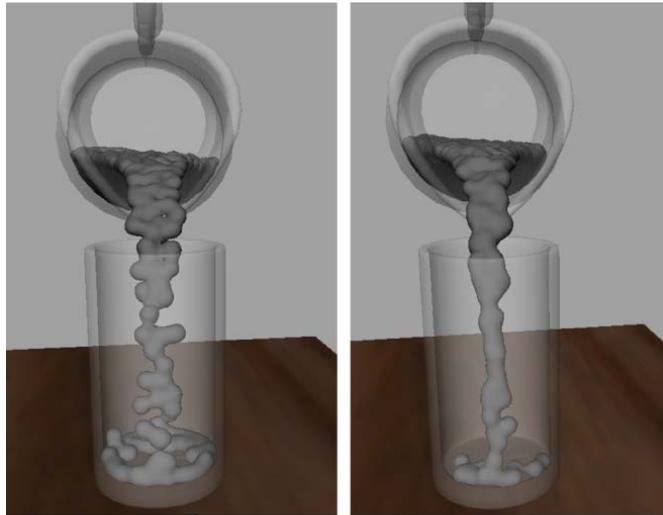


Output mesh with Surface Tension

To add surface tension

- 1 Go to frame 122, then dolly and tumble the scene so that you have a close-up view of the mesh pouring into the glass.
- 2 In the Outliner, select *Water_Mesh*.

- 3 In the Attribute Editor, click the nParticle_WaterShape tab.
- 4 In the Liquid Simulation section, set Surface Tension to 0.3.
- 5 Hide the *Water_Mesh* object and cache your simulation.
To cache your simulation, follow the steps in [Cache your nParticle simulation](#) on page 994. You can replace your existing nCache by selecting nCache> Replace Cache instead of nCache > Create New Cache.
- 6 Go to frame 122.
The mesh is noticeably more uniform, and looks more like a flowing liquid.



Surface Tension: 0

Surface Tension: 0.3

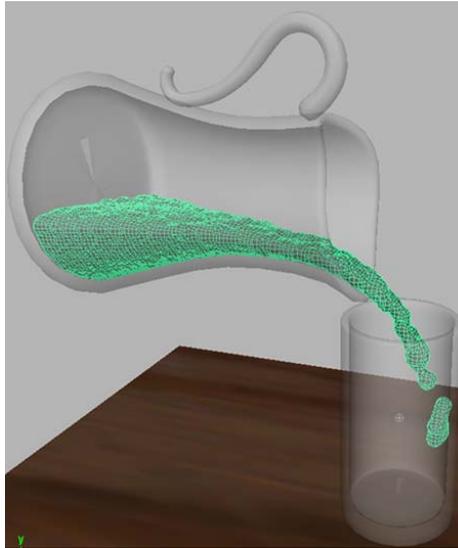
Adding Motion Streak

You can set the Motion Streak attribute to optimize the motion of the nParticles during the pouring sequence of the animation. Motion Streak is useful for creating a motion blur type of effect, and for shaping the flow of liquids.

To set Motion Streak

- 1 Go to frame 119, then dolly and tumble the scene so that you have a close-up view of the water nParticles pouring into the glass.

At this frame, notice that the droplet mesh pouring into the glass looks too spherical. It does not resemble the continuous flowing behavior of water poured into a glass.

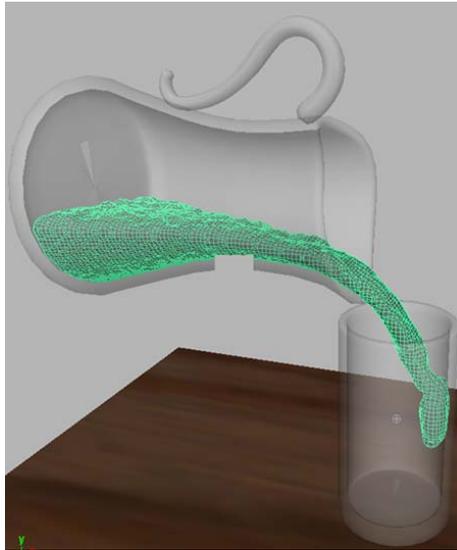


To make the mesh a continuous flowing object, you can adjust the Motion Streak attribute.

2 In the Attribute Editor, click the nParticle_WaterShape tab.

3 In the Output Mesh section, set Motion Streak to 0.50.

Maya updates the output mesh with the new setting. To further improve the quality of the mesh, you can perform a polygon smooth operation.



- 4 To smooth the mesh, select it, and from the Polygon menu set, select Mesh > Smooth.

The mesh is noticeably smoother and looks more like a liquid. Be aware that smoothing a mesh adds divisions to the polygons on the mesh, making it denser, which increases simulation time.



In the next section, you will assign material shaders to the pitcher, glass, and water meshes and render frames of the simulation using mental ray for Maya renderer.

Open the third scene for the lesson

To ensure this lesson works as described, open the scene file named `Liquid_Simulation_3.mb`. This file is in the `GettingStarted` directory that you set as your Maya project:

```
GettingStarted/nParticles/Liquid_Simulation_3.mb
```

Render your liquid simulation

In this section, you render your nParticle effect using the mental ray for Maya renderer. Since rendering a particle effect can take several minutes to an hour, in this lesson, you only render selected frames of the simulation. Rendering simulated frames is the easiest way to see how some attributes affect your simulation.

To prepare your scene for rendering, you assign mental ray `mia_material_x` shaders to the pitcher, glass, and water meshes. Lights have been created and placed in the scene so that realistic shadows and reflections appear in the rendered frame. The scene also includes a camera named `renderCam` that is positioned for the rendered view. For more information about rendering, see the *Rendering and Render Setup* guide.

Preparing to render

To prepare your scene and workspace for rendering, you will do the following:

- Cache your simulation so that you can play it or scrub through it to locate the simulated frame you want to render. See [Cache your nParticle simulation](#) on page 994.
- Change the view arrangement of your Maya workspace to make it easier to assign material shaders to meshes and view rendered frames of your simulation.
- Show hidden light objects and the camera you use to render a simulated frame.

If you have not yet cached your simulation, do so now. See [Cache your nParticle simulation](#) on page 994.

