
3D Studio MAX® R2

Introduction

Welcome to 3D Studio MAX R2

Welcome to 3D Studio MAX, the program that makes it easy for you to animate everything! You can translate your creative ideas into astonishing results.

This guide takes a conceptual, *task-based* approach. The emphasis is on what you want to accomplish, and strategies for how to use the program to meet those ends. Detailed reference information and step-by-step procedures are provided online. You can also learn the program through a series of printed and online tutorials.

This introduction presents information on the following topics:

- What's new in 3DS MAX and where to find information about it.
- How to use 3DS MAX printed and online documentation.
- How to connect to the 3DS MAX Web site to receive updated documentation and new tutorials.

What's New in 3D Studio MAX R2.0?

The second release of 3D Studio MAX is a powerhouse package for film and video production. It is balanced for image quality, speed, freedom of animation, and world-class special effects, all within an accessible user environment.

The following is a quick overview of major enhancements. See the Online Reference for details about all new features and see the referenced chapters for new feature concepts.

General Modeling—Ch. 6, 7, 15

- You can cut, copy, and paste modifiers from one object to another.
- There are many new parametric objects and compound objects.
- There are many new modifiers and space warps.

Spline and Loft Modeling—Ch. 8, 9

- The Section Shape object creates cross-sectional splines based on any geometry.
- Enhanced Text tool now has justification, kerning, leading, and multi-line text.
- New spline modifiers include fillet, chamfer, trim, and extend.
- You can now animate vertices and tangents.
- Use the Constant Cross-Section option for uniform loft widths at path corners.

NURBS Modeling—Ch. 10

- NURBS surfaces are a powerful way to model 3D geometry. A model can be a single surface or a collection of curves and surfaces.
- You can use NURBS curves as an alternate form of spline for such things as lofting and motion paths.
- NURBS models include both curve and surface sub-objects.
- Dependent NURBS surfaces maintain modeling relationships as you edit or animate the model. Dependent surfaces include bland, transform, mirror, offset, extrude, cap, lathe, ruled, and U-loft surfaces.
- You can animate NURBS curves, surfaces, and control vertices.

Mesh Modeling—Ch. 12

- The Preserve modifier controls edge lengths, face angles, and the volume of an object.
- New modifiers include tessellate, extrude, cap, and modify faces.
- The Delete Mesh modifier supports parametric and animated deletion of the active selection.
- You can animate vertices, faces, and edges.

Particle Systems—Ch. 14

- Particle Spawn creates new particles as they collide or die.
- MetaParticles generate blobby, metaball sprays.
- You can instance any object as a particle type.
- You can make particles bounce off objects, obey physical forces, crawl across surfaces, swim along splines, or explode.

Cameras and Lighting—Ch. 17, 18

- You can control specular and diffuse illumination to isolate or remove specular highlights.
- Omni lights now support 360 degree shadows and image projection.
- The Sun System places and animates the sun according to geographic location and time.

- Lens sizes are now coordinated to film formats for control of camera aperture width.
- You can control field-of-view by either horizontal, vertical, or diagonal measure.
- The Camera Match Utility can adjust cameras so the camera view of the scene matches its background image.
- Orthographic cameras can view and animate orthogonal and axonometric views.

New Materials—Ch. 20, 21

- You can display up to 24 material sample windows, magnify samples, and display custom sample objects.
- New Raytrace material and map generates accurate ray-traced reflections and refractions. Options include translucent, fluorescent, and black light effects.
- You can selectively ray-trace only relevant parts of the scene, or material properties.
- Use Reflection Dimming for realistic control of reflections in shadow.
- Apply the Camera Map modifier for perspective matching of background composites.
- Crop, scale, and position bitmaps within the Material Editor.
- There are many new map types including Adobe Photoshop and Premier filters.
- You can now assign up to 15 material effects channel IDs (G-buffer).

Inverse Kinematics—Ch. 23, 24

- You can assign a procedural IK controller for real-time IK solutions.
- The Auto Boning function creates Bone Systems with IK constraints from any hierarchy.
- The controllers Attachment, Link, and Surface Position work with IK.

Animation and Dynamics—Ch. 25-28

- You can blend and weight multiple morph targets for facial animation and lip synching.
- You can use the Motion Capture panel for real-time motion input with key reduction and sampling options.
- Animated objects can fall, slide, collide, and bounce using dynamic simulation.
- There are many new animation controllers.

Rendering and Video Post—Ch. 29-31

- You can animate lens flares, and create highlights and glows with gradients.
- You can add nebula, fire, and electric effects.
- Rack Focus can adjust camera depth-of-field and rack-focus effects.
- Adobe Premiere Plug-ins are available for animated filters and transitions in Video Post.
- New Image motion blur for rendering rapid motion.

Productivity and Precision Tools

- You can control precision with 3D object snaps that support sub-object selections and gizmos for accurate modeling.
- You can drag and drop materials, maps, and geometry into the scene for quick workflow.
- Zoom and Pan background images in the viewports.
- Undo and Redo are now unlimited.
- The new MAXScript language helps you develop custom tools without compiling.
- You can rapidly access commands with a custom stroke recognition system.

Documentation for 3D Studio MAX

The documentation set for 3D Studio MAX combines printed and online material.

- *3D Studio MAX User's Guide*—Covers fundamental concepts and strategies for using 3D Studio MAX.
- *3D Studio MAX Online Reference*—Contains complete reference material describing the features of 3D Studio MAX, as well as many step-by-step procedures.
- *3D Studio MAX Tutorials*—Introduces 3D Studio MAX through guided exploration.
- *Learning 3D Studio MAX Online*—Online tutorials that guide you through new features of 3D Studio MAX.

How to Use This Guide

This guide is designed to present concepts, strategies, and techniques for using 3D Studio MAX without delving into the details of every command. Use this guide to explore how to work with 3DS MAX and use the online reference to check details and procedures for all commands.

Read off-line—Whenever and wherever you find it convenient. Make notes in the margins for things you want to check when you get to the office.

Read online—In front of your computer. Experiment with the interface as you read about it. Follow references to online information and try a hands-on tutorial example.

How to Use the Online Reference

The 3DS MAX Online Reference replaces a conventional reference manual. Details of every command and dialog are included, along with a substantial set of procedures and new tutorials.

The Online Reference is part of an overall plan for continuing information delivery. It complements the *User's Guide* and *Tutorials*. It offers a quick reference for users who want detailed information while at the computer.

To open the Online Reference, do one of the following:

- On the Help menu, choose Online Reference or Learning 3D Studio MAX.
- Press F1.

Use the Contents panel, to the left of the online window, to browse and search the Online Reference.

Contents—Click topics in the contents hierarchy to open different levels of topics, and select topics for display.

Index—Lists all the help topic titles alphabetically in index format. You can search the help topic titles for certain keywords.

Search—Generates a list of all words in all help topics. When you enter the words you want to search for, you see a list of all topics containing those words.

Often, the quickest or most effective way to get the information you want is to use the Index panel, which searches the titles of every topic. For even finer detail, use the Search panel, which searches every word in every topic.

How to Update Documentation from the 3DS MAX Web Site

The Online Reference is a living information system that can be extended and updated via links to the 3DS MAX web site. Look for regular additions to the web site, delivering the following information:

- The latest information about 3DS MAX developments and plug-ins.
- New tutorials on 3DS MAX features and techniques to make you more productive.
- Updates and revisions to the documentation to keep you informed.

To open the 3DS MAX web site:

1. Make sure you are connected to your internet service provider.

2. On the Help menu, choose:
Connect To Support And Information.

3DS MAX opens your default web browser and connects to the 3DS MAX web site. Click web site links for support, training, and news.

You can manually connect to the 3DS MAX web site at:

http://www.ktx.com/3dsmax

How to Contact Kinetix

We're interested to hear your views and wishes about the 3D Studio MAX documentation set, print and online. Please let us know what you like in the documentation, what you don't like, what works, what doesn't. Tell us what you'd like to see beefed up or cut back, or done a different way. We look forward to hearing—learning—from you soon.

Please contact us using the feedback form on our home page:

http://www.ktx.com

Chapter

1

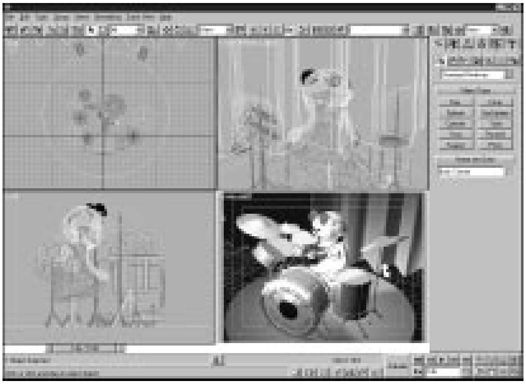
Overview of 3D Studio MAX

You use 3D Studio MAX to quickly create professional quality 3D models, photorealistic still images, and film quality animation on your PC.

This chapter presents a these brief topics designed to help you quickly start using 3D Studio MAX.

- Getting started
- Setting scenes
- Modeling, animating and rendering scenes
- Using the 3D Studio MAX window
- Managing files used by 3D Studio MAX

Getting Started with 3D Studio MAX



Once you have installed 3D Studio MAX, you launch it from the Start menu or use any other Windows method. The figure above shows the 3DS MAX application window with a scene file loaded.

3DS MAX is a single document application, meaning you can work on only one scene at a time. However, you can launch more than one copy of 3DS MAX and open a different scene in each copy. Launching additional copies of 3DS MAX requires a lot of RAM. For the best performance you should plan to launch one copy and work on one scene at a time.

Note: Launching multiple copies of 3DS MAX is not supported on Windows 95.

Modeling Objects

The cornerstone of 3DS MAX is an integrated modeling environment. You perform 2D drawing, 3D modeling, and animation within the unified workspace. Your modeling, editing and animation tools are always available in the command panels and toolbar.

Material Design

3DS MAX provides a sophisticated Material Editor in a floating window. You use the Material Editor to create realistic materials by defining hierarchies of surface characteristics. The surface characteristics can represent static materials or be animated for special effects.

Lights and Cameras

You create lights with various properties to illuminate your scene. The lights can cast shadows, project images, and create volumetric effects for atmospheric lighting.

The cameras you create have real-world controls for lens length, field of view, and motion control such as truck, dolly, and pan.

Animation

You begin animating your scene at any time by clicking the Animate button. Click the button again to move back and forth between modeling and animating. You can also perform animated modeling effects by animating the parameters of objects in your scene.

You use Track View to control animation. Track View is a floating window where you edit animation keys, set up animation controllers, or edit motion curves for your animated effects.

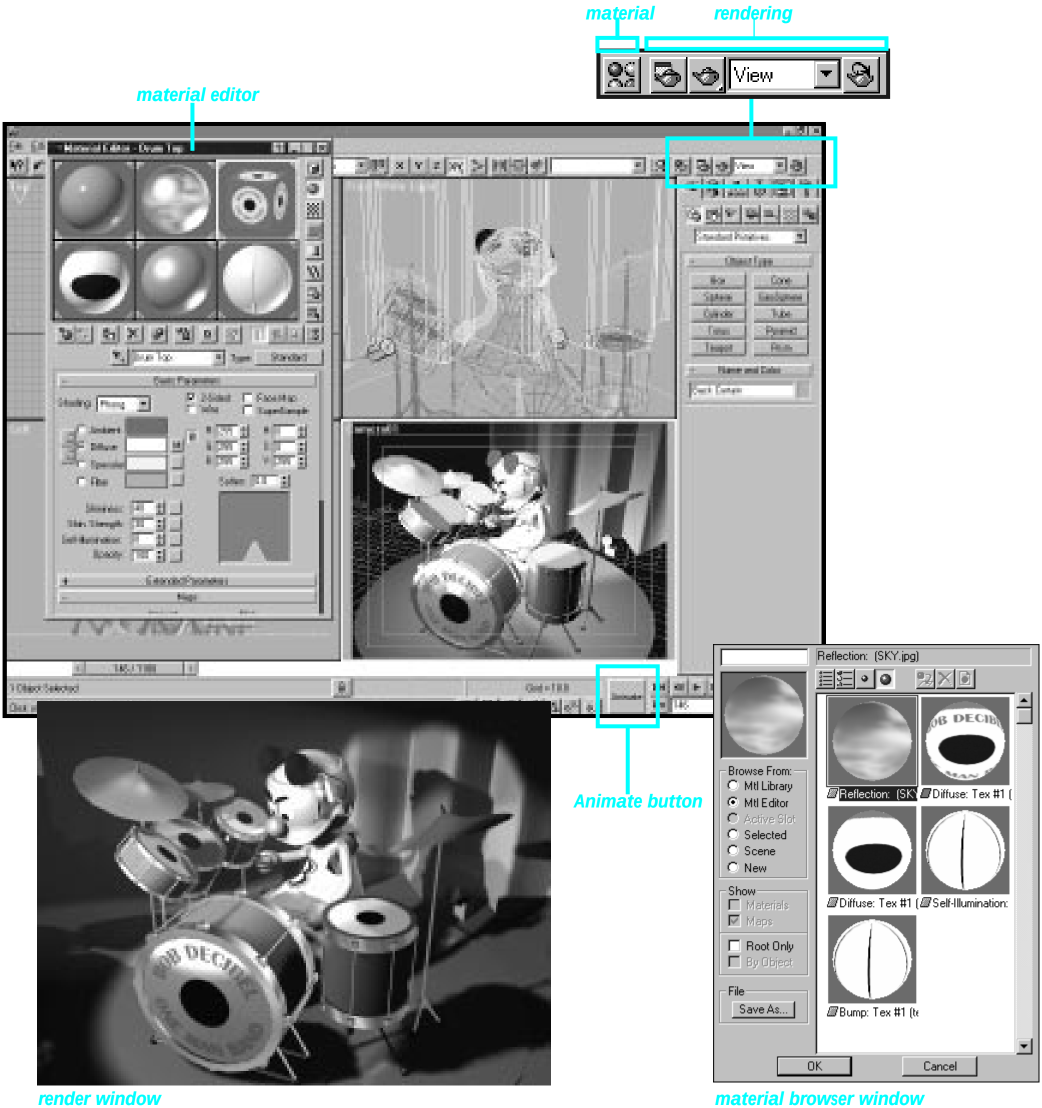
Rendering

The 3DS MAX renderer includes features such as selective ray tracing, analytical antialiasing, motion blur, volumetric lighting, and environmental effects. You can also render and composite multiple views with animations stored on disk using the Video Post window.

If your workstation is part of a network, 3DS MAX supports network rendering to distribute rendering jobs over multiple workstations.