To change the view arrangement and display hidden objects

- 1 Select Windows > Saved Layouts > Hypershade/Render/Persp.
The Hypergraph, Render View, and the current perspective view are displayed. This view arrangement makes it easier to assign shaders to your objects and to view rendered frames of your simulation.
- 2 Open the Outliner in a separate window by selecting Window > Outliner.
- 3 Show the *Water_Mesh* object by selecting it in the Outliner then selecting Display > Show > Show Selection.
- 4 In the Outliner, Ctrl-click the *main*, *rim*, and *fill* lights, then select Display > Show > Show Selection.



The lights used for rendering the simulation appear in the scene.

- 5 In the perspective view, select Panels > Perspective > renderCam.
You now view the scene through the camera that you will use to render.

Assigning material shaders

To prepare the geometry for rendering, you assign the mental ray *mia_material_x* shader to the pitcher, glass, and water meshes. Using the *mia_material_x* shader presets for the glass and water provide a quick and easy way to have proper refraction, reflectivity, and transparency values set on these materials. For more information about material shaders, see the mental ray *mia_material/mia_material_x* shader attributes section in the *Rendering and Render Setup* guide.

To assign a material to the pitcher and glass objects

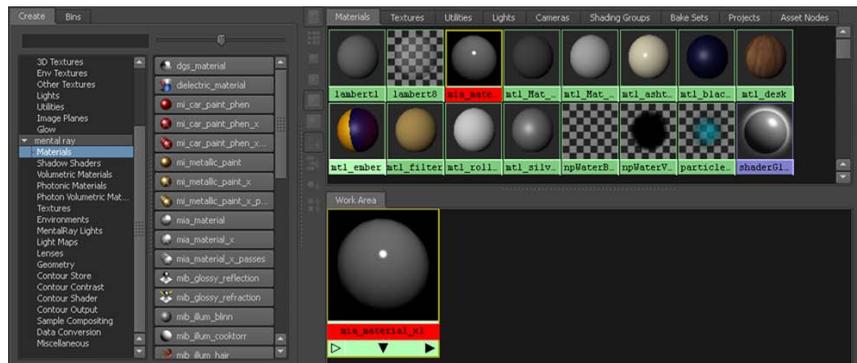
- 1 In the Hypershade, under mental ray, click Materials.

The right panel of the Hypershade now shows the list of mental ray shader nodes.



- 2 From the list, select `mia_material_x`.

A `mia_material_x` material swatch appears in the Work Area tab of the Hypershade, and the material attributes appear in the Attribute Editor.



- 3 In the Attribute Editor, rename the material to `mia_material_pitcher`.
- 4 Click and hold the Presets button, select GlassThin, then Replace (GlassThin > Replace).
- 5 In the Refraction section, click the color swatch beside Color.

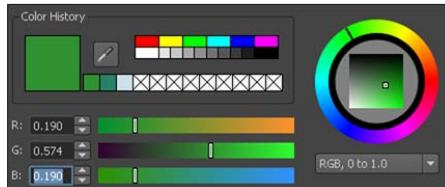
The Color Chooser appears.

6 In the Color Chooser, type the following color values into the RGB fields:

■ R: 0.190

■ G: 0.574

■ B: 0.190

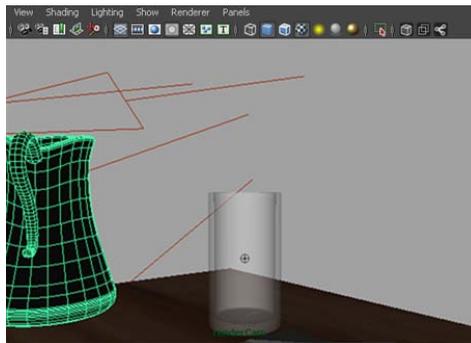


7 Move your mouse off the Color Chooser to close it.

8 In the Outliner, select the *grp_Pitcher* object.

9 To assign the material to the pitcher, in the Work Area tab of the Hypershade, right-click *mia_material_glass* then select Assign Material To Selection from the marking menu.

In the scene view, the pitcher turns black. You can only see the true color and texture of the material after the frame is rendered using mental ray for Maya renderer.



10 Assign materials to the glass and water meshes by repeating steps 2 to 10 using the following settings:

For the glass (*geo_glass*) mesh:

■ Rename the material to *mia_material_glass*.

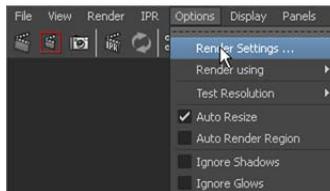
- Select Presets > GlassThin > Replace.
- In the Refraction section, set Color to R: 0.797, G: 0.875, and B: 0.915.
For the water (*Water_Mesh*) mesh:
 - Rename the material to *mia_material_water*.
 - Select Presets > Water > Replace.
 - In the Diffuse section, set Color to R: 0.164, G: 0.512, B: 0.402.
 - In the Refraction section, set Transparency to 1.0.

Rendering a simulated frame

Now that you have assigned *mia_material_x* shaders to the objects in the scene, you can use the mental ray for Maya renderer to render selected frames of your simulation.

To render a simulated frame

- 1 In the timeline, go to frame 119.
You will render this frame, where the output mesh pours into the glass.
- 2 In the Render View, select Options > Render Settings.

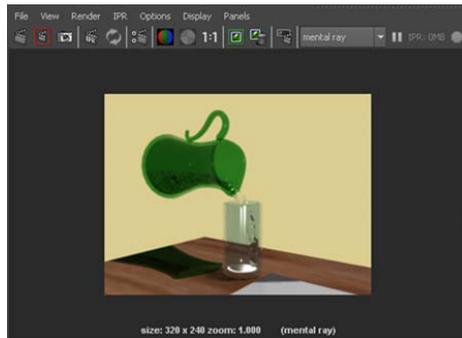


The Render Settings window appears.

- 3 From the Render Using list, select mental ray.
- 4 Click the Common tab and set the following:
 - In the Renderable Cameras section, select *renderCam* from the Renderable Camera list.
 - In the Image Size section, select 320×240 from the Presets list.

- 5 Click the Quality tab, and set the following:
 - From the Quality Presets list, select Production.
 - In the Raytracing section, turn on Raytracing.
 - Set Shadows to 4.
- 6 Click Close to close the Render Settings window.
- 7 To render the current frame, in the Render View, select Render > Render > renderCam.

The current frame appears as a rendered image in the Render View.



You can continue to make changes to the render settings, change the renderCam view, or select another simulated frame to render. After you are satisfied with your render settings, you can render the frame as a full-size image.

To render a full-size image

- 1 In the Render View, select Options > Render Settings.
- 2 Click the Common tab and in the Image Size section, select 640×480 from the Presets list.
- 3 From the Rendering menu set, select Render > Render Current Frame.

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 solver's Space Scale attribute to suit the scale of your simulation.
- Adjust Liquid Simulation attributes to set the volume and fluidity characteristics of your liquid nParticles.
You can continue to adjust the Liquid Simulation attributes, such as Liquid Radius Scale and Viscosity, to create liquid simulations of thicker substances such as oil, mud, or lava.
- Create and play back nParticle caches.
- Convert your nParticle object to a polygon mesh.
- Optimize the appearance of your nParticle output mesh using Output Mesh attributes such as Threshold, Mesh Smooth Iterations, and Motion Streak.
- Assign mia_material_x mental ray shaders to your nParticle output mesh.
- Render a single frame of your nParticle simulation using the mental ray for Maya renderer.

Index

- { } in expressions 641
- % in expressions 648

- 2D fluid container
 - creating 824
- 3D fluid container
 - creating 834, 843
- 3D Paint Tool 612

- A**
- Accuracy 532
- active rigid body 558
- Add Attribute button 645
- Add Attributes 548
 - for Current Render Type 547
- Add Dynamic Attributes 548, 645
- Add Influence 339
- Add Keys Tool 216
- Add Per Particle Attribute 548, 645
- Add Preview Plane 853
- Add/Edit Contents 824, 829, 844
- age, particles 549
- Along Axis 540
- Alpha Gain 79
- Amplitude 837
- animation
 - curves 210
 - planning, storyboarding 305
- Animation Preferences 206
- Annotation
 - creating 291
 - description 291
- anti-aliasing 457, 516
- Append to Polygon Tool 108
- arms
 - joint chain 321
- Around Axis 540, 543
- array
 - description 548
- Artisan Erase 614

- Assign Edit Textures button 612
- Assign New Material 466
- Assign Paint Effects Brush to Hair 813
- Assign Textures button 620
- assigning a texture map 364
- Attribute Editor
 - attribute tabs 66
 - description 65
 - Fur 881
 - viewing 65
- Attribute to Paint 612, 620
- attributes
 - controlling multiple 631
 - description 37
 - displayed in Attributes list 628
 - linking 631
 - shading 68
- audio clips 275
- Autodesk
 - technical support 11
 - Web site 11
- Automatic Mapping 374
 - applications 374
 - triplanar projection 374
- Away From Axis 540, 543
- Axial Magnitude ramp 977
- axis indicator 19

- B**
- bald region 875
- base object
 - influence object 339
- base shape
 - blend shape 357
- base width 885
- Bend
 - hair 794
- Bend Amount 794
- Bend Curves Options 794
- Billow 837

- Billow Density 838
- birth, particles 549
- blend modes 613
- blend shape deformer 350
 - adding shape to 355
 - editor 351
 - improving deformations 357
- blending animation types 235
- Blinn shading material 441, 443
 - description 68
- Blobby Radius Scale, nParticles 992
- Blur, 3D Paint 619
- boat locator 858
- boat wakes 859
- bones 315
- Boolean operations 132
- border edges 86
- Bounce
 - nParticles 955
- Boundary Draw 841
- Boundary X, Y 828
- Bounding Box 218
- braces, in statements 641
- brackets, double angle 646
- Break Tangents
 - Graph Editor 213
- Bridge 105
- Browse button 616, 621
- brush profile 617
- brushes
 - 3D Paint, Paint Effects 617
 - description 577
 - Kelp 594
 - Mesh 599
 - Teapot 600
 - Thin Line 610
- bump, painting 620
- Buoyancy 858

C

- caching
 - nCloth 929–930
- caching fluids 832
- camera
 - applications 495
 - create 496
- Camera Attribute Editor 453
- camera tools
 - Dolly Tool 43
 - overview 42
 - Track Tool 45
 - Tumble Tool 45
- cameras
 - animating 497, 500
 - Film Gate 500
 - undoing changes 498
- Canvas
 - Clear 569
 - Set Size 568
- Canvas Clear 566
- Caustic Photons 533
- Caustics
 - Caustic Photons 528
 - description 519
 - Emit Photons 528
 - light sources 534
 - Photon Intensity 528
 - photon maps 527
 - photons 527
- Channel Box 24
 - blend attributes 237, 239
 - description 37
 - hiding, showing 24
 - locking channels 296
 - naming and renaming objects 38
 - transforming and rotating objects 37
 - translation, rotation, scaling 24
- character set 248
- character setup
 - description 313
- Circle 176
- clear canvas 566
- clip cycle
 - description 263
- clip library 270
- clip manipulator 259
- clips
 - applications 244
 - audio 275
 - blending 274

- components 247
- creating 246, 249
- cycling 263
- description 241, 244
- duration 247
- importing 254
- inserting 270
- keyframing 275
- name 247
- Outliner 275
- regular clips 246
- renaming 249
- repositioning 251
- reusing 254
- scale 248
- scaling 259
- source 246, 254
- Visor 275
- cloth texture 445, 447
- Clump Width
 - hair 803
- cluster deformer 344
- cluster handle 345
- cluster weights 347
- Collapse 292
- collapse hierarchy 292
- Collide Sphere 798
- Collide Width Scale 956
 - nParticles 989
- Collision Flag 908–910
- collision object
 - fluid 831
- collisions
 - nCloth 900–910, 930–931
 - nParticles 950, 988
- collisions, rigid bodies 554
- color
 - Christmas light effect with
 - particles 648
 - emitting 829
 - fluid property 824
 - fluid shader 849
 - fur 882
 - giving object constant color 645
 - giving particles randomly changing
 - color 646
 - particles 548
 - ramp 549
- Color 613
 - nParticles 953
- color bleeding 515
- Color Chooser 69, 455
- Color Feedback
 - Paint Fur Attributes Tool 878
- Color Input 836, 850
- Color Method 829
- Combine 105
- components
 - description 62
- conditional statements
 - if 636
- Conserve attribute 545
- constraints 555, 912–923
 - adding and removing members 917
 - description 289
 - nCloth 913, 917, 939–940
 - orient 294
 - parent 302
 - point 292
 - Point to Surface 913
 - rigid bodies 555
- construction history 120, 157, 183
- container
 - boundaries 828
 - creating 824, 834, 843
 - resolution 834
 - size 834
- Contents Method
 - dynamic fluid 827
 - Dynamic Grid 826, 829, 844, 848
 - Gradient 835
- control object (IK) 289
- control vertices (CVs) 156
 - description 63
- Convert Selection
 - hair 798
- converting a UV selection 382
- Crease Proxy Edge Tool 130
- creasing
 - subdivision surface 199
- Create
 - Expression Editor 628