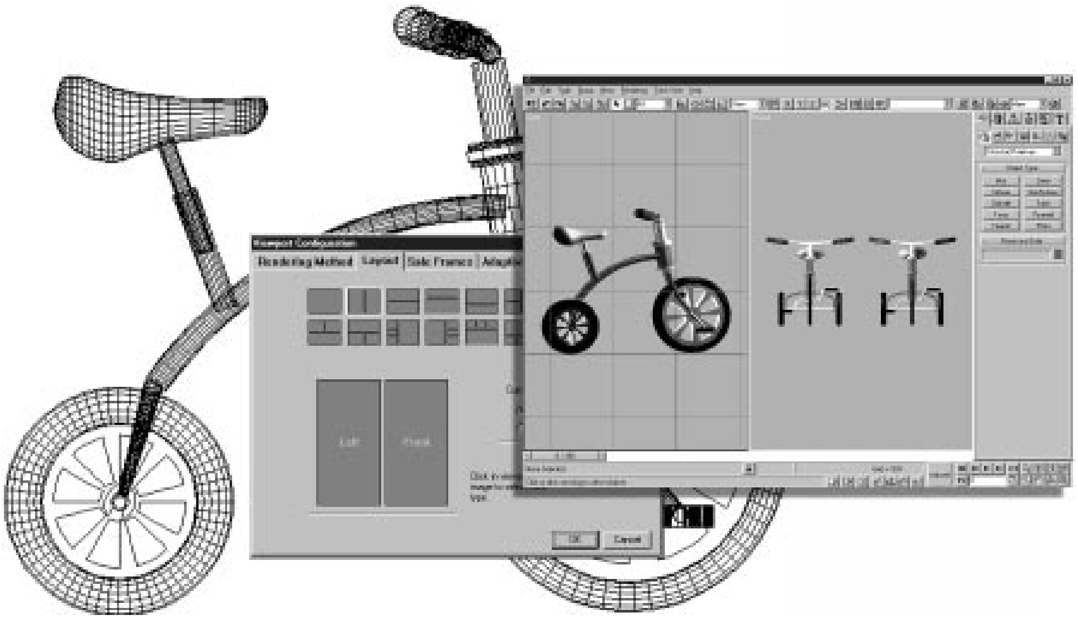
A Typical Project Workflow

The next six topics explain the basic procedures for creating 3DS MAX scenes.



material
rendering
View
material editor
Animate button
Reflection: (SKY.jpg)
Browse From:
Mtl Library
Mtl Editor
Active Slot
Selected
Scene
New
Show
Materials
Maps
Root Only
By Object
File
Save As...
Reflection: (SKY
Diffuse: Tex #1 (
Diffuse: Tex #1 (
Self-Illumination:
Bump: Tex #1 (t
OK
Cancel
render window
material browser window

Setting Up Your Scene



You start with a new unamed scene when you launch 3D Studio MAX. You can also start a new scene at any time by choosing New or Reset from the File menu.

Setting the System Unit

The system unit setting, on the Preferences dialog, determines how 3DS MAX stores distance information in your scene. The setting also determines the range for round-off error. Consider changing the system unit value only when you model very large or very small scenes.

Choosing a Unit Display

You choose a system of unit display on the Units Setup dialog. Choose from Metric, Standard US, and Generic methods, or design a custom measuring system. You can switch between different systems of unit display at any time.

Setting Grid Spacing

Set spacing for the visible grid on the Home Grid panel of the Grid and Snap Settings dialog. You can change grid spacing at any time.

See chapter 5, “Precision and Drawing Aids” for information about the three previous headings.

Setting Viewport Display

The default four viewports in 3DS MAX represent an efficient and popular screen layout. Set options in the Viewport Configuration dialog to change viewport layout and display properties.

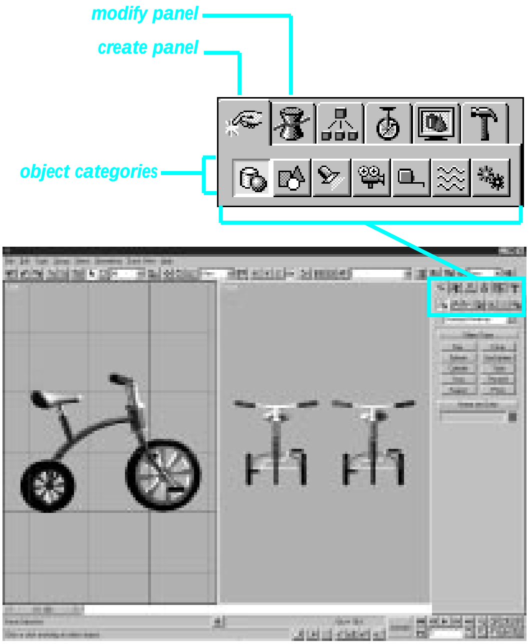
See chapter 2, “Viewing and Navigating 3D Space” for more information.

Saving Scenes

Save your scene frequently to protect yourself from mistakes and loss of work. See “Backing Up and Archiving Scenes,” later in this chapter.

Modeling Objects

You model objects in your scene by creating standard objects, such as 3D geometry and 2D shapes, and then applying modifiers to those objects. 3D Studio MAX includes a wide range of standard objects and modifiers.



Creating Objects

You create objects by clicking an object category on the Create panel and then clicking, or dragging, in a viewport to define the object's creation parameters. 3DS MAX organizes the Create panel into seven basic categories: Geometry, Shapes, Lights, Cameras, Helpers, Space Warps, and Systems and each category often contains multiple subcategories you can choose from.

See chapter 6, "Creating and Modifying Objects."

Selecting and Positioning Objects

You select objects by clicking or dragging a region around them. You can also select objects by name or other properties such as color or object category.

After selecting some objects, you position them in your scene using the transform tools of Move, Rotate, and Scale. You can also use alignment tools to precisely position objects.

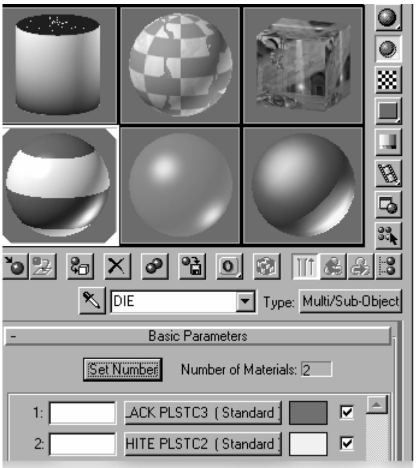
See chapter 3, "Selecting Objects", chapter 4, "Positioning Objects", and chapter 5, "Precision and Drawing Aids."

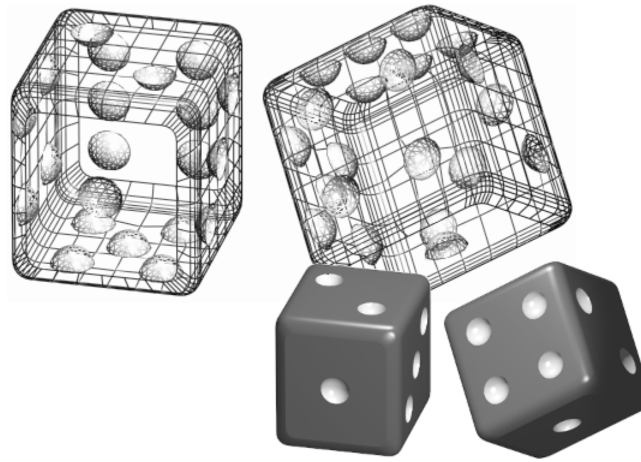
Modifying Objects

You sculpt and edit objects into their final form by applying modifiers from the Modify panel. The modifiers you apply to an object are stored in a stack. You can go back at any time and change the effect of the modifier, or remove it from the object.

See chapter 6, "Creating and Modifying Objects."

Designing Materials





You use the Material Editor to design materials and maps to control the appearance of object surfaces. Maps can also be used to control the appearance of environmental effects such as lighting, fog, and the background.

Basic Material Properties

You set basic material properties to control such surface characteristics as default color, shininess, and level of opacity. You can create realistic, single color materials using just the basic properties.

See chapter 20, “Designing Basic Materials.”

Using Maps

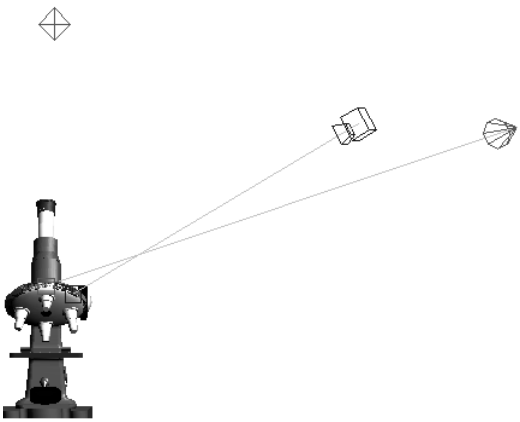
You extend the realism of materials by applying maps to control surface properties such as texture, bumpiness, opacity, and reflection. Most of the basic properties can be enhanced with a map. Any image file, such as one you might create in a paint program, can be used as a map, or you can choose procedural maps that create patterns based on parameters you set.

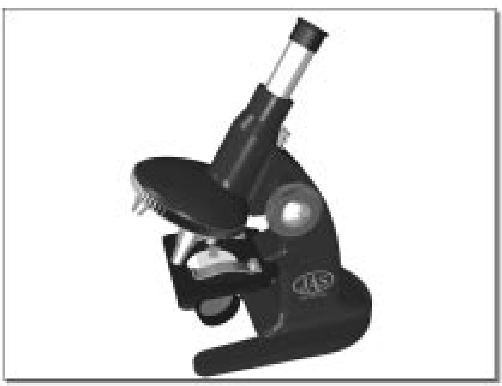
3DS MAX also includes a ray-trace map for creating accurate reflections and refraction. See chapter 21, “Designing Mapped Materials.”

Viewing Materials in the Scene

You can view the effect of materials on objects in a shaded viewport, but the display is just an approximation of the final effect. Render your scene to view accurate materials.

Placing Lights and Cameras





You place lights and cameras to complete your scene much the same way lights and cameras are placed on a movie set prior to filming.

Default Lighting

Default lighting evenly illuminates the entire scene. Such lighting is useful while modeling but it is not very artful or realistic.

Placing Lights

You create and place lights from the Light category of the Create panel when you are ready to take control of your scene lighting.

3DS MAX includes three light types: omni, spot, and directional lights. You can set a light to any color and even animate the color to simulate dimming or color shifting lights. All of these lights can cast shadows, project maps, and use volumetric effects.

See chapter 17, “Lighting Your Scene.”

Viewing Lighting Effects in the Scene

When you place lights in a scene, the default lighting turns off and the scene is illuminated by the lights you create. The illumination you see in a viewport is just an approximation of the true lighting. Render your scene to view accurate lighting.

Placing Cameras

You create and place cameras from the Camera category of the Create panel. Cameras define viewpoints for rendering, and cameras can be animated to produce cinematic effects such as dollies and truck shots

See chapter 18, “Setting Up Cameras.”

Animating Your Scene

You can animate almost anything in your scene. Click the Animate button to turn it on, drag the Time Slider, and change something in your scene to create an animated effect.

Controlling Time

3D Studio MAX starts each new scene with 100 frames for animation. Frames are a way of measuring time, and you move through time by dragging the Time Slider. You can also open the Time Configuration dialog to set the number of frames used by your scene and the speed at which the frames are displayed.

Animating Transforms and Parameters

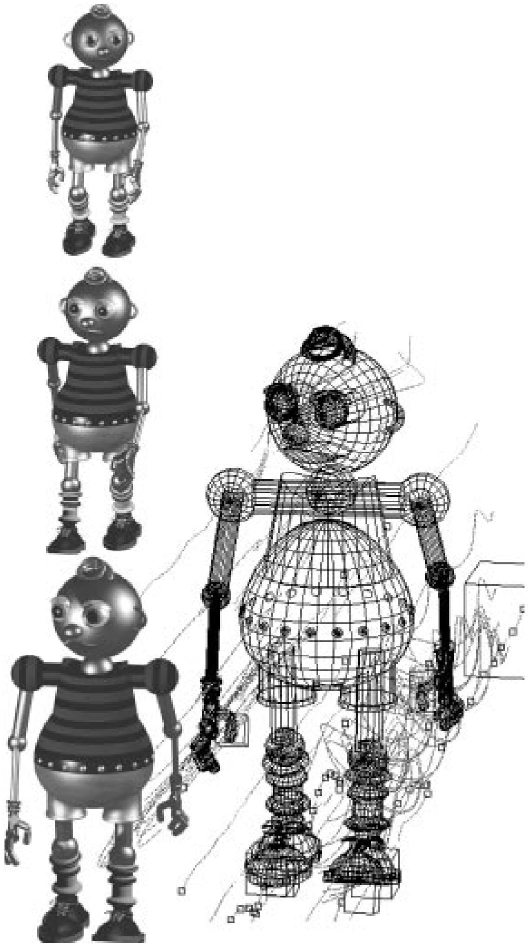
When the Animate button is on, 3DS MAX creates an animation key whenever you transform an object or change a parameter. To animate a value over a range of frames you need only to specify the value at the first and last frames of the range. 3DS MAX calculates the values for all of the frames in between.

See chapter 22, “Animation Concepts and Methods.”

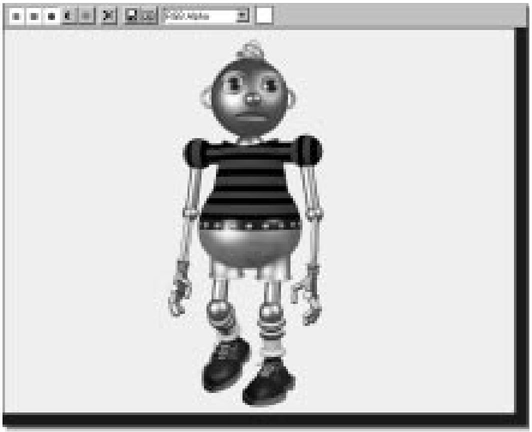
Editing Animation

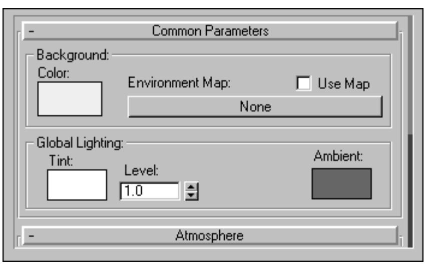
You edit your animation by opening the Track View window or by changing options on the Motion panel. Track View is like a spreadsheet that displays animation keys along a time line. You edit the animation by changing the keys. You can also display the animation as a series of function curves that graphically show how a value changes over time.

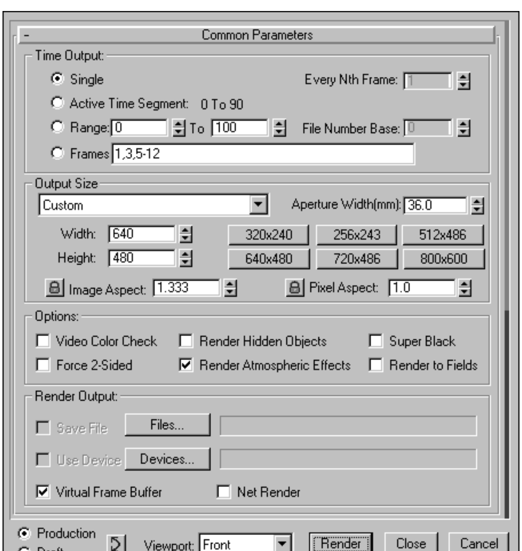
See chapter 25, “Basic Track View Use.”



Rendering Your Scene







You use the rendering features of 3D Studio MAX to define an environment and to produce the final output from your scene.

Defining Environments and Backgrounds

Rarely do you want to render your scene against the default black background. Open the Environment dialog to define a background for your scene or to set up effects such as fog.

See chapter 29, “Creating Atmospheres and Environments.”

Setting Rendering Options

To set the size and quality of your final output, you can choose from many options on the Rendering dialog. You have full control over professional grade film and video properties as well as effects such as reflection, antialiasing, shadow properties, and motion blur.