- Create 2D Container 824
- Create 2D Container with Emitter 825
- Create 3D Container 834, 843
- Create Blend Shape 350
- Create Clip
 - options 262, 264
- Create Cluster 345
- Create Constraint 557
 - hair 798, 815
- Create Hair 788, 810
- Create Ocean 853
- Create Polygon Tool 98
- Create Ramp 549
- creating
 - constraints 913–917, 939–940
 - nCloth 898–900, 926
 - nParticles 948, 960
 - passive collision objects 900, 928
- creation expressions 645
 - assigning to rgbPP 645
- cube primitive 21, 80
 - options 46
- Curl
 - hair 803
- Curl Frequency
 - hair 803
- Current Position 789
- curve
 - direction 156
 - edit revolved 157
 - start 156
- curves, dynamic 783
- curves, hair 786
- cutting
 - subdivision surface 194
- CV Curve Tool 155
- cycling a clip 263

D

- deforming objects with fluids 833
- Delete by Type 121
- Delete Keys 329
- Density
 - description 825
 - emitting 824
 - fluid property 824
 - fur setting 885
 - painting 843
- Depth Map Shadows 804
- Depth Max 837
- detail
 - subdivision surface 195
- Dim Image 375
- directional light
 - create 486
 - description 486
 - editing attributes 487
- displacement
 - ocean 853
- Displacements
 - hair 803
- display
 - creased edges 131
 - only selected object 353
 - Poly Count 132
 - soft and hard edges 127
 - template 61
- Display Image 377, 382
- Display Level 196
- Display Quality
 - hair 790–791
- Display Resolution 453
- Display Start Position
 - hair 792
- Display Unfiltered 385
- Dolly Tool 43
 - description 43
- double angle brackets 646
- Draft rendering setting
 - description 504
- driven keys
 - description 219
- duplicate
 - symmetrical 863
- Duplicate Tool 178
 - description 39, 55
 - options 39, 48
- Duration 247
- dynamic attributes
 - adding 546, 644

- dynamic curves 808
 - hair 783
- dynamic fluid
 - behavior, changing 826
 - Contents Method 827
 - creating (3D) 842
 - description 823
- Dynamic Grid 826, 829, 844, 848
- Dynamic Properties 923, 926
- dynamics
 - definition 535

E

- Eccentricity 857
- edge loop 86
 - description 86
 - termination 118
- edges
 - border 86
 - creasing 126
 - hardening 126
 - selecting subdivision surface 197
- Edit button
 - expressions 631
- Edit Deformers > Blend Shape > Add 355
- Edit Ramp 554
- elbow
 - influence object 339
- electronic tablet
 - sculpting surfaces 168
- else keyword
 - expressions 640
- emission rate
 - nParticles 954
- Emit Fluid Color 830
- Emit from Object 538
- emitter
 - nParticles 948, 960, 962
- emitters 536
 - creating 537
 - Omni 538
- emitting
 - color 829
 - Density 824
 - fluid 824

- fluid from surface 832
- End Bounds 582
- End Time 206
- equal to (==) operator 648
- Equalize
 - hair 789
- errors
 - syntax 637
- Essential Skills Movies 5
 - closing the movie window 6
 - playing 6
- exiting Maya 27
- Expression Editor 627
- expressions 858
 - advantage of separate 632
 - advantage of single 632
 - creating 628
 - description 625
 - editing 630
 - execution details 633
 - naming conventions 627
- Extrude 100
- extrusion manipulator
 - subdivision surfaces 193

F

- face selection mode 84
- faces
 - splitting 110
- facial animation 341
- FCheck 461
- feedback
 - fur 870
- fields 536, 553
- file saving 26
- Fill Object, nParticles 982
- fingernails 197
- fingers 189
- floating geometry 857
- Flood 847, 880
 - Sculpt Geometry Tool 165
- Flood All 615
- fluid container
 - boundaries 825
 - creating 824, 834, 843

- fluid dynamics 823
- fluid properties 824
 - adding constant values 835
 - emitting 824
 - painting 843
- Fluid Slice Location 846
- fluids
 - attributes, changing 826
 - color 829
 - deforming geometry with 833
 - emitting 824
 - emitting from surface 832
 - forces 825
 - painting properties 843
 - resolution 834
 - self-shadowing 840
 - shader attributes 836
 - texturing 836
- follicle
 - attributes 802
 - hair 784
- forces
 - container boundaries 825
 - Gravity 827
- Forward Kinematics (FK) 276, 325, 329
- Four Panes 155
- Frame In 247
- Frame Out 247
- Frame Selection
 - Graph Editor 211
- frames per second setting 440
- Freeze Transformations
 - description 290
- Friction
 - nParticles 955
- Fuel
 - painting 843
- Full Crease Edge/Vertex 199
- fur
 - Attribute Editor 881
 - attributes 870
 - base width 885
 - color modification 882
 - Density 885
 - description 868
 - direction 872

- Direction Offset 873
- feedback 870
- Inclination 885
- Length 885
- Paint Attributes Tool 875
- Polar 873
- presets 862
- reduce length 875
- Roll 873
- Scraggle 885
- tip width 885

G

- Get Brush icon 617
- Get Fluid Example 842
- Getting Started with Maya
 - about the lessons 2
 - before you begin 3
 - introduction 1
 - lesson conventions 4
 - using lesson files 5
 - using tutorials online 3
- Global Illum Photons 509, 515
- Global Illumination
 - Accuracy 513
 - color bleeding 515
 - description 501
 - light sources 517
 - photon maps 508
 - photons 508
 - Radius 513
 - rendering process 508, 511
- Glue Strength 816
- Go to Bind Pose 338
- Graph Editor 209
 - path animation 231
 - Set Driven Key 223
 - tangent modification 232
- Gravity 558, 584, 827, 923
 - hair 791, 796
- Gravity and Wind
 - nParticles 977
- ground plane 579
- group nodes
 - animating 218

- Group Under 343
- grouping objects
 - description 52
- Growth attributes 607

H

- hair
 - assigning a Paint Effects brush 813
 - colliding with objects 812
 - constraining 814
 - creating 788
 - creating Rest curves 795
 - improving playback
 - performance 791
 - modifying 802
 - Paint Hair Tool 784
 - playing the hair simulation 790
 - rendering 801, 817
 - setting up collisions 797
 - shadowing 804
 - styling 792
- Hair Color 813
- hair curves 786
- hair system 784
 - attributes 802
- Hair to Hair constraint 814–815
- Hair Width 803
- hairConstraintShape 816
- Hairs Per Clump 802
- hairSystemShape 791
- Harden Edge 128
- Hardware renderer
 - description 451
- Help
 - Find Menu 10
 - launching 7
- Help resources 5
- hide layers 876
- Hide Selection 183
- hierarchy
 - about 278
 - collapse 292, 301
 - description 52
 - grouping 52
 - skeleton 315

- viewing in hypergraph 279
- Hinge constraint 557
- hole
 - creating subdivision surface 194
- Hotkeys
 - brush scale 595
 - description 55
 - setting 55
- Hypergraph 53
 - character setup 317
 - description 53
 - overview 53
 - viewing 53
 - viewing hierarchies 53
- Hypershade
 - Create Textures 477
 - description 473
 - rename operation 476

I

- if statements
 - expressions 635
- if-else statements
 - expressions 639–640
- Ignore Solver Gravity 962
- IK handles 325
 - control object 289
 - creating 287
 - creating and animating 326
 - description 287
 - parenting 298
- IK Rotate Plane handle 328
- IK Solvers
 - Enable and Disable 338
 - ikSCsolver 287
- image plane
 - description 77
 - importing 78
 - transparency 79
- image size 505, 516, 524
- Import Clip 254
- Incandescence 850
- Incandescence1 618
- Inclination 873, 885

- Incompressibility
 - nParticles 990
- Influence list
 - Paint Skin Weights Tool 334
- influence object
 - smooth skin 337
- initial orientation
 - hinge constraint 557
- Initial State 936–938
- Input Mesh Attract 933
- Insert Edge Loop Tool 106, 114
- Insert Isoparms 172
- Inverse Kinematics (IK)
 - description 276, 325
 - limiting motion 296
 - translation limits 301
- IPR 621
 - description 451
 - region to be updated 455
- IPR Render Current Frame 454
- Isolate Select 353
- isoparms 161
- Iterations
 - hair 796

J

- jaw
 - joint 319
- joint chains 315
- Joint Size 316
- Joint Tool 316
- joints 315
 - influence on skin 332
 - moving 322

K

- Keep Aspect Ratio 612
- Keep Hard Edge 129
- Keep Image 507, 525
- keyframes (keys)
 - adding 216
 - deleting 215
 - description 204
 - setting 207

- keying into a clip 275
- keys
 - delete redundant 216
 - Key Selected 310
 - moving 212
 - setting 308

L

- Lambert shading material 441–442
- Layer Editor 866
- layers
 - assign objects to 868
 - create 867
 - description 866
 - editor 866
 - hide 876
 - reference 867, 883
 - settings 883
 - template 883
- layouts
 - changing panel layout 32
 - four view layout 39
 - shortcuts 32
- Leaf Width Scale attributes 606
- Leaves attributes 607
- legs
 - joint chains 315
- Length
 - hair 789
- Length Flex
 - hair 796
- level of detail
 - subdivision surface 195
- lifespan attribute 549
- Lifespan, nParticles 964
- lights
 - add to a scene 888
 - applications 485
 - attributes 491
 - directional 586
 - dropoff 492
 - intensity 587
 - penumbra 492
 - spotlight 488, 506
 - spotlights 888

- Line Width 547
- Linear
 - Graph Editor tangents 212
- linking attributes 631
- Liquid Radius Scale
 - nParticles 987
- Liquid Simulation attributes 986
- liquid simulations
 - nParticles 981
- Load Driver
 - Set Driven Key 221
- Load Selected Characters 246
- Locators
 - hair 798, 801
- Lock Length
 - hair 793, 815
- Loft Tool 176, 180
- Look Through Selected 490

M

- Magnitude 540
- main menu 20
- Make Boats 858
- Make Collide 831
 - hair 813
- Make Motor Boats 859
- Make Paintable 594
- manipulators, subvolume 844
- Map Visualizer 518, 534
- mapping UVs 373
 - applications 373
- marking menu 63
- masks and modes, selection
 - description 56
- Mass 923
- material
 - assigning 466
 - Blinn 441
 - default shading 67
 - description 441, 466
 - Lambert 441
 - surface 67
- Max Collision Iterations 942
- Max Displacement 166
- Max Influences 332
- Max Photon Depth 530
- Max Self Collide Iterations 941
- Maya
 - about 1
 - installing 3
 - launching 15
 - user interface 15–16
 - using Maya Help 7
- Maya Hardware renderer
 - description 451
- Maya Help
 - index 7
 - popup help 8
 - Tutorials 3
 - using 7
 - using Help index 7
 - using Help Line 10
 - using Search 8
- Maya nCloth 895
- Maya nParticles 945
- Maya Nucleus 895, 945
- Maya Software renderer
 - description 451
- Maya Vector renderer
 - description 451
- mental ray for Maya renderer 438, 452, 468, 523, 801
 - description 450
- mental ray shaders
 - assigning to an nParticle output mesh 1002
- menu sets 20
 - description 20
 - Polygons 74
 - selection 20
- Mesh brushes
 - applications 599
 - attributes 604
 - description 599
- meteors 546
- Mirror Across 318
- Mirror Geometry 122
- modeling
 - description 73, 719
 - polygons 74
 - shaded mode 82