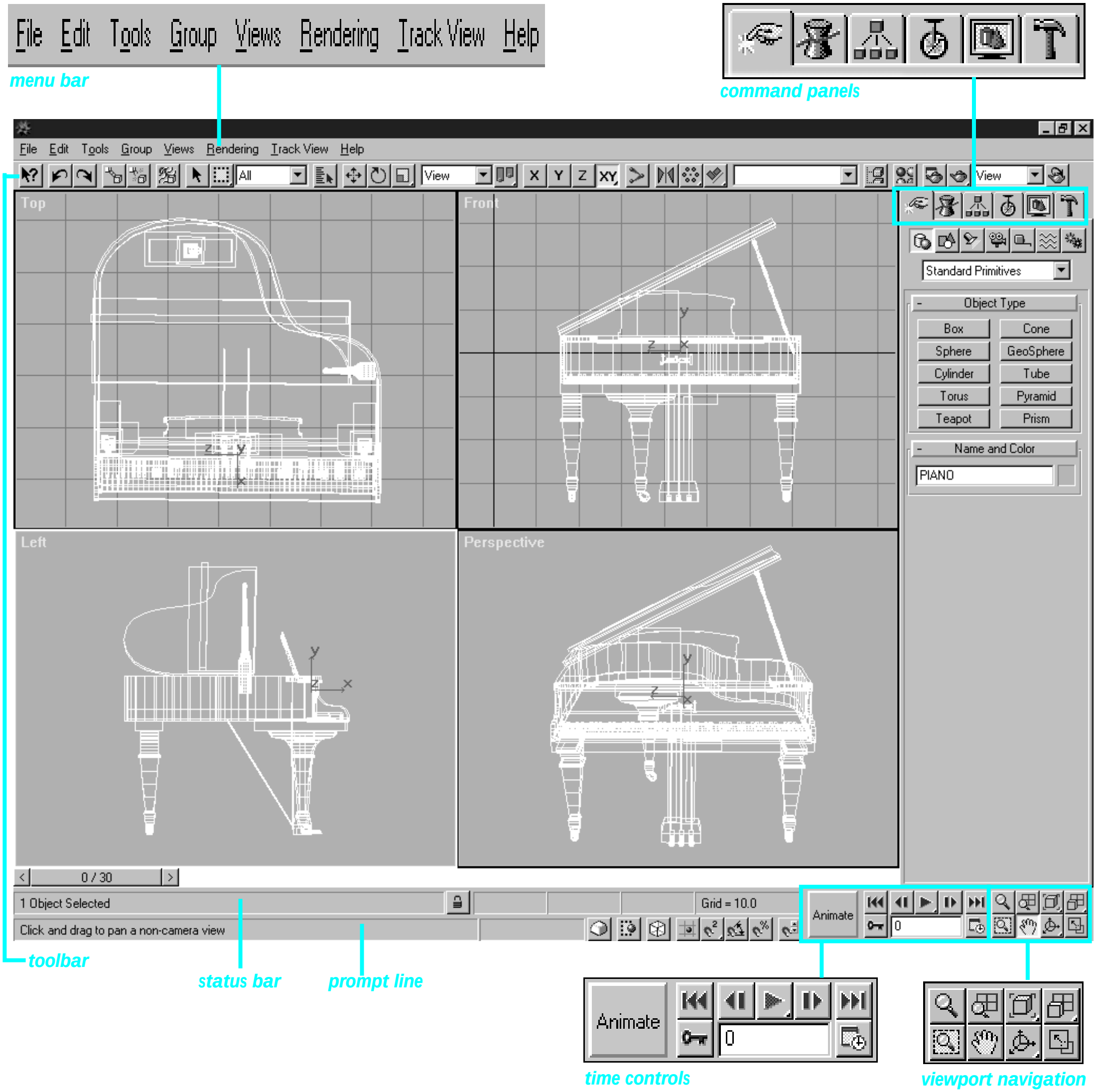
Rendering Images and Animation

You render a single image by setting the Renderer to render a single frame of your animation. You specify what type of image file to produce and where 3DS MAX stores the file.

Rendering an animation is the same as rendering a single image except that you set the renderer to render a sequence of frames. You can choose to render an animation to multiple single frame files or to popular animation formats such as *.flc* or *.avi*.

See chapter 30, “Rendering Scenes.”

The 3D Studio MAX Window



Most of the 3D Studio MAX window is occupied by the viewport area where you view your scene. The remaining areas of the window hold controls and show status information.

Menu bar—A standard Windows menu bar with typical File, Edit, Views, and Help menus. Special 3DS MAX menus include:

- Tools contains duplicates of many of the tool bar commands.
- Group contains commands for managing grouped objects.
- Rendering contains commands for rendering, Video Post, and the environment.
- Track View contains commands for opening and managing Track View windows.

Toolbar—Buttons for selecting, transforming, and rendering geometry in a scene. Some toolbar buttons are shortcuts for menu commands; others appear only on the toolbar.

Time controls—The Animate button turns on animation mode. The other controls navigate through time and play back an animation.

Command panels—Six panels that hold most of the modeling and animation features.

- Create holds all object creation.
- Modify holds modifiers and editing.
- Hierarchy holds linking and inverse kinematics parameters.
- Motion holds animation controllers and trajectories.
- Display holds object display controls.
- Utilities holds miscellaneous utilities.

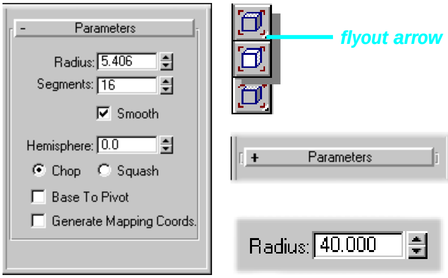
Status bar and prompt line—These two lines display prompts and information about your scene and the active command. They also contain system toggles controlling selections, precision, and display properties.

Viewports—You can display from one to four viewports. Multiple viewports can show multiple views of the same geometry.

Viewport Navigation buttons—The button cluster at the lower-right of the 3DS MAX window contains controls for zooming, panning, and navigating within the viewports.

Special Controls

3D Studio MAX uses some special user interface controls.



Flyouts

A flyout is similar to a menu, except that the items are buttons rather than text. A flyout button is indicated by a small arrow in the lower-right corner. Clicking a button on a flyout makes the button active.

Rollouts

Rollouts are areas in the command panels and dialogs that you can expand (roll out) or collapse (roll in) to manage screen space.

To open and close rollouts:

- Click the rollout title bar to toggle between expanded and collapsed.

Scrolling Panels

Sometimes a command panel or dialog is not large enough to display all of its rollouts. In this case, a pan cursor appears over the nonactive parts of the panel.

To scroll panels:

1. Place the pointer over an empty area of a panel to display the pan cursor.

2. When the pointer becomes a , drag the panel up or down.

A thin scroll bar also appears on the right side of the scrolling panel. You can use the pointer to drag the scroll bar as well.

Spinners

A spinner is a control for numeric fields. You can click or drag the spinner arrows to change the value in the field.

To change a value using a spinner:

- Click the spinner to increment or decrement the value.
- Drag up to increase the value, or drag down to decrease it.
- Press CTRL while you drag to increase the rate at which the value changes.
- Press ALT while you drag to decrease the rate at which the value changes.
- Right-click on a spinner to reset the field to its minimum value.

Entering Numbers

You can change a numeric value by a relative offset by typing **R** or **r** followed by the offset amount.

For example, a radius field shows 70 and you highlight it:

- If you type **R30,** 30 is added to the radius and the value changes to 100.
- If you type **R–30**, 30 is subtracted from the radius and the value changes to 40.

Controls and Color

The 3DS MAX interface uses color cues to remind you what state the program is in.

Red for animation—The Animate button and the border of the active viewport turn red when you are in Animate mode.

Green for modal function buttons—When you turn on a button that puts you in a generic creation or editing mode, the button turns green.

You can exit a functional mode by clicking another modal button. Other exit methods supported by some buttons include right-clicking in a viewport, or clicking the modal button a second time.

Blue for IK mode—When you turn on IK (Inverse Kinematics) mode the button turns blue. This indicates that the transform buttons behave differently when IK is active.

Yellow for special action modes—When you turn on a button that alters the normal behavior of other functions, the button is highlighted in yellow. Common examples of this behavior include, sub-object selection and locking your current selection set.

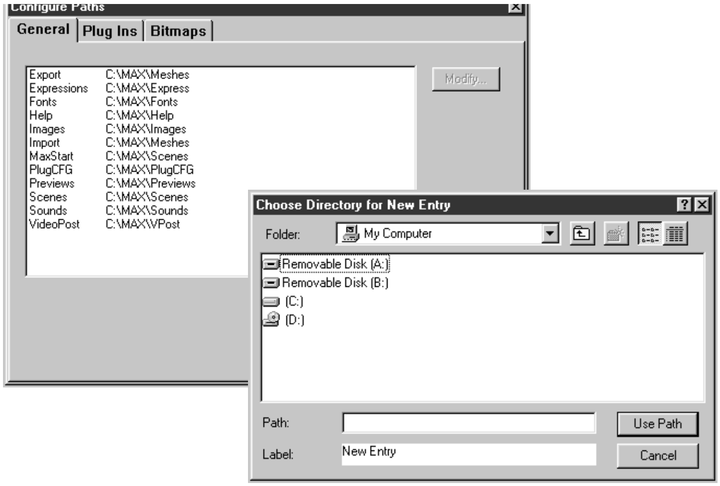
Undoing Actions

You can easily undo changes you make to your scene and your viewports. 3D Studio MAX manages separate Undo buffers for both the scene objects and each viewport.

Use the Undo and Redo buttons or Undo and Redo commands in the Edit menu to reverse the effects of most scene operations. Most things you do in 3DS MAX can be undone.

You can also undo actions by using the Hold and Fetch commands on the Edit menu. Choose Edit > Hold to save a copy of your scene in a temporary file. Then choose Edit > Fetch to discard your current scene and revert to the held scene at any time.

Managing Files



3D Studio MAX supports many types of files for working with plug-ins, image maps, models from other programs, rendering images and animations, and of course saving and opening your MAX scene files.

Configuring File Paths

The default locations that 3DS MAX searches for all file types are specified in the Configure Paths dialog.

You can choose to open and save files in any path location. The Configure Paths dialog contains three panels for the general categories of 3DS MAX support files.

Setting General File Paths

The General panel contains paths for the standard 3DS MAX support files. You can specify one path for each of the following file types.

File Type	Path Used for
Export	Saving scenes you export to other file formats
Expressions	Loading and saving text files used by expression controllers
Fonts	Loading font files
Help	Loading online help files
Images	Saving and viewing image files
Import	Loading files you import from other programs
MaxStart	Loading *maxstart.max* that provides initial 3DS MAX scene settings
PlugCFG	Loading plug-in configuration files
Previews	Saving and viewing animation files rendered with preview renderer
Scenes	Opening and saving for 3DS MAX scene files
Sounds	Loading sound files
Video Post	Opening and saving Video Post queues

Setting Plug-In File Paths

Most of what you see in 3D Studio MAX is implemented as a plug-in. This means you can change and extend the functionality of 3DS MAX by adding new plug-ins from Kinetix or third-party developers.

Two plug-in path entries are set during 3DS MAX installation:

- The "Standard MAX plug-ins" entry sets the path to where the plug-ins that ship with 3DS MAX are located.
- The "Additional MAX plug-ins" entry sets a default path where third-party plug-ins should be located.

You tell 3DS MAX where to find additional plug-in files by adding path entries to the Plug Ins panel. If you place all of your plug-ins in a single directory, plug-in file management quickly becomes a mess. That's why 3DS MAX supports multiple entries in the Plug Ins panel.

Setting Bitmap File Paths

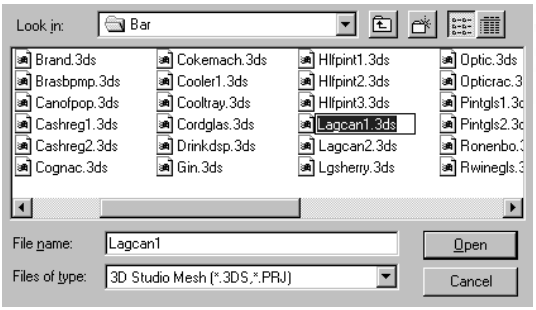
The Bitmaps panel contains multiple path entries that 3DS MAX searches for image files. Image files are used for many purposes in 3DS MAX such as material and map definition, spotlight projections, and environment affects. It is easy to imagine a single 3DS MAX scene using many image files.

You identify the default locations of for image files by adding paths to the Bitmaps panel. Use this technique to identify the standard directories used by most of your scenes.

3DS MAX also reads the full path of any image you load, and saves it as the image filename. When 3DS MAX needs to reload the image, it searches for it in the following order:

- The path saved with the image file.
- The directory of the current scene.
- All sub-directories below the current scene.
- The paths listed in the Bitmaps panel, starting at the top of the list.

Importing, Merging, and Replacing Scenes



You realize great productivity gains when you reuse work by combining geometry from 3D Studio MAX scenes or other 3D programs. 3DS MAX supports this technique with the Import, Merge, and Replace commands.

Importing Geometry from Other Programs

Use Import to bring objects from other 3D programs into 3DS MAX. The types of files that you can import are listed in the Files of Type list in the Select File To Import dialog.

Depending on the file type you choose, you may have options available for that import plug-in. Check the 3DS MAX online documentation for details on each import plug-in.

Merging Scenes Together

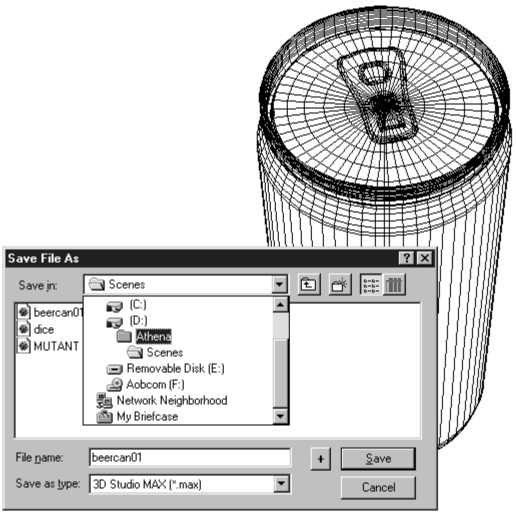
Use Merge to combine multiple 3DS MAX scenes into a single large scene. When you merge a file you can select which objects to merge. If objects being merged have the same name as objects in your scene, you have the option to rename or skip over the merged object.

Replacing Scene Objects

Use Replace to replace objects in your scene with objects in another scene that have duplicate names. Replace is useful when you want to set up and animate your scene with simplified objects, and then replace the simple objects with detailed objects before rendering.

The Replace dialog looks and functions the same as Merge, except that it only lists objects that have the same name as objects in your current scene

Backing Up and Archiving Scenes



Bad things can happen, so it makes sense to regularly back up and archive your work.

Saving Incremental Files

One convenient method is to save incremental copies of your scenes. This method creates a history of your work process.

If you select the Increment On Save option in the Preferences dialog, the current scene is renamed each time you save by incrementing a two-digit number appended to the end of the file. For example, if you open a file named *myfile.max* and then save it, the saved file is named *myfile01.max*. Each time you save the file its name is incremented, producing the files *myfile02.max*, *myfile03.max*, etc.

You can also use Save As to manually increment the filename with a two-digit number by clicking the increment button on the Save As dialog.

Using Autobackup

You can automatically save backup files at regular intervals by setting the Auto Backup options on the Preferences dialog. The backup files are named *autobak#.mx*, where the # is a number from 1 to 9. You can rename *autobak#.mx* with a *.max* file extension and load it like any other scene file.

Archiving a Scene

3D Studio MAX scenes make use of many different files. When you want to exchange scenes with other 3DS MAX users or store scenes for archival purposes you need to save more than just the scene file.

Use the Archive command to pass the scene file and any bitmap files used in the scene to a PKZIP compatible archiving program.

Chapter

2

Viewing and Navigating 3D Space

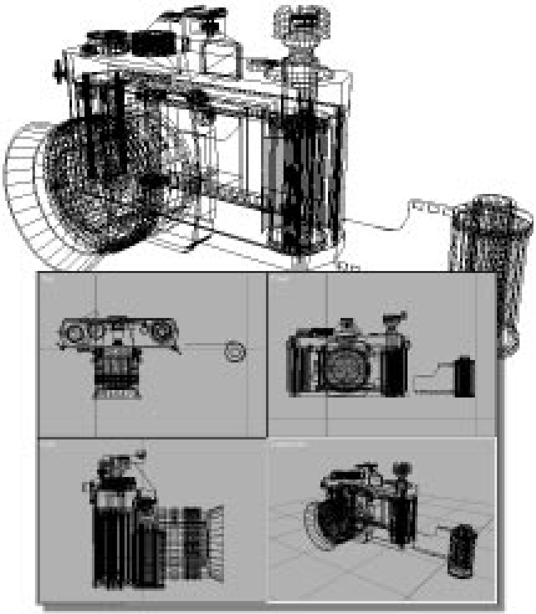
Everything you create in 3D Studio MAX is located in a three-dimensional world. You have a variety of options for viewing this enormous stage-like space—from the details of the smallest object to the full extent of your scene.