- symmetry 84
- types 73
- Modify Curves
 - hair 793
- modules 20
- modulus operator (%) 648
- morphing facial expressions 351
- motion
 - speeding up with Graph Editor 213
- motion capture data
 - creating clips from 261
 - description 260
 - extending the length of 263
 - redirecting 266
- Motion Path
 - Attach to Motion Path 226
- motion path animation
 - description 224
- Motion Streak
 - nParticles 998
- move
 - key points 212
 - objects 35
 - UVs 382
- Move and Sew UVs 380
- Move Snap Settings 87
- Move Tool 35, 88
 - N (normal) manipulator 96, 198
 - X, Y, Z manipulator 35
- mute track 258, 271

N

- nCaching
 - nParticles 994
- nCloth 895
 - caching 929–930
 - Collision Flag 908, 910
 - collision thickness 902, 908
 - collisions 900, 910, 930–931
 - constraints 912, 923, 939–940
 - creating 898, 900, 926
 - Dynamic Properties 923–926
 - Initial State 936, 938
 - Input Mesh Attract 933
 - Mass 923

- Max Self Collide Iterations 941
- painting properties 933–936
- Quality Settings 940–942
- Self Trapped Check 941
- Stretch Resistance 932
- Tangential Drag 924
- neck
 - joints 319
- new scene 29
- nodes
 - description 37
 - initialShadingGroup 66
 - input 66
 - nParticleShape 951
 - Nucleus 951
 - parent 54
 - pivot point 58
 - renaming 54
 - root 54
 - shape 66
 - transform 66
- Noise
 - hair 813
- non-dynamic fluid 823
 - creating 833
 - description 823
- non-hair simulation 808
- non-proportional scaling 51
- nonlinear animation
 - description 242
- normals
 - reversing 871
 - surface 164, 870
- Normals
 - Harden Edge 128
 - Soften Edge 127
- nParticle tutorials
 - creating nParticles 947
 - overview 946
 - simulating liquids 981
 - simulating smoke 959
- nParticles 945
 - Blobby Radius Scale 992
 - Bounce 955
 - Collide Width Scale 956, 989
 - collisions 950, 988

- Color 953
 - converting to polygons
 - Convert nParticles to Polygons 990
 - creating 948, 960
 - creating liquid simulations 981
 - emission rate 954
 - emitter 948, 960, 962
 - Fill Object 982
 - Friction 955
 - Ignore Solver Gravity 962
 - Incompressibility 990
 - Lifespan 964
 - Liquid Radius Scale 987
 - Liquid Simulation attributes 986
 - Motion Streak 998
 - nCaching 994
 - nParticleShape node 951
 - Nucleus nodes 951
 - Nucleus wind 977
 - Output Mesh 992
 - Radius 953, 964
 - Radius Scale 966
 - rendering 1001
 - Self Collide Width Scale 956
 - self-collisions 956
 - Shading attributes 972
 - Stickiness 955
 - Surface Tension 996
 - Threshold 993
 - tutorial 946
 - Nucleus 895, 945
 - Gravity 923
 - Gravity and Wind 977
 - Max Collision Iterations 942
 - node 951
 - Space Scale 932, 952, 985
 - Substeps 942
 - Wind Speed 923
 - Num Frequencies 855
 - Number of Sections 161
 - Circle 176
 - Number of Spans 161
 - Numeric Display 827
 - NURBS
 - applications 153
 - creating NURBS curves 155
 - editing tools 183
 - Loft Tool 176
 - sculpting NURBS surfaces 160
 - Smoothness 162
 - NURBS curves
 - hair 784
- O**
- object selection 34, 123
 - ocean
 - attributes, modifying 854
 - creating 852
 - plane 853
 - shader 852–853
 - offset 292
 - Opacity 613, 838
 - Paint Cluster Weights Tool 348
 - Sculpt Geometry Tool 167
 - Opacity Input 836
 - operators
 - assigning values to 648
 - equal to 648
 - less than 636
 - order of statements 638
 - orient constraint
 - creating 295
 - description 294
 - origin 18
 - orthographic view 42
 - Outliner 246, 249, 275
 - description 157, 182
 - parenting objects 182
 - placing clips from 255
 - Output
 - hair 784
 - Output Mesh
 - nParticles 992
- P**
- paint brushes
 - blending 572
 - Paint Effects
 - 3D objects 590, 594

- applications 609
- Brush Settings window 618
- brushes, 3D Paint 617
 - making surfaces paintable 594
 - panel 587
- Paint Effects hair 784, 801
- Paint Effects to Polygons 602
- Paint Fluids Tool 844
- Paint Fur Attributes Tool 875
 - Color Feedback 878
 - Flood 880
 - Reflection 880
 - Smooth operation 879
- Paint Hair Tool 784
- Paint Selection Tool 85, 345
- Paint Skin Weights Tool 334
- Paint Weight Menu 336
- painting
 - 2D strokes 564
 - 3D strokes 576
 - 3D strokes in scene view 579
 - bump 620
 - Density 843
 - fluid properties 843
 - Fuel 843
 - interactively 620
 - pressure sensitivity 576
 - strokes 577
 - Temperature 848
 - UV preparation 623
- pairBlend attribute 239
- panel 18
- panel layout 510, 522
- parent constraint
 - description 302
 - weighting 304
- parent node 54
- parenting
 - description 284
- particle attributes, per object and per
 - particle 548
- particle object 538
- particles
 - applications 535
 - changing form of 545
 - description 536
 - render type 545
 - streaks 546
- passive collision object 900, 928
- Passive Fill
 - hair 789
- path animation
 - blending 235
 - description 224
 - position marker 230
 - Twist attribute 234
 - UValue 228
- per object attributes 548
- Per Particle (Array) Attributes 645
- per particle attributes 548
 - how to distinguish 548
- Perlin Noise 836
- perspective view 43
- Photon Intensity 509, 512, 531
- photon maps 527, 534
 - description 508
 - visualizing 518
- Photon Refractions 530
- photons
 - Accuracy 532
 - Caustic Photons 533
 - description 508
 - Emit Photons 509
 - Global Illum Photons 509
 - Max Photon Depth 530
 - Photon Intensity 509, 512, 531
 - photon maps 534
 - Photon Refractions 530
 - Radius 532
- Pitch 858
- pivot point
 - description 57
- playback
 - animation 310
 - controls 206
 - quickening 217
 - speed 206
- Playblast 209, 217
 - description 240
- Plug-in Manager
 - Fur 862