Using the view options discussed in this chapter you move from one view to another, as your work and imagination require. You can fill your screen with a single, large viewport, or set multiple viewports to track various aspects of your scene. For exact positioning, flat drawing views are available, as are 3D perspective and axonometric views.

You navigate 3D space by adjusting the position, rotation and magnification of your views. You have full control over how objects are rendered and displayed on the screen.

This chapter discusses how to organize viewports and navigate through 3D space. See the Online Reference for details about viewport commands.

General Viewport Concepts



Viewports are openings into the three-dimensional space of your scene, like windows looking into an enclosed garden or atrium. But viewports are more than passive observation points. While creating a scene, you can use them as dynamic and flexible tools to understand the 3D relationships among objects.

At times you might want to look at your scene through a large, undivided viewport, giving you a "picture window" view of the world you're creating. More often, you use multiple viewports, each set to a different orientation.

If you want to move an object around, you might do this in a top viewport, looking directly down on the object as you move it. At the same time, you could be watching a shaded perspective viewport to see when the object you're moving slides behind another. Using the two windows together, you can get exactly the position and alignment you want.

You also have pan and zoom features available in either view, as well as grid alignment. With a few mouse clicks or keystrokes, you can reach any level of detail you need for the next step in your work.

Another way to use viewports is to place a camera in your scene and set a viewport to look through its lens. When you move the camera, the viewport tracks the change. You can do the same thing with spotlights.

Active Viewport

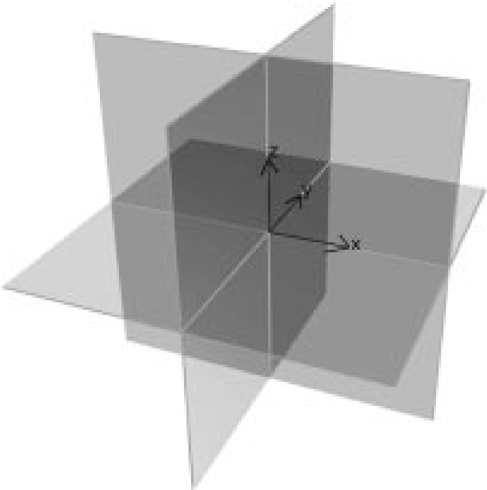
One viewport, marked with a white border, is always *active*—the active viewport is where commands and other actions take effect. Only one viewport can be in the active state at a time. Other viewports are set for observation only; unless disabled, they simultaneously track actions taken in the active viewport.

Saving the Active Viewport

You can save the view in any active viewport and later restore it by choosing Save Active View and Restore Active View on the Views menu. One view can be saved for each of the following view types: Top, Bottom, Left, Right, Front, Back, User, Perspective.

As an example, while in the Front view, you choose Save Active Front View, and then zoom and pan that view. You then activate the Top viewport, choose Save Active Top View, and then click Zoom Extents. You return to the Front view, and choose Restore Active Front View to return to its original zoom and pan. At any time, you can activate the Top viewport, and then choose Restore Active Top View to restore its saved view.

Home Grid: Views Based on the World Coordinate Axes



The grids you see in the viewports represent one of three planes that intersect at right angles to one another at a common point called the *origin*. Intersection occurs along three lines—the *world coordinate axes*, X, Y, and Z—familiar from geometry as the basis of the Cartesian coordinate system.

Home Grid

In 3D Studio MAX, the three planes based on the world coordinate axes are called the *home grid*, the basic reference system of the 3D world of 3DS MAX.

To simplify the positioning of objects, only one plane of the home grid is visible in each viewport. The figure shows all three planes as they would appear if you could see them in a single perspective viewport.

Axes, Planes, and Views

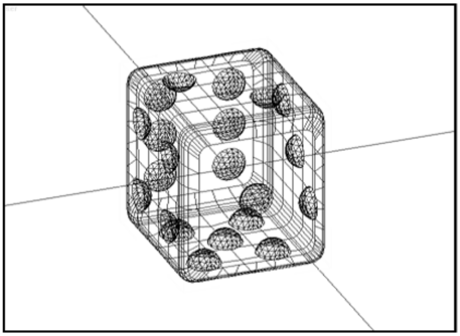
Two axes define each plane of the home grid. In the default perspective viewport, you are looking across the XY plane (*ground plane*), with the X axis running left-to-right, and the Y axis running front-to-back. The third axis, Z, runs vertically through this plane at the origin.

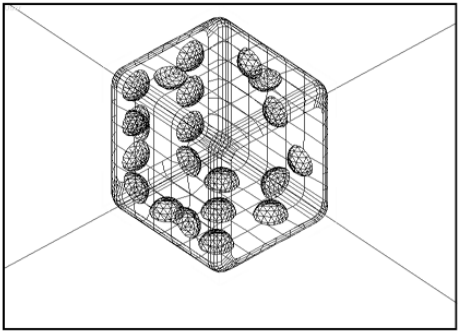
Home Grid and Grid Objects

The home grid is aligned with the world coordinate axes. You can turn it on and off for any viewport, but you can't change its orientation.

For flexibility, the home grid is supplemented by *grid objects*—independent grids you can place anywhere, at any angle, aligned with any object or surface. They function as "construction planes" you can use once and discard or save for reuse. See chapter 5, "Precision and Drawing Aids".

Understanding Axonometric Views





Axonometric views refer to the parallel projection of a 3D object onto a drawing surface (or computer screen). If the object is inclined so that three sides are visible, the projection maintains horizontal and vertical scale, but distorts diagonals and curved lines. Orthographic views, rotated user views, and isometric views are all forms of axonometric views.

Orthographic Views

Whether produced on computer or paper, most 3D design relies on *orthographic* views for accurate description of objects and their positioning. Maps, plans, cross-sections, and elevations are all orthographic views. In familiar terms, you might think of these views as "flat" or "straight-on," or as "looking at right angles."

Orthographic views are two-dimensional, each defined by two world coordinate axes. Combinations of these axes produce three pairs of orthographic views: top, bottom; front, back; left, right.

User Views

The label *User* appears on rotated axonometric viewports. In such views parallel lines remain parallel, no matter how extreme the foreshortening. You get a sense of three-dimensional relationship without the single viewpoint implied by perspective views.

You can change any view to a User view by doing one of the following:

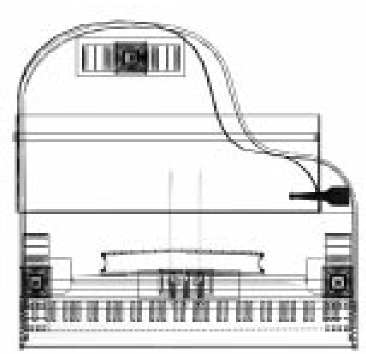
- Press **U** as a keyboard shortcut.
-  Use Arc Rotate to rotate an orthographic view. The viewport label changes to *User* as you do this.
- Right-click the label of the viewport, then choose Views > User.
- Use the Layout panel in the Viewport Configuration dialog to change the viewport type. for details.

Isometric Views

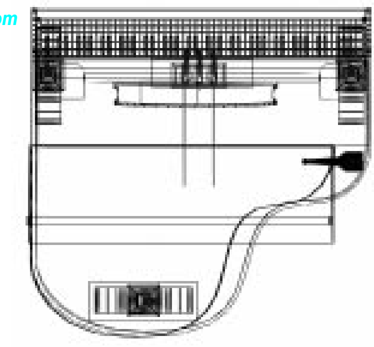
Isometric is special case of axonometric, with the sides of the object equally inclined to the screen, producing equal foreshortening along the edges.

You can produce an isometric view in 3D Studio MAX by rotating the home grid in the viewport. But you cannot produce an *isometric drawing* view, which maintains scale along the diagonals and therefore has no foreshortening.

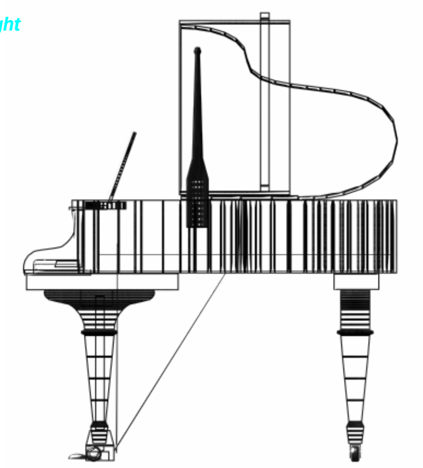
Top



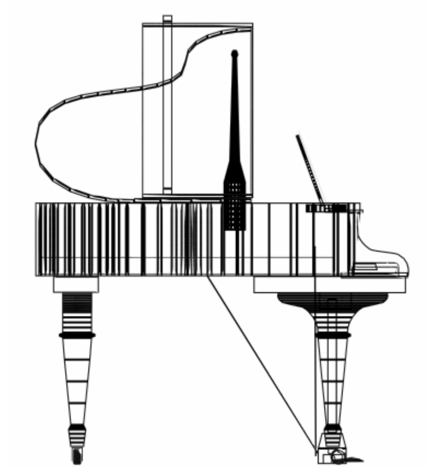
Bottom



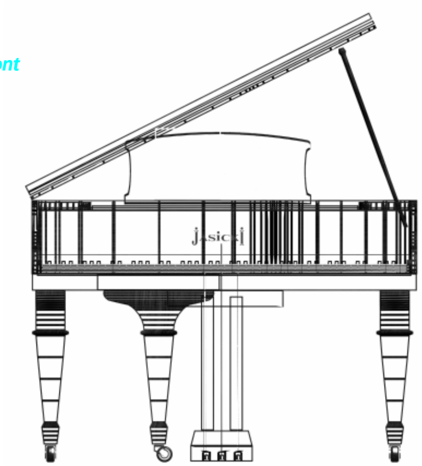
Right



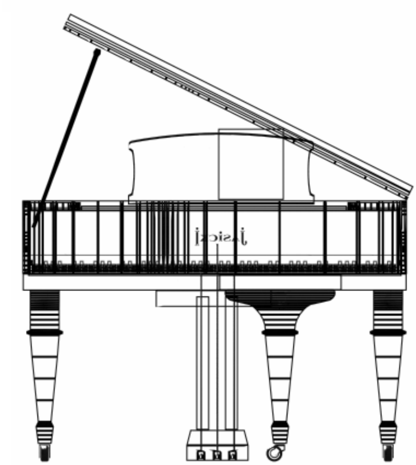
Left



Front

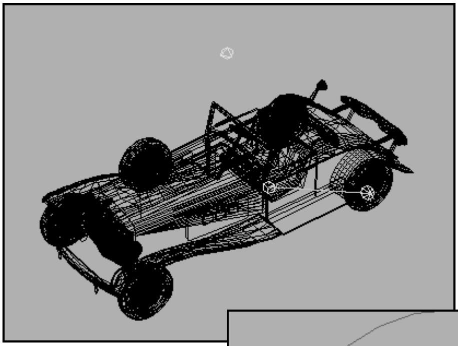


Back

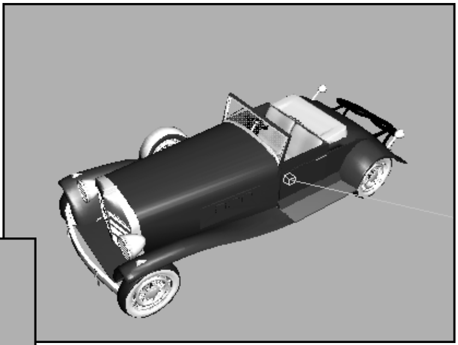


Piano mesh as seen in orthographic views

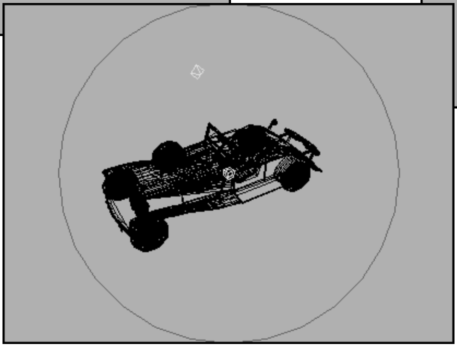
Understanding Perspective Views



Perspective View



Camera View



Spotlight View

Perspective views most closely resemble human vision. Objects appear to recede into the distance, creating a sense of depth and space. For most 3D computer graphics, this is the view used in the final output. It's what the client sees on screen or on the page.

Perspective View

A perspective viewport, labeled *Perspective,* is one of the startup viewports in 3DS MAX. You can change any active viewport to this "eye-like" point of view by pressing the keyboard shortcut **P**.

Camera View

Camera view requires that you first create a camera object in your scene. You can change the active viewport to a camera view by pressing the keyboard shortcut **C** and then selecting from a list of cameras in your scene.

A camera viewport tracks the view through the lens of the selected camera. As you move the camera (or target) in another viewport, you see the scene move accordingly.

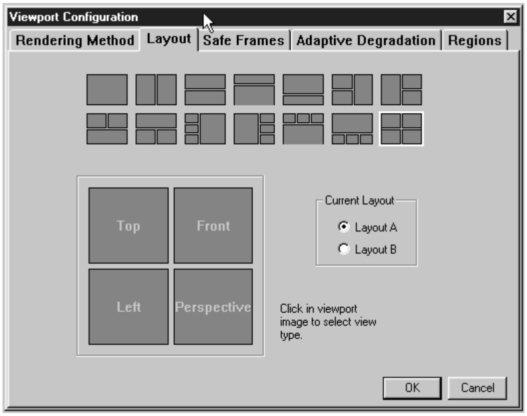
Note: If you select Orthographic Projection on a camera's Parameters rollout, that camera produces an axonometric view like a User view.

See chapter 18, "Setting Up Cameras."

Spotlight View

Spotlight view works much like a targeted camera view. You first create the spotlight and then set the viewport to that spotlight. See chapter 17, "Lighting Your Scene."

Setting Viewport Layout



3D Studio MAX defaults to a two-over-two arrangement of viewports. Thirteen other layouts are possible, but the maximum number of viewports on the screen remains four.

Using the Layout panel of the Viewport Configuration dialog, you can set up two different layouts and customize the viewports in each. You can then switch between these layouts as you work. Your choices are saved with the file.

Changing the View Type

As you work, you can quickly change the view in any viewport—for example, switching from front view to back view. You can use either of two methods: menu or keyboard shortcut.

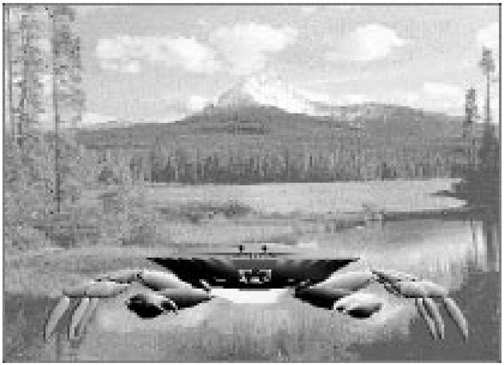
- Right-click the label of the viewport you want to change and click Views. Then, click the view type that you want.
- Click in the viewport you want to change, and then press one of the keyboard shortcuts in the following table.

See the Online Reference for details about all viewport layout and view type options.

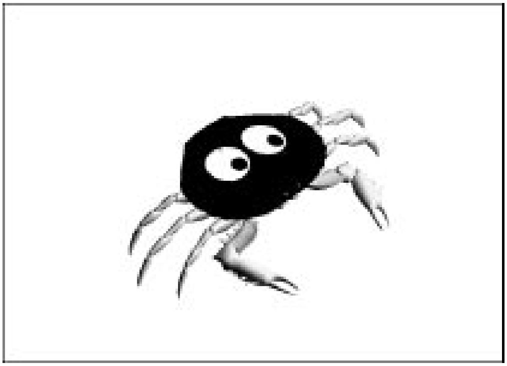
Key	View type
T	Top view.
B	Bottom view.
F	Front view.
K	Back view.
L	Left view.
R	Right view.
C	Camera view. If your scene has only one camera, or you select a camera before using this keyboard shortcut, that camera supplies the view. If your scene has more than one camera, and none are selected, a list of cameras appears. See chapter 18, “Setting Up Cameras.”
$	Spotlight view. Works like Camera view. See chapter 17, “Lighting Your Scene.”
P	Perspective view. Retains viewing angle of previous view.
U	User (axonometric) view. Retains viewing angle of previous view.
G	Grid view. Automatically changes to the active grid type. If you first activate a grid object, the view switches to an orthographic top view of that object. If you don’t activate a grid object, the view switches to the ground plane—identical to a Top view. See chapter 5, “Precision and Drawing Aids.”
E	Track view. Displays the same view as a Track View window. See chapter 25, “Basic Track View Use.”
None	Shape view. Use menu method. Automatically aligns view to the extents of a selected shape and its local XY axes. See chapter 8, “Creating Spline Shapes.”
W	Toggles the active viewport between full screen and the current layout.

Controlling Viewport Rendering

You can choose from multiple options to display your scene. You can display objects as simple boxes to render them with smooth shading and texture mapping. If you want, you can choose a different display method for each viewport.



Background displayed in front viewport



Texture map displayed in perspective viewport

Using Viewport Rendering Controls

Viewport rendering options are found on the Rendering Methods panel of the Viewport Configuration dialog. Using this panel you choose a rendering level and any options associated with that level. You can then choose whether to apply those settings to the active viewport or all viewports.

The rendering level you choose is determined by your need for realistic display, accuracy, and speed. For example, Box Mode display is much faster than Smooth Shading with Highlights. The more realistic the rendering level, the slower the display speed.

After choosing a rendering level, you can set a number of rendering options. Different options are available for different rendering levels.

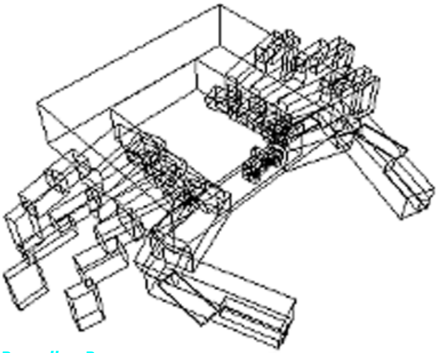
Note: Viewport rendering has no effect on final renderings produced by clicking Render Scene.

Rendering Methods and Display Speed

The various rendering methods not only affect the quality of your view display, they can also have a profound effect on display performance. Using higher quality rendering levels and realistic options slows display performance.

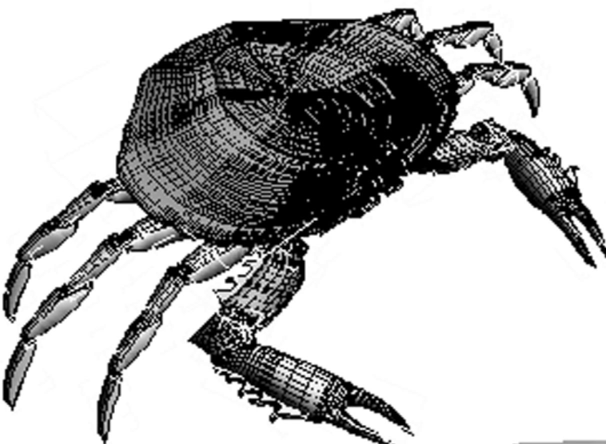
After setting a rendering method, you can choose additional options that adjust pisplay performanceOne of these controlls, Adaptive Degradation speeds up display performance when you use realistic rendering levels.

See the Online Reference for details on all viewport rendering options.

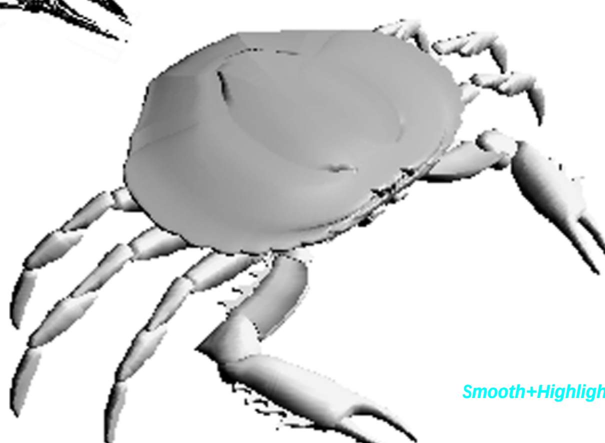


Bounding Box

Wireframe

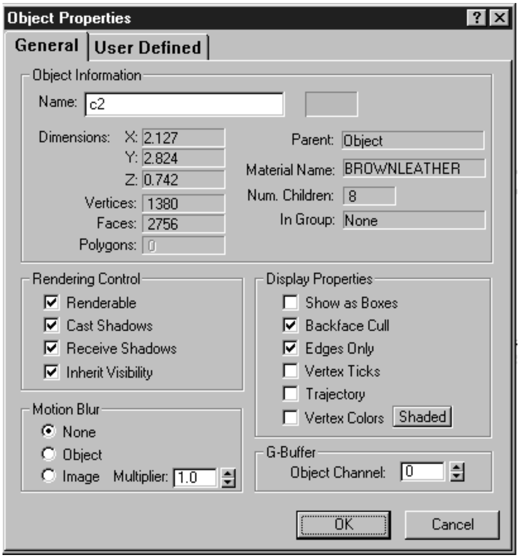


Edged Faces



Smooth+Highlights

Controlling Display Performance



3D Studio MAX contains a number of controls to help you adjust display performance—the balance between quality and time in displaying objects.

Depending on your needs, you might give up some display speed to work at higher levels of rendering quality, or you might choose to maximize display speed by using Wireframe or Bounding Box display. Which method you choose depends on your preferences and the requirements of your work.

Display Performance Controls

You use display performance controls to determine how objects are rendered and displayed.

Viewport Prefences—The Viewport panel on the Prefences dialog contains options to fine tune the performance of the viewport display software. See the Online Reference for details about these options.

How objects are rendered—The Viewport Configuration dialog contains a panel labeled Adaptive Degradation that dynamically drops the rendering level on display performance. You set the parameters controlling the trade-off between display quality and display speed.

How objects are displayed—Right-click an object to see its Display Properties. These options affect display performance much the same way as viewport rendering options. For example, selecting Vertex Ticks for an object with a lot of vertices will slow performance.

Which objects are displayed—One way to increase display speed is *not* to display something. You use the Hide and Freeze features on the Display panel to change the display state of objects in your scene.

Hide and Freeze features also affect final Rendering and Video Post output. See chapter 30, “Rendering Scenes and Animations,” and chapter 31, “Using Video Post.”

Setting Adaptive Degradation

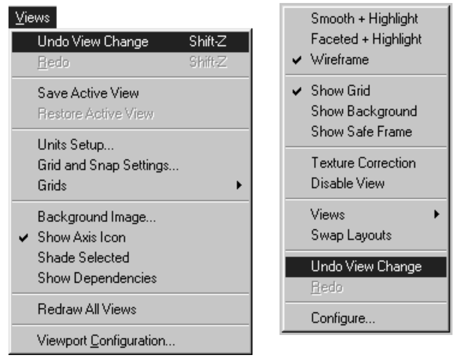
Adaptive Degradation dynamically adjusts your rendering levels to maintain a desired level of display speed. You have direct control over how much “degradation” occurs and when it occurs.

Active and General Degradation use the same choices as the viewport Rendering Levels panel. Active Degradation controlls rendering in the active viewport while General Degradation controlls rendering in all other viewports.

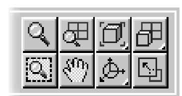
The selected levels determine which rendering levels 3DS MAX falls back to when it cannot maintain the desired display speed. You can choose as many levels as you want but you are advised to choose only one or two levels for each type of degradation.

Using Standard View Navigation

To navigate through your scene, use view navigation buttons located at the lower-right corner of the 3D Studio MAX window. All view types, except for Camera and Spotlight views, use a standard set of view navigation buttons.



Location of viewport Undo/Redo command



Standard view navigation commands

Button Operation

Clicking standard view navigation buttons can produce two results:

- Executes the command and returns to your previous action.
- Activates a view navigation mode.

  You can tell that you are in a mode because the button remains selected and is highlighted in a green background color. This mode remains active until you right-click or choose another command.

  While in a navigation mode, you can activate other viewports of the same type, without exiting the mode, by clicking in any viewport.

Undoing Standard View Navigation Commands

You use Undo View and Redo View commands to undo and redo standard view navigation commands without affecting other viewports or the geometry in your scene. These commands are found in the Views menu, and in the menu displayed when you right-click a viewport label.

Undo View and Redo View are separate from Undo and Redo on the Edit menu or the toolbar. 3DS MAX maintains separate Undo/Redo buffers for scene editing and for each viewport.

The View Undo/Redo buffer stores your last 20 view navigation commands for each viewport. You can step back through the Undo/Redo buffer until you have undone all of the stored view-navigation commands.

Zooming, Panning, and Rotating Views

When you click one of the view navigation buttons, you can change these basic view properties:

View magnification—Zooming

View position—Panning

View rotation—Rotating

Zooming a View

Click Zoom or Zoom All and drag in a viewport to change view magnification. Zoom changes only the active view while Zoom All simultaneously changes all non-camera views.

If a perspective view is active, you can also click Field of View (FOV). The effect of changing FOV is similar to changing the lens on a camera. As FOV gets larger you see more of your scene and perspective becomes distorted, similar to using a wide-angle lens. As FOV gets smaller you see less of your scene and the perspective flattens, similar to using a telephoto lens.

Zooming a Region

Click Zoom Region to drag a rectangular region within the active viewport and magnify that region to fill the viewport. Zoom Region is available for all standard views except the Perspective view. In a perspective view Zoom Region is replaced by Field-of-View.

Zooming to Extents

Click the Zoom Extents or Zoom Extents All flyouts to change the magnification and position of your view to display the extents of objects in your scene. Your view is centered on the objects and the magnification changed so the objects fill the viewport.

- The Zoom Extents buttons zoom to the extents of all visible objects in the scene.
- The Zoom Selected buttons zoom to the extents of the current selection.

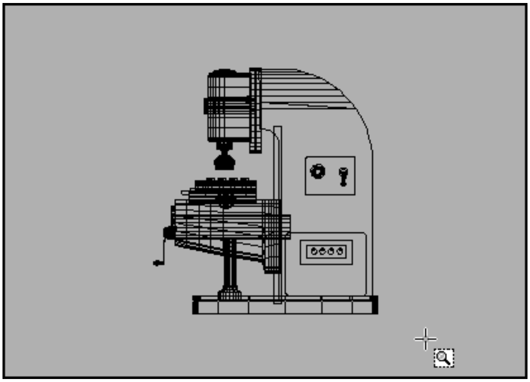
Panning a View

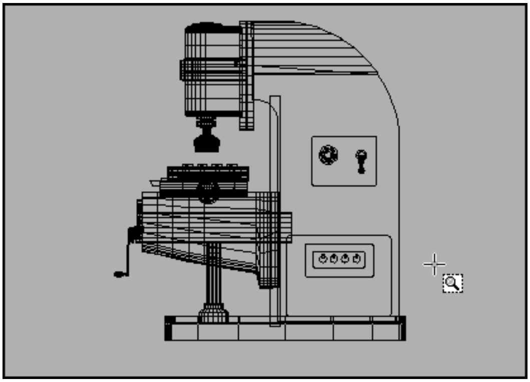
Click Pan and drag in a viewport to move your view parallel to the viewport plane.

Rotating a View

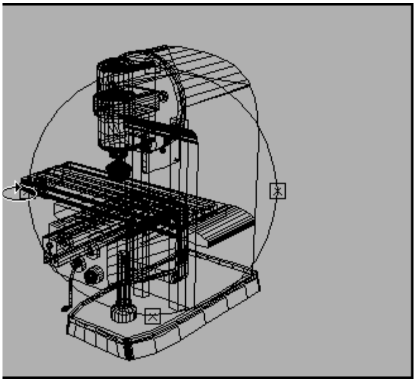
Click Arc Rotate to rotate your view around a center point. When you rotate an orthogonal view, such as a Top view, it is converted to a User view.

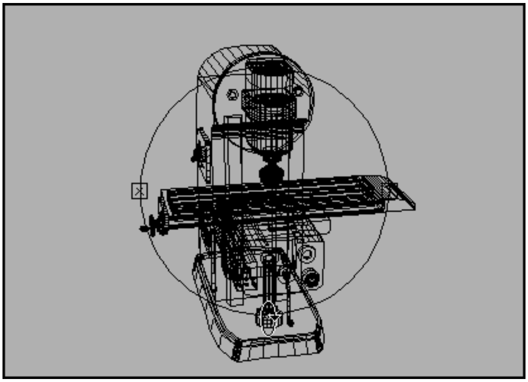
- Click the Arc Rotate button to rotate about the view center. If objects are near the edges of the viewport they may rotate out of view.
- Click Arc Rotate Selected to rotate about the center of the current selection. The selected objects remain at the same position in the viewport while the view rotates around them.





Before and after zooming a viewport



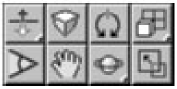


Before and after rotating a viewport

Navigating Camera and Spotlight Views

Original view

Rotated view



The Camera and Spotlight view navigation buttons are the same with a few exceptions. The buttons are visible when a viewport with a Camera or Spotlight view is active. The Camera and Spotlight view navigation buttons do more than adjust your view. They transform and change the parameters of the associated camera or spotlight object.

Spotlight views treat the light as if it were a camera. The spotlight falloff is treated the same as the camera field of view.

Keep in mind the following information:

- Using the Camera and Spotlight viewport navigation buttons is the same as moving or rotating the camera or spotlight, or changing their base parameters.
- Changes made with Camera or Spotlight view navigation buttons can be animated the same as other object changes.

Zooming a Camera or Spotlight View

You zoom a camera view by clicking FOV then dragging in the Camera viewport.

The Field of View defines the width of your view as an angle with its apex at your eyepoint and the ends at the sides of the view. The effect of changing FOV is exactly like changing the lens on a camera. As the FOV gets larger you see more of your scene and the perspective becomes distorted, similar to using a wide-angle lens. As the FOV gets smaller you see less of your scene and the perspective flattens, similar to using a telephoto lens. See chapter 18, “Setting Up Cameras.”

Click Light Hotspot for a spotlight view to achieve the same effect as zooming.

The hotspot is the inner of the two circles or rectangles visible in a Spotlight view. Objects inside the hotspot are illuminated with the full intensity of the spotlight. Objects between the hotspot and falloff are illuminated with decreasing intensity as objects approach the falloff boundry. See chapter 17, “Lighting Your Scene.”

Moving a Camera or Spotlight View

You move a camera or spotlight view by clicking one of the following buttons and dragging in the camera or spotlight viewport.

Dolly—Moves the camera or spotlight along its line of sight.

Truck—Moves the camera or spotlight and its target parallel to the view plane.

Pan—Moves the target in a circle around the camera or spotlight. This button is a flyout that shares the same location with Orbit.

Orbit—Moves the camera or spotlight in a circle around the target. The effect is similar to Arc Rotate for non-camera viewports.

Rolling a Camera or Spotlight View

Click Roll, and drag in a camera or a spotlight viewport to rotate the camera or the spotlight about its line of sight. The line of sight is defined as the line drawn from the camera or spotlight to its target. In 3D Studio MAX, the line of sight is also the same as the camera's or the spotlight's local Z axis.

Changing Camera Perspective

Click Perspective, and drag in a camera viewport to change the Field of View (FOV) and dolly the camera simultaneously. The effect is to change the amount of perspective flare while maintaining the composition of the view.