- point constraint
 - description 292
- Point Method 816
- Points Per Hair 789
- Polar 873
- pole
 - surface 166
- Pole Vector Constraint 328
- Pole Vector XY Z 328
- polygon count 132
- Polygon Proxy Mode 189
- polygonal cylinder
 - creating 30
 - options 31
- polygons
 - component types 74
 - display settings 76
 - extruding 100
 - four-sided quads 80
 - selection settings 76
 - smoothing 81, 124
- Polygons to Subdiv 188
- Ponds 859
 - creating 860
- preferences
 - modeling 76
- preview plane 853
- primitives
 - applications 28, 30
 - creating 21, 79
 - interactive creation 77
 - modifying 180
 - NURBS sphere 49
 - NURBS torus 61
 - plane 590
 - polygonal cube 46, 79
 - polygonal cylinder 30, 47
 - types 30, 175
- procedural textures 442
 - description 445
- process 437
- projection mapping UVs 374
- PSD file format 366

Q

- quads 80
- Quality Settings 940, 942
- Quaternion rotation 239

R

- Radius 532, 613
 - Circle 176
 - nParticles 953
 - Sphere 161
- Radius Scale
 - nParticles 966
- Radius, nParticles 964
- Radius(L) 168
- Radius(U) 164
 - Paint Cluster Weights Tool 348
 - Paint Skin Weights Tool 336
- rain 546
- ramps 549, 554
- Randomization
 - hair 789
- Range Slider 206
- raytracing 503
- Redirect Tool 266
 - creating redirect control 267
 - keyframing control 269
 - positioning control 268
- reduce fur length 875
- reference layer 867, 883
- Refine Selected Components 198, 200
- reflect
 - Paint Fur Attributes Tool 880
- Reflection
 - Sculpt Geometry Tool 169
- Reflection X 614
- refractions
 - Max Trace Depth 530
 - photon refraction 530
 - refraction levels 529
- renaming clips 249
- renaming surfaces 865
- render a scene 892
- Render Current Frame 585
- Render Settings 457, 504

- render types, setting for particles 545
- Render View 451, 453, 468, 524, 551, 621, 801, 818, 890
 - Keep Image 507, 525
- rendering
 - anti-aliasing 516
 - batch rendering 458, 462
 - description 437
 - hardware 438, 463
 - image size 505, 516
 - methods 438
 - nParticles 1001
 - overview 439
 - panel layout 510, 522
 - raytracing 503
 - render settings 504
 - sampling 516
 - shading surfaces 464
 - shadows 506
 - software 438, 450
 - view region to be rendered 453
- rendering particles 554
- Replace
 - Paint Cluster Weights Tool 348
- resolution, fluid 834
- Rest Position 789
- Retain component spacing 87
- return to start time 206
- reusing clips 254
- reverse normals 871
- Revolve Tool 154
 - creating surfaces 157
 - editing surfaces 157
- rewind 206
- rgbPP attribute 548
 - use in expressions 645
- ribs
 - joints 322–323
- ridges
 - subdivision surface 199
- rigid bodies 554
 - active 558
 - applications 535
 - collisions 554
 - constraints 555
 - passive 559

- rigid skinning 341
- Roll 858, 873
- root
 - skeleton 324
- root node 54
- Rotate Tool
 - description 36
 - X, Y, Z manipulator 36
 - X, Y, Z, manipulator 92
- Rotate UVs 383
- rotating objects
 - options 50
- rotInterpolation 239
- runtime expressions 646
 - assigning rgbPP in 647

S

- sampling 516
- Save As 611
- Save Scene 26
- saving files 26, 40, 59
- scale
 - non-proportional 51
- Scale
 - Paint Skin Weights Tool 337
- scale, multiplying by percentage 642
- scaling clips 259
- scene painting view 588
- scene view vs Render View 451
- Scraggle
 - fur setting 885
- scrub 207
- Sculpt Geometry Tool 160, 163
 - basic techniques 163
 - preparing surfaces 161
 - sculpting terrain 592
- Section Radius 543
- Selected Position 839
- selection
 - border edges 91
 - Convert Selection 99
 - faces 84
 - hierarchy 57
 - hierarchy, object, component 56
 - masks and modes 22, 56

- mode 22
- objects 24, 34, 123
- Paint Selection Tool 85
- polygon components 89
- Select Border Edge Tool 92
- UVs 377–378
- vertices 89
- Self Collide Width Scale 956
- self collisions
 - nParticles 956
- Self Shadow 840
- Self Trapped Check 941
- self-shadowing
 - fluid 840
- semicolon terminator
 - expressions 628
- Set Driven Key 221
 - Graph Editor 223
- Set Initial Position 558
- Set Key 207
- Set Preferred Angle 326
- Set Rest Position 796
- Set Start Position 792, 797
- Set to Dynamic 830, 844, 848
- sewing UVs 379
- shaded mode 51, 82, 603
- shader
 - fluid 836
- Shading
 - hair 805
- Shading attributes
 - nParticles 972
- shading material
 - assigning 67
 - Blinn 68
- shading networks
 - description 445
- shading surfaces 464
- shadowing
 - attribute for spotlights 890
- shadows 506, 523
 - description 494
- Shelf 22
 - preset brushes 578
 - saving preset brushes 571
- Show/Hide Editor 22
- Size Rand 838
- Size X, Size Y 612, 620
- size, container 834
- skeletons 314
 - creating 282
 - hierarchy 281
 - moving group node 328
 - parenting models 284
 - posing and animating 326
 - root 324
- skin 330
- skin weights 332–333
 - modify 335
- slice 844
- Slices Per Voxel 847
- Smear 619
- Smear, 3D Paint 619
- Smooth
 - hair 795
 - Paint Fur Attributes Tool 879
 - Paint Skin Weights Tool 337
- Smooth Shade All 82, 441
- smooth skin
 - influence object 337
 - unnatural deformation 333
- smoothing polygons 81, 124
- Snap Mode 22
 - Snap to Grid 87
 - Snap to Points 177
- Soft Modification Tool 133
- Soften Edge 127
- Software renderer
 - description 451
- software rendering 450
- solo track 257
- source clip 246
- Source In 247
- Source Out 247
- Space Scale 932, 952, 985
- spans, displaying 162
- Specular
 - hair 805
- Specular Power
 - hair 805
- Specularity 857