Chapter

3

Selecting Objects

Most actions in 3D Studio MAX are performed on *selected* objects in your scene—you must choose something in a viewport before you can apply a command. As a result, the act of selection is an essential part of the modeling and animation process.

This chapter presents the selection tools available in 3DS MAX. Besides the basic techniques of selecting single and multiple objects using mouse and keyboard, these topics cover the use of named selection sets and other features that help you manage object selection, such as hiding and freezing objects. Also included is an introduction to sub-object selection, essential to working with an object's underlying geometry.

A technique for *grouping* objects is presented. Grouping lets you create more "permanent" selections that have many of the characteristics of independent objects.

Introducing Object Selection

3D Studio MAX is an *object-oriented* program. This means that each object in the 3D scene carries instructions that tell the program what you can do with it. These instructions vary with the type of object.

Because each object can respond to a different set of commands, you apply commands by first selecting the object and then selecting the command. This is known as a *noun-verb* interface because you first select the object (the noun), and then select the command (the verb).

Identifying the Selection Interface

In the 3DS MAX interface, selection commands appear in the following areas:

- Toolbar
- Edit menu
- Tool menu
- Track View
- Display panel

The buttons in the toolbar are the most direct means of selection. The Selection Floater on the Tools menu is handy. The Edit menu provides more general selection commands, plus methods of selecting objects by property. Track View lets you select objects from a hierarchical list.

Toolbar Selection Buttons

The toolbar has four selection-mode buttons. When any of the four selection buttons is active, the program is in a state where you can select objects by clicking them.

Of the four buttons, you use the Select Object button when you want selection only. The remaining three buttons let you both select and transform your selection. Transforms move, rotate, and scale your selection. See chapter 4, "Using Transforms."

Edit Menu Commands

The Edit menu contains selection commands that operate globally on your objects.

Edit Menu commands include:

- Select All
- Select None
- Select Invert
- Select By > Color

  Select By > Name (also a Toolbar button)
- Region > Window (also a Toolbar button)

  Region > Crossing (also a Toolbar button)
- Edit Named Selections (dialog)

Tool Menu Commands

The Tool menu contains two options for modeless selection dialogs or "floaters." These can be placed anywhere on the screen or minimized.

Selection Floater—Same features as Select By Name.

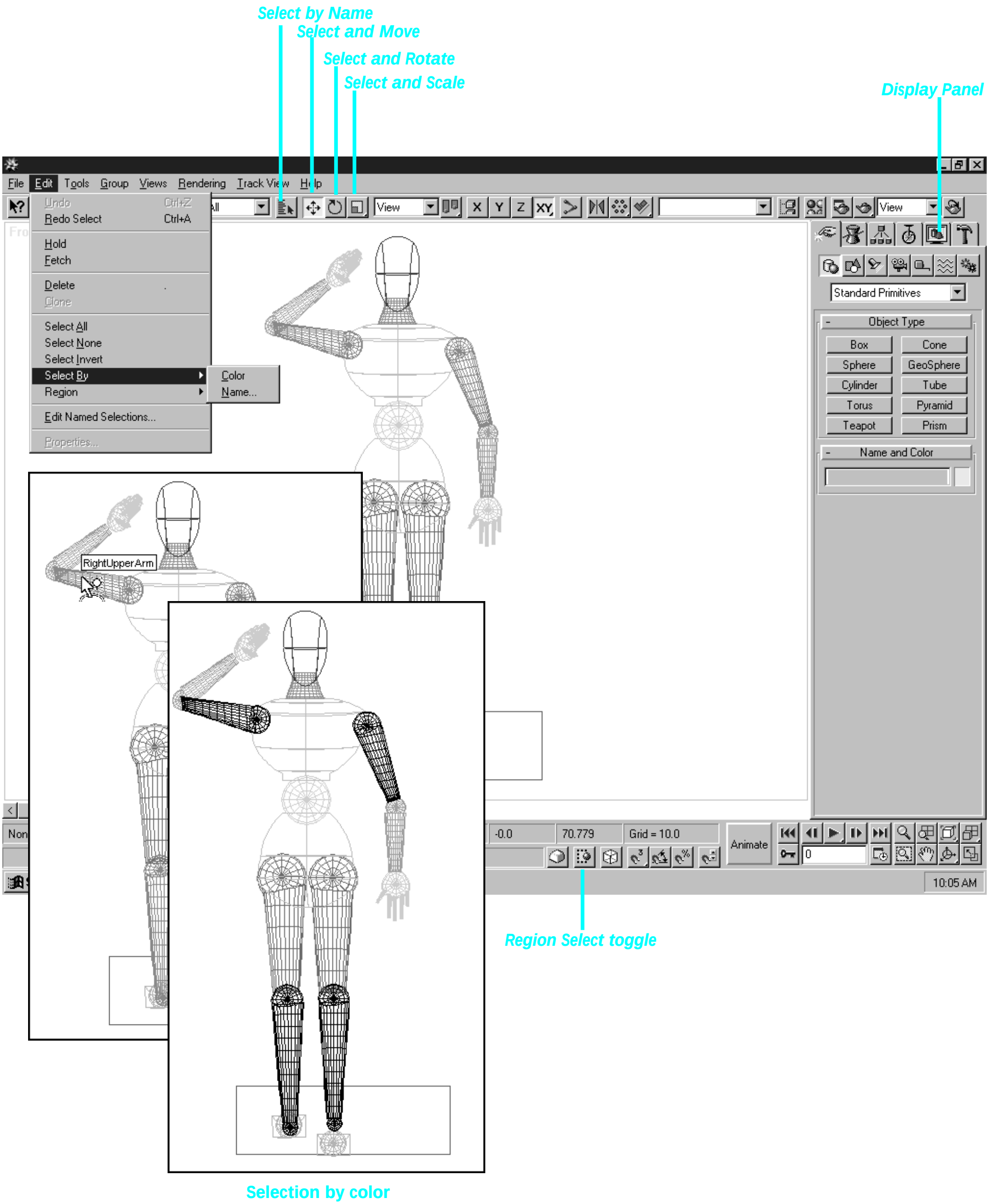
Display Floater—Provides options for hiding and freezing selections as well as some display options.

Track View Selection

Track View is primarily designed as an animation tool but you can also use its Hierarchy List window as an alternative method of selecting objects by name and hierarchy.

Display Panel Selection

The Display panel provides options for *hiding* and *freezing* objects. These techniques exclude objects from other selection methods, and are useful in simplifying complex scenes. Frozen objects are still visible, but hidden objects are not.


Select by Name
Select and Move
Select and Rotate
Select and Scale
Display Panel
File
Edit
Tools
Group
Views
Rendering
Track View
Help
Undo Ctrl+Z
Redo Select Ctrl+A
Hold
Fetch
Delete
Clone
Select All
Select None
Select Invert
Select By
Region
Edit Named Selections...
Properties...
Color
Name...
View
X
Y
Z
XY
Standard Primitives
Object Type
Box
Cone
Sphere
GeoSphere
Cylinder
Tube
Torus
Pyramid
Teapot
Prism
Name and Color
RightUpperArm
-0.0
70.779
Grid = 10.0
Animate
0
10:05 AM
Region Select toggle
Selection by color

Basics of Selecting Objects

The most basic selection techniques in 3D Studio MAX use either the mouse, or the mouse in conjunction with a keystroke.

To select an object:

1. Click one of the four selection buttons in the toolbar—Select Object, Select and Move, Select and Rotate, or Select and Scale.

2. In any viewport, move the cursor over the object you want to select.

 The cursor changes to a small cross when it's over an object that can be selected.

The valid selection zones of an object depend on the type of object and the display mode in the viewport. In shaded mode, any visible surface of an object is valid. In wireframe mode, any edge or segment of an object is valid—including hidden lines.

3. While the cursor displays the selection cross, click to select the object (and to deselect any previously selected object).

A selected wireframe object turns white. A selected shaded object displays white brackets at the four corners of its bounding box.

To select all objects:

- Choose Edit > Select All.

  This selects all objects in your scene.

To invert the current selection:

- Choose Edit > Invert.

  This reverses the current selection pattern. For example, assume you begin with five objects in your scene, and two of them are selected. After choosing Invert, the two are deselected, and the remaining objects are selected.

To extend or reduce a selection:

- Hold down CTRL while you click to make selections.

  This toggles the selection state of the objects you select. Use this method to select or deselect objects. For example, if you have two objects selected and CTRL+click a third, the third object is added to the selection. If you now CTRL+click any of the three selected objects, that object is deselected.

To lock a selection:

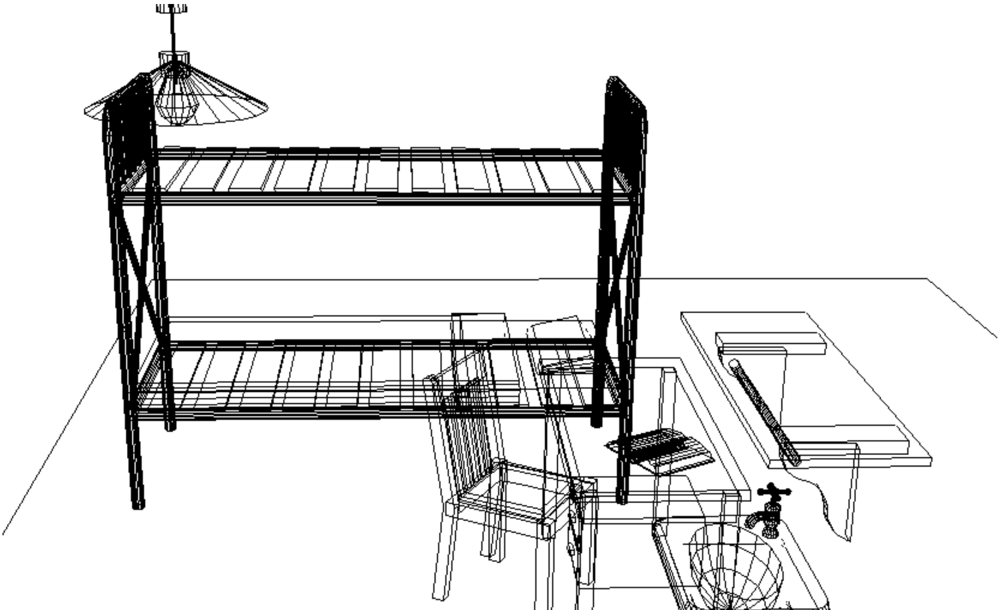
1. Make your object selection.

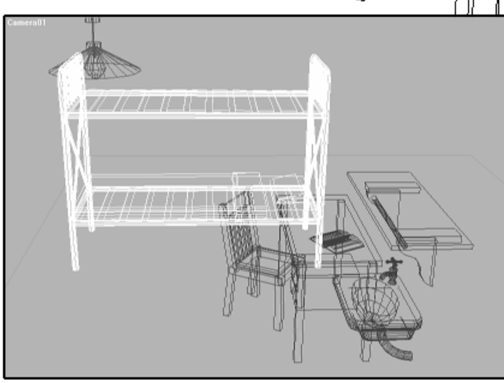
2.  Click the Lock button in the status line to turn on locked selection mode.

While your selection is locked, you can drag the mouse anywhere on screen without losing the selection. The cursor displays the current selection icon. When you want to deselect or alter your selection, click the **Lock** button again to turn off locked selection mode. SPACEBAR is the keyboard toggle for locked selection mode.

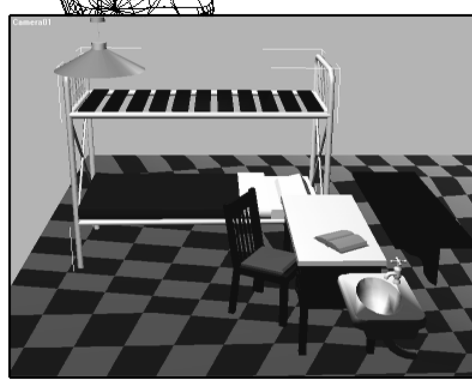
To deselect an object, do any of the following:

- Click anywhere outside the current selection.
- Hold down the ALT key and either click an object to deselect it, or drag a region around the object to deselect it.
- Hold down the CTRL key and click to either deselect selected objects, or select deselected objects.
- Choose Edit > Select None to deselect all objects in the scene.





Bed selected in wireframe view



Bed selected in smooth and shaded view

Selecting by Region

Using any of the four selection modes, you can select one or more objects by dragging the mouse to define a *region* around them.

To make a region selection using defaults:

1.  Click Select Object.

2. Drag the mouse to define a region.

A dash-lined rectangle appears.

3. Release the mouse button to select all objects within or touching the region.

The selected objects turn white.

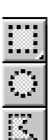
You can perform the same type of region selection while using the Select and Transform buttons in the toolbar, *as long as you begin your drag while over an unselectable area of the viewport.* Otherwise, you'll transform the object beneath your mouse when you begin to drag.

Region selection

As a default, when you drag the mouse a rectangular region is created. When you release the mouse all objects *within* the region and *touched* by the region are selected—again by default. You can change each of these settings, as described in the following topics.

Setting Region Type

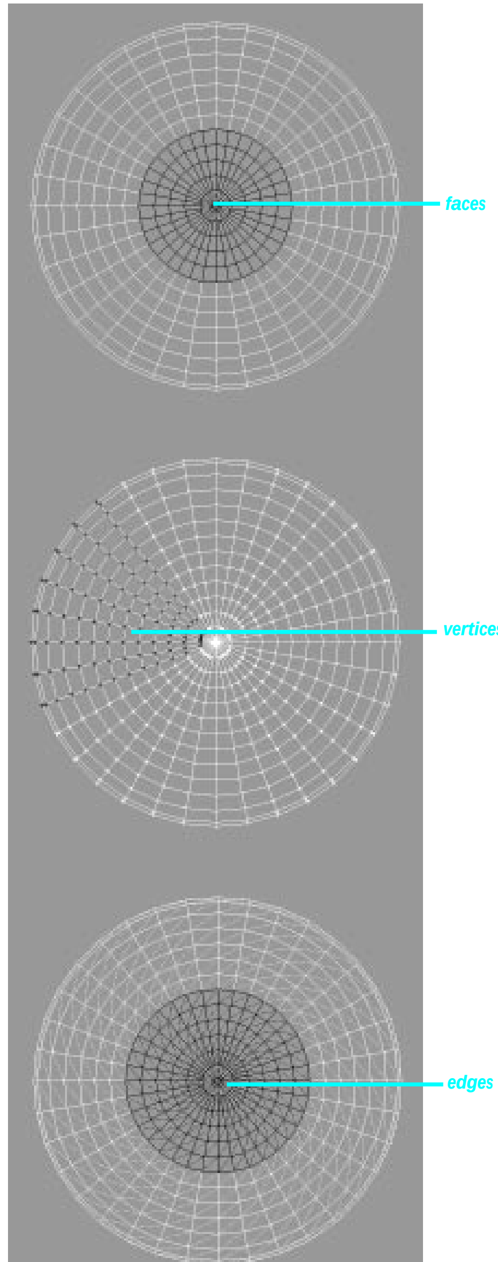
The type of region you define when you drag the mouse is set by the Region flyout button to the right of the Select Object button. You can choose three types of region:



Rectangular Region—Dragging the mouse creates a rectangular region.

Circular Region—Dragging the mouse creates a circular region.

Fence Region—Dragging the mouse and clicking repeatedly creates an irregular region.



Sub-object selections by region

To select objects with Fence region:

1. Drag to draw the first segment of a polygon and release the mouse button.

   A “rubber-band line” is now attached to the cursor, anchored at the point of release.

2. Move the mouse and click to define the next segment of the “fence.” You can make as many segments as you want.

3. To complete the fence region, click on the first point. A crosshair appears when you’re near enough to click on the first point.

   Objects within or touching the fence region are selected, depending on region inclusion, discussed next.

Until the selection is complete, you can right-click to cancel fence selection.

Setting Region Inclusion

Two items in the Edit menu let you set the method for selecting objects by region.

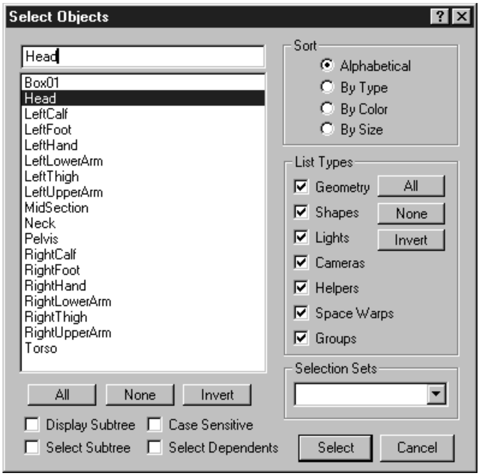
Choose Edit > Region to display a submenu of the following two items. Only one can be active at a time.

Window—Selects only those objects that are *within* the region.

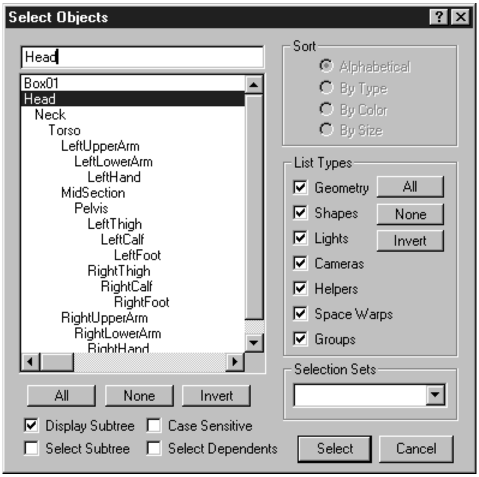
Crossing—Selects all objects that are *within* the region *and crossing* the boundaries of the region. This is the default region.

 The Window/Crossing buttons on the prompt line also toggle between these two modes.

Using Select By Name



Options at lower left cleared



Subtree option selected

You can select objects by their assigned names, and avoid mouse clicks completely.

To select objects by name:

1. Do one of the following:

-  Click Select By Name in the toolbar.
- Choose Edit > Select By > Name on the menu bar.
- Choose Tools > Selection Floater.

The Select Object or Selection Floater dialog appears. By default, these dialogs list all objects in the scene. Currently selected objects are highlighted in the list.

2. Choose one or more objects in the list. Use CTRL or SHIFT to add to the selection.

3. Click Select to make the selection.

Select Object closes, while Selection Floater remains active.

Typing an Object Name

The field at the top of the Select Object dialog lets you highlight an object in the list by entering its name. You can also use wild cards to select multiple objects.

To enter object names for selection:

1. In the field above the list, type the name of the object you want to select. Based on selected options, matching names in the list are highlighted as you type.

2. Press ENTER again to make the selection.

You can use wild cards to specify one or more object names. The wild cards are the standard question mark (?) and asterisk (*). The question mark is used as a placeholder for any single character. The asterisk is used as a placeholder for all remaining characters.

Changing the List

Using various controls in the Select Objects dialog, you can change the way objects are displayed in the list. These include the following:

All/None/Invert—Let you select all items in the list, deselect all items in the list, or invert the pattern of the current selection.

Display Subtree—Displays the hierarchy of linked objects. When selected, objects in hierarchies appear in an indented list, as shown in this example:

Thigh
  Shin
    Foot

Select Subtree—Selects objects linked as children to the selected parent object. Display Subtree is not required for this selection to work.

Select Dependents—Selects dependent objects, such as instances and references, and objects sharing a common modifier.

Case Sensitive—Determines the match when you enter object names in the field above the list. When selected, exact matches are required for names. This setting also affects the sorted order of the list itself. Default=cleared.

- When cleared, entering **Tr*** as a wild card specification would highlight Tree, tree, TRACK, and TracK. Entering the same specification when selected would highlight Tree and TracK.
- When selected, uppercase letters appear before lowercase letters in an alphabetically sorted list.

List Types—Filters the objects displayed in the list by category. When you clear a category, that category of object is hidden from the list. Default=all selected.

The List Type area has its own set of All/None/Invert buttons to display all categories, no categories, or the reverse of the current pattern of selected categories.

Sort—Changes the order of listed objects. You can sort alphabetically by name, by category of object, by assigned wireframe color, or by size.

Size is measured by the number of faces in the object. Objects with the least number of faces are listed first.

Alphabetical sort is affected by the Case Sensitive setting. If Display Subtree is selected, all Sort options are grayed out.

Listing Selection Sets and Groups

Names of groups appear in the main list, while named selection sets have a separate dropdown list.

Groups—Names of groups appear in the list surrounded by square brackets, for example [Group01]. If the group is opened, objects in the group appear in the list, indented to show their hierarchy.

When Case Sensitive is off, all groups appear at the top of the list.

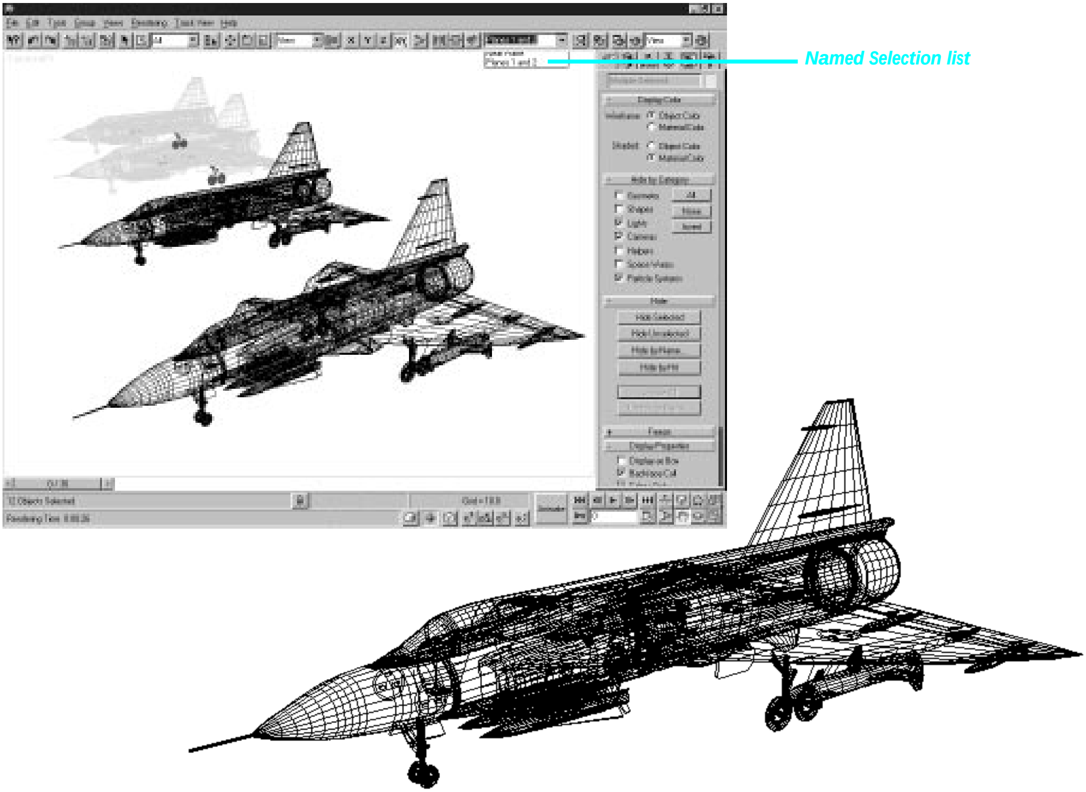
Selection Sets—Lets you choose any named selection set in your scene. When you choose a set, its member objects are highlighted in the main list. Changes to the list do not affect the contents of the named selection set.

Other Uses of Select Object Dialog

As you use 3DS MAX, you'll see the Select Objects dialog appear in other contexts with a different name on the title bar. It is used as a program-wide standard for managing objects by name.

For example, when you unhide objects by name, the dialog is called Unhide Objects. Add To Named Selection also uses this dialog. See next topic. In every case, the dialog is the same in layout and function.

Using Named Selection Sets



You can assign a name to the current selection of objects, and then later reselect those objects by picking their selection name from a list. You can also edit the contents of named sets from a dedicated dialog.

To assign a name to a selection set:

1. Select one or more objects using any combination of selection methods. Click in the Named Selection field in the toolbar.

2. Enter a name of up to 15 characters (including spaces). Names are case sensitive. Press ENTER to complete the selection set.

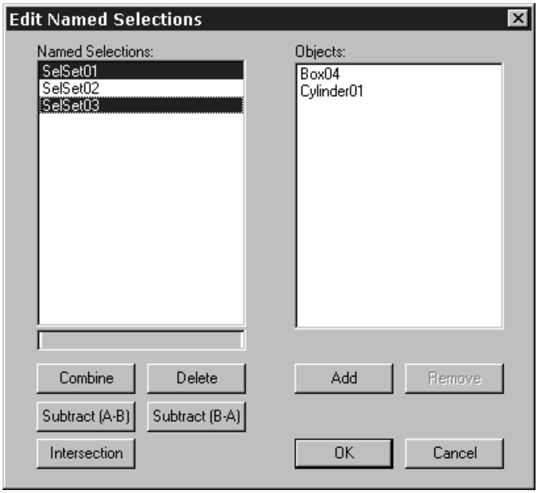
You can now select another combination of objects and repeat the process to create another named selection set.

To retrieve a named selection set:

1. In the Named Selection field, click the arrow.

2. On the list, click a name.

The objects defined by that name are selected. The name of the selection set is displayed in the Named Selection field.



Editing Named Selections

As you model and create a scene, you're likely to rearrange the objects making up your named selection sets and need to edit the contents of those sets.

To edit named selection sets:

- Choose Edit > Edit Named Selections to display the Edit Named Selections dialog.

The Named Selections list on the left contains all named selection sets in your scene. Click a name to highlight it. Use CTRL or SHIFT to add to the selection.

The field below the Named Selections list lets you change the name of a single highlighted set. The objects in the set appear in the Objects list on the right. If you highlight two or more sets, the Objects list shows only what objects appear in *all* the highlighted sets. If no object is common to these sets, the list is blank.

The following options apply to highlighted sets in the Named Selections list. Except Delete, these options require two or more sets. For some options, a dialog prompts you to name the new selection set, which is then added to the list.

Delete—Removes any selected set from the list, discarding the selection.

Combine—Creates a single, new selection set containing all objects in the highlighted sets. The original sets remain, but can be deleted.

Intersection—Creates a single, new selection set containing all objects in common between the highlighted sets. The Objects list shows what objects will be part of the intersected set.

Subtract (A–B), Subtract (B–A)—Subtracts one set from another. When you highlight two sets to be operands A and B, the Objects list shows what objects will be subtracted. The highlighted set closest to the top of the list is defined as operand A. Clicking Subtract (A–B) subtracts items in the bottom set from the top set. Neither set is renamed in this process.

The following options apply to highlighted sets in the Objects list.

Add—Displays an Add To Named Selection dialog, a version of Select By Name.

Remove—Highlight one or more objects to activate this button, then click to remove from set.

Editing Sub-Object Named Selections

When you use the Edit Named Selections dialog with named sub-object selections, only the Named Selection options are available. See chapter 6, "Basics of Creating and Modifying Objects," for more information on naming sub-object selections.

Using Selection Filters

The Selection Filter list is in the 3D Studio MAX toolbar. You use it to deactivate selection for all but a specific category of object. As a default, all categories can be selected, but you can set the Selection Filter so that only lights, for example, can be selected. You can also create combinations of filters to add to the list.

To use the Selection Filter:

- Click the Selection Filter arrow and click a category from the Selection Filter list.

  Selection is now limited to objects defined in this category. The category remains in effect until you change it.

The following categories are available:

All—All categories can be selected. This is the default setting.

Geometry—Only geometric objects can be selected. This includes meshes, patches, and other kinds of objects not specifically included in this list.

Shapes—Only shapes can be selected.

Lights—Only lights (and their targets) can be selected.

Cameras—Only cameras (and their targets) can be selected.

Helpers—Only helper objects can be selected.

Warps—Only space warps can be selected.

Using Combo

The Combo feature allows you to combine two or more categories into a single filter category.

To create a Combo category:

1. Choose Combo on the dropdown list to display the Filter Combinations dialog. All single categories are listed. Select the categories you want to combine.

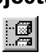
2. Click Add. The combination appears in a list to the right, abbreviated by the first letter of each category. Click OK.

For example, if you selected Geometry, Lights, and Cameras, the Combo would be named GLC. This name appears below Combo on the dropdown list.

Selecting with Track View

Track View provides sophisticated methods to edit your animation tracks. In addition, its Hierarchy List window displays all objects in the scene by name and hierarchy. Using Track View, you can select any object in the scene by clicking its object icon in the Hierarchy List window.

To open Track View and display and select objects:

1.  Click Track View in the toolbar.
2. Right-click the Object text label to display a pop-up menu.
3. Click Expand Objects. A list of all objects in the scene appears. Each object is signified by a yellow-cube icon.
4.  Click any yellow cube in the list to select the named object.

You can make the following kinds of selections:

- Select several adjacent objects in the list. Click the first object, hold down SHIFT**,** and click another object elsewhere in the list.
- Add to the selection by pressing CTRL while clicking. CTRL lets you toggle individual items on and off without deselecting others in the list.
- Select all items at one level of the hierarchy by holding down ALT and clicking one object at that level.
- Select all descendants of an object. Select the object, right-click, and choose Select Children from the menu.

Tip: You can open a Track View window for the sole purpose of selecting objects by name. Shrink the window until only a portion of the Hierarchy appears, and then move the window to a convenient area on your screen.

Freezing and Unfreezing Objects

You can *freeze* any selection of objects in your scene. Frozen objects, whether wireframe or rendered, turn a dark gray. They remain on screen, but can't be selected, and therefore can't be directly transformed or modified. Freezing lets you protect objects from accidental editing and speeds up redraws.

Frozen objects are similar to hidden objects. Linked, instanced, and referenced objects behave when frozen just as they would if unfrozen. Frozen lights and cameras and any associated viewports continue to work as they normally do.

Freezing Objects

You can freeze one or more selected objects. This is the usual method to put objects "on hold."

You can also freeze all objects that are *not* selected. This method lets you keep only the selected object active, useful in a cluttered scene, for example, where you want to be sure no other objects are affected.

To access Freeze options, do one of the following:

-  Open the Display panel, then expand the Freeze rollout.
- Choose Tools > Display Floater. This modeless dialog has the same options as the Freeze rollout. It also contains Hide options.

Freeze Options

These are the various ways you can freeze objects. Frozen objects are displayed as a uniform dark gray, regardless of color or material assignment.

Freeze Selected—Select one or more objects in your scene, then click. This is how you would freeze a named selection set, for example.

Freeze Unselected—Select one or more objects you want to work with, then click. All *unselected* objects are frozen.

Freeze By Hit—Click to turn on this selection mode. Use the Select cursor to freeze each object you click ("hit") in the viewport.

Freeze By Name—Displays a Freeze Objects dialog. This is the same as the Select By Name dialog discussed in an earlier topic.

Sorting Options

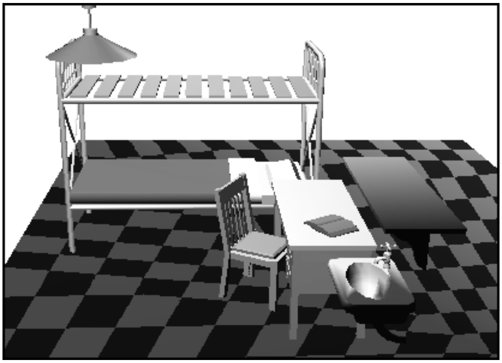
If you have a large number of objects in your scene, you can sort objects using options in the Sort area of the Freeze Objects dialog.

Alphabetical—Sorts list in alphabetical order, with numbers at the top. This is the default.