- sphere primitive 49
 - options 64
- sphrand function, use with random
 - color 647
- spinal column
 - joint chain 319
- spine
 - parenting arms and legs 324
- Split Polygon Tool 110
- spotlight
 - aiming 489
 - Cone Angle 492
 - description 488
- spotlights 888
- Spottyness 838
- Stamp Depth (3D) 848
- Stamp Spacing 616
- Standard Mode 195
- Start Position 789
 - setting 792
- starting Maya 15
- statements
 - expression order 638
- static channels
 - remove 216
- Static Grid 848
- Status Line 20–21
 - expanding and collapsing items 22
- Stickiness
 - nParticles 955
- Stiffness
 - hair 791, 796
- Stiffness Scale
 - hair 796
- Streak render type 546
- Stretch Resistance 932
- stroke
 - attributes 581
 - attributes, Global Scale 583
 - attributes, Gravity 584
 - attributes, Turbulence 595
 - description 577
 - Stroke Refresh 588
- strokes
 - deleting 597
 - polygon conversion 602
- rendering 584, 598, 601
 - tubes 580
- Stylus Pressure 168, 576, 613
- Sub Segments
 - hair 802
- Subdiv Proxy
 - Crease Proxy Edge Tool 130
 - description 124
 - toggle display 126
 - using 125
- subdivision surfaces
 - converting polygons 187
 - cutting 194
 - Extrude Face 191
 - extrusion manipulator 193
 - selecting faces 191
 - selecting vertices 193
 - Split Polygon Tool 189
- Substeps 942
- subvolume manipulators 844
- summary track
 - description 256
- surface
 - material 67
 - normals 164
 - pole 166
- surface render 832
- Surface Tension
 - nParticles 996
- surfaces normals 870
- symmetry 84
- syntax, errors 637

T

- Tail Fade 547
- Tail Size 547
- Tangent
 - Graph Editor 218
 - handles in Graph Editor 212
- Tangential Drag 924
- target
 - blend shape 350
- Temperature
 - fluid property 824
 - painting 848

- template brush
 - description 566, 578
 - modifying settings 566
 - resizing 566
- template display 61, 315
 - applications 61
 - description 61
 - untemplating objects 65
- template layer 883
- texture map
 - applications 359
 - assigning 364
 - description 359
 - procedural 442
 - UVs 361
- Texture Opacity 836
- Texture Scale 837
- Texture Time 841
- Texture Type 836
- textures
 - cloth 445, 447
 - interactive 3D placement 480
 - procedural 445
- texturing, fluid 836
- The Art of Maya 3
- Thickness 902–908
- Thinning
 - hair 802
- Threshold
 - nParticles 993
- ticks
 - red key 207
- time
 - expressions 635
 - predefined variable 628
 - value at different frames 628
- Time option 440
- Time Slider 206
- time warps 275
- tip width 885
- Toggle Texture Borders 381
 - applications 381
- Tool Settings 611
- Toolbox 31
 - transformation tools 33, 87
- torus
 - options 61
- track 247
 - description 247
 - muting 258
 - soloing 257
 - view area 248
- Track Tool 45
 - description 45
- transform nodes
 - not used for particle expressions 645
- transparency, skin 332
- Trax Editor
 - character sets 248
 - clip library 270
 - clip manipulator 259
 - cycling a clip 265
 - Graph Anim Curves 253
 - Load Selected Characters 246
 - motion capture data 260
 - mute track 258
 - Offset attribute 266
 - panel layout 243
 - repositioning clips 251
 - scaling clips 259
 - solo track 257
 - summary track 256
 - time warps 275
 - Toolbar 249
 - track 247
- triplanar projection 374
- tubes
 - attributes 605
 - Creation 601
 - editing attributes 570
 - Flower 601
 - Leaves 601
- Tumble Tool
 - description 45
- turbulence 827
- Turbulence
 - options 596
- Twist 328
 - hair 794
- Twist attribute 234
- Two Panes Stacked 620

U

- U Count
 - hair 789
- UI Elements 611
- undo 37, 593
- Update on Stroke 621
- Use Depth Map Shadows 491, 494
- user interface 15–16
- user preferences
 - restoring 12
 - saving 11
- UV Texture Editor
 - applications 369
 - converting a selection 382
 - description 369
 - Dim Image 375
 - Display Image 377, 382
 - Display Unfiltered 385
 - Move and Sew UVs 380
 - Rotate UVs 383
 - Toggle Texture Borders 381
 - viewing UVs 371
- UVs
 - Automatic Mapping 374
 - description 361
 - image range 372
 - mapping 373
 - modifying 377
 - moving 382
 - preparing for painting 623
 - projection mapping 374
 - selecting 377–378
 - sewing 379
 - shells 380
 - viewing 369, 371

V

- V Count
 - hair 789
- Value
 - Paint Cluster Weights Tool 348
- Vector renderer
 - description 451

- velocity
 - controlling in particles 545
- Velocity, fluid property 824
- vertices
 - subdivision surface 193–194
- viewing rendered frames 461
- viewing shaded objects 51
- views
 - orthographic 42
 - perspective 43
 - shaded 51
 - wireframe 54
- Visor 246, 275
 - description 255
 - placing clips from 255
 - preset brushes 578, 594
- volume axis field 539
- Volume Axis field 968
- volume pixels 826
- Volume Shape 539
- voxels 826, 834

W

- Wakes
 - creation 859
- water
 - open 852
- Wave Length Max 855
- Wave Peaking 856
- Wind Speed 923
- Wireframe on Shaded mode 82
- workspace 17
- world space coordinates 96

X

- X-Ray 83, 332
- X, Y, Z
 - axes 19
 - axis indicator 19
 - origin 18
- X, Y, Z Resolution 834
- X, Y, Z Size 834
- X, Z ground plane 579