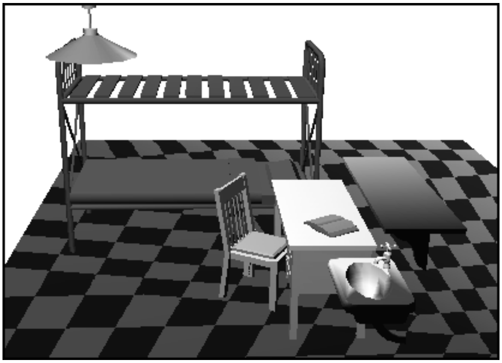
By Type—Sorts list by category of object: geometry, shapes, lights, and so on.

By Color—Sorts list by color. All objects with the same color are grouped together in the list.

By Size—Sorts list by the number of faces in each object, with the smallest number at the top.



Nothing frozen



Bed frozen

Unfreezing Objects

Use the Unfreeze All option to return all frozen objects to their normal state.

Use these options to unfreeze selectively:

- Unfreeze By Hit
- Unfreeze By Name

These options parallel those for freezing. When you click Unfreeze By Name, the same dialog appears, now called Unfreeze Objects. The Unfreeze buttons turn gray when all objects have been unfrozen.

Hiding and Unhiding Frozen Objects

While frozen objects can't be selected, they can be hidden, using Hide Unselected. See the next topic.

You can unfreeze a hidden and frozen object with Unfreeze All or Unfreeze By Name, but the object remains hidden. When you unhide the object, it reappears in its normal, unfrozen state.

Hiding and Unhiding Objects by Selection

You can *hide* any selection of individual objects in your scene. They disappear from view, making it easier to select remaining objects. Hide also speeds up redraws. You can then unhide them all at once or by individual object name. You can also filter the names by category, so only hidden objects of a certain type are listed.

Hidden objects are similar to frozen objects. Linked, instanced, and referenced objects behave when hidden just as they would if unhidden. Hidden lights and cameras and any associated viewports continue to work as they normally do.

Hiding Objects

Hiding objects is similar to freezing them. See previous topic. You can hide one or more selected objects. You can also hide all objects that are *not* selected.

Another option is to hide objects by category. See next topic.

To access Hide options, do one of the following:

-  Open the Display panel. Click Hide, if necessary, to expand the rollout:
- Choose Tools > Display Floater. This modeless dialog has the same options as the Hide rollout. It also contains Freeze options.

Hide rollout on Display Panel

Display Floater

Hide Options

These are the various ways you can freeze objects using selection. Hidden objects disappear entirely from the scene.

Hide Selected—Select one or more objects in your scene, then click. This is how you would hide a named selection set, for example.

Hide Unselected—Select one or more objects you want to work with, then click. All *unselected* objects are hidden.

Hide By Hit—Click to turn on this selection mode. Use the Select cursor to hide each object you click ("hit") in the viewport.

Hide By Name—Displays a Hide Objects dialog. This is the same as the Select By Name dialog discussed in an earlier topic.

If you have a large number of objects in your scene, you can sort objects using options in the Sort area of the Hide Objects dialog. See "Sorting Options" in the previous topic, "Freezing and Unfreezing Objects."

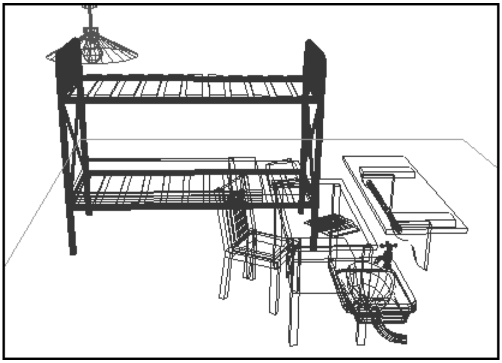
Unhiding Objects

You can unhide objects in either of two ways:

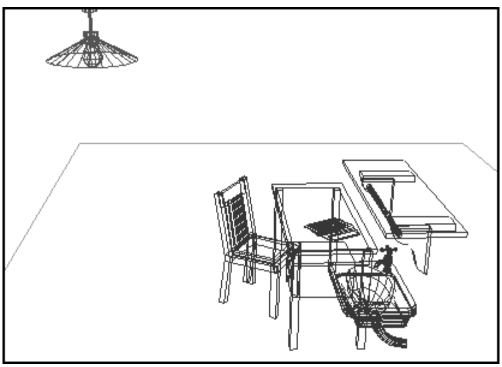
- Use Unhide All to unhide all objects at the same time.
- Use Unhide By Name to unhide object selectively. When you click Unhide By Name, the same dialog appears as for hiding, now called Unhide Objects.

The Unhide buttons turn gray when no object in the scene is hidden.

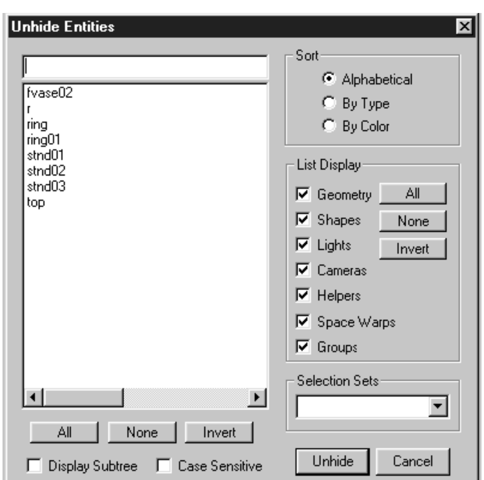
Note: Objects that were first hidden by selection and then hidden by category do not reappear. Although they are unhidden at the selection level, they are still hidden at the category level. See the next topic, “Hiding and Unhiding Objects by Category,” for more details.



original scene



scene with bed hidden



Hiding and Unhiding Objects by Category

You can hide objects by *category*—the basic types of object. For example, you can hide all lights in your scene at one time, or all shapes, or any combination of categories. By hiding all categories, your scene appears empty. Hidden objects, while not *displayed*, continue to exist as part of the geometry of your scene but cannot be selected.

To hide a category of objects:

1.  Open the Display panel.

2. Click Hide by Category, if necessary, to expand the rollout. By default, all categories are cleared (unhidden) on this rollout.

3. Select the category you want to hide. *All* objects of that category disappear from your scene as soon as you make the selection.

The same Hide By Category options appear on the Object Level panel of the Display Floater (Tools > Display Floater).

To unhide a category of objects:

- Clear the category.

All objects in the category reappear, unless some have been hidden by selection. See "Effects of Hiding by Category."

Hiding Geometry and Particle Systems

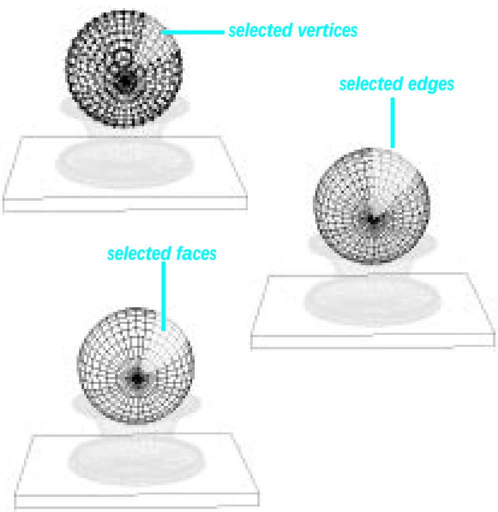
Geometry and particle systems have separate categories, even though particle systems are also geometry.

- Selecting Geometry hides all geometry in the scene, including particle systems. The option for particle systems turns gray.
- Selecting Particle Systems only hides these objects, leaving the other geometry unaffected.

Effects of Hiding by Category

- If you create an object in a category that is hidden, the category selection is cleared and the objects in that category are unhidden.
- Unhiding by category has no effect on objects hidden with the controls on the Hide rollout (see previous topic). They remain hidden. You need to use the controls on that rollout to unhide them.
- Unhiding by selection does not return a hidden object to the scene if the *category* of the object is hidden. The Unhide All and Unhide By Name controls continue to work, but the effect is not seen until the category is cleared.
- Lights hidden by category continue to shine. Views through cameras and targeted spotlights are still active.
- Linked, instanced, and referenced objects behave when hidden just as they would if visible.

Sub-Object Selection



Sub-object selections

When modeling an object, you often edit a portion of its underlying geometry—such as a set of its faces or vertices. You can use the selection methods described in this chapter to make these *sub-object* selections.

You access an object's geometry through the Modify command panel. The most common technique is to convert an object into an "editable mesh." Once converted, the various levels of its geometry are available for selection.

This is a general introduction to sub-object selection. See chapter 8, "Editing Meshes," and chapter 9, "Creating and Editing Shapes."

To make a sub-object selection:

1. Select the object you want to edit.

2. Open the Modify panel.

3. On the Modifier Stack rollout, click Edit Stack. On the Convert To menu, click Editable Mesh.

4. On the Modifier Stack rollout, click the Sub-Object button.

When active, this button turns yellow, providing access to the Selection Level list.

5. From the Selection Level list, choose a level of selection—vertex, face, or edge.

If you are working in a shaded viewport, switch to Wireframe by right-clicking the viewport label. For a detailed selection, you might want to zoom in on the object.

6. Click one of the four toolbar selection buttons, and then use the same selection methods you'd use on objects to select the sub-object components.

Restoring Object Selection

When you edit an object at sub-object level, the program is in a mode that only allows you to select components of that object. While the Sub-Object button is on, you can't deselect the current object, nor can you select other objects.

To exit sub-object selection mode, do one of the following:

- Click the Sub-Object button to turn it off.
- Open another panel. This turns off the Sub-Object button.

If you've turned off the Sub-Object button and object selection is still not restored, it might be due to the following reasons:

- Your selection is locked. Click the Lock button in the prompt line to turn it off.
- You've set the Selection Filter in the toolbar to a specific category of object, so you can't select any of the other categories. To fix this, select All in the Selection Filter list.

Using Groups

Grouping lets you combine two or more objects into a single *grouped* object. The grouped object is given a name, and then treated much like any other object.

To define a group:

1. Select one or more objects.

2. Choose Group > Group.

3. Enter a name for the group and click OK.

Group names are similar to object names, except that they're carried by the group object. In lists like the one for the Select Object dialog, group names appear in square brackets, for example [Group01].

General Features of Groups

When a group is formed, all of its member objects are rigidly linked to an invisible dummy object. The pivot point and the local transform coordinate system of the dummy object become those of the group object.

Once you group objects, you can treat them as a single object in your scene. You can click any object in the group to select the group object.

Groups can be nested. That is, groups can contain other groups, up to any level.

Transforming and Modifying a Group

You can transform or modify a group as if it were a single object, and you can animate the transforms and the modifiers.

Modifiers are applied to the objects within the group as instances and remain with those objects, even if you remove them from the group.

Transforms, on the other hand, are applied only to the group—specifically, the dummy object that represents the group.

Individual objects within a group can be transformed and animated independently from the group. However, when the group itself is transformed, all grouped objects are equally affected. The group transform is uniformly added to their independent motions. An analogy would be a cage of birds, each flying around on its own, while the cage itself is being moved. In the case of groups, the "cage" (the dummy object) expands to surround all objects in the group, wherever the objects' independent transforms take them.

Accessing Objects in a Group

You can *open* and *close* groups to access the individual objects contained in them without dissolving the group. These commands maintain the integrity of the group.

Open—Temporarily opens the group so that you can access its member objects. While a group is open, you can treat the objects (or nested groups) as individuals. You can transform them, apply modifiers, and access their modifier stacks.

Close—Restores the group when you're finished working with the individual objects.

Dissolving Groups

You can permanently dissolve groups by either *ungrouping* or *exploding* them. Both commands dissolve groups, but to different levels.

Ungroup—Goes one level deep in the group hierarchy. It separates the current group into its component objects (or groups), and deletes the group dummy object.

Explode—Similar to Ungroup, but dissolves all nested groups as well, leaving independent objects.

When you Ungroup or Explode a group, the objects within the group lose all *group* transforms that were applied on nonzero frames. However, objects retain any individual animation.

To transform or modify the objects *within* a group, you must first remove them from the group—either temporarily or permanently. The Open command lets you do this.

Comparing Groups with Other Selection Methods

In comparing the various methods you can use to combine two or more objects in 3D Studio MAX, grouping is more permanent than selection sets, but less permanent than attaching objects.

Selection sets—Form a temporary collection of objects to which you apply the current commands. As soon as you select another object, the selection set is gone.

Named selection sets—Let you reselect the same pattern of objects, but the positional relationship between those objects (their transforms) might be different each time you recall the named set.

Grouped objects—Maintain their positional relationships—unless you open the group and rearrange them. A group also keeps its identity as an individual object.

Each object in a group retains its modifier stack, including its base parameters. At any time, you can open the group to edit an object, and then close the group to restore the group identity.

Attached objects—Form a single object. The modifier stacks of the original objects are lost—including their base parameters. You can regain the form of the original objects by detaching them, but they become plain meshes. See online reference for details of attaching objects.

Chapter

4

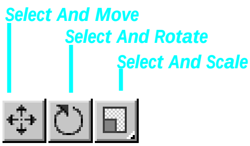
Using Transforms

A *transform* is an adjustment of an object's position, orientation, or scale, relative to the 3D world—or *world space*—in which you're working. You can apply three types of transform to an object:

- Position
- Rotation
- Scale

This chapter shows you how to transform objects, and how to animate your transforms. It also describes the tools you use to modify and control the transforms.

Applying Transforms



To use a transform, you click one of the three transform buttons in the toolbar or choose a transform from a shortcut menu. You then apply the transform to a selected object using the mouse, a type-in dialog, or both.

To transform an object using the toolbar:

1. In the toolbar, click one of the three transform buttons, labeled Select And Move, Select And Rotate, or Select And Scale. These buttons are usually referred to as Move, Rotate, and Scale.

2. Position the mouse over the object you want to transform.

- If the object is already selected, the cursor changes to indicate the transform.

- If the object is not selected, the cursor changes to a small plus sign to show that the object can be selected.

3. Drag the mouse to apply the transform.

   If you drag the mouse over an unselected object, it becomes selected and is also transformed.

To cancel a transform:

- Right-click while you're dragging the mouse.

To transform an object from the shortcut menu:

1. Right-click a selected object. A shortcut menu lists the three transforms.

2. Choose one of the transforms. The equivalent transform button is selected in the toolbar.

3. Drag the object to apply the transform.

Using Transform Type-In

Transform Type-In is a small dialog you can keep on your screen while you work. Its title bar and contents update to match the currently active transform and selected object.

To use the Transform Type-In dialog:

1. Choose Tools > Transform Type-In to display the dialog.

2. Apply a transform to a selected object.

3. You can do any of the following, switching from one to the other as required.

- Type a value in an axis field and press ENTER to apply the transform change to the object in the viewport.

- Drag a spinner in an axis field to update the object in the viewport.

- Drag the object to apply the transform and read the resulting change in the dialog.

For example, if Move is active, the dialog fields read out both the absolute and offset positions of the selected object in world space. If no object is selected, the fields turn gray.

Using Type-In with Sub-Object Selection

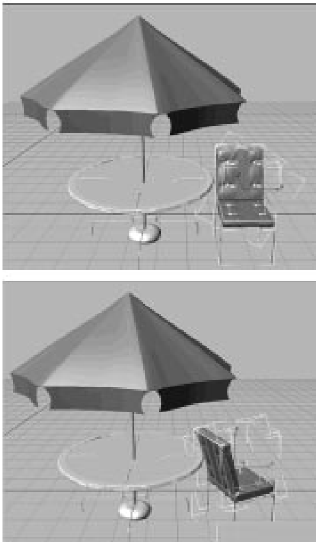
You can use the Transform Type-In dialog with any sub-object selection or gizmo. The transform affects the axis tripod for the selection.

Absolute and offset world coordinates are those of the tripod. If multiple vertices are selected, the tripod is at the center of the selection and its location is given in world coordinates.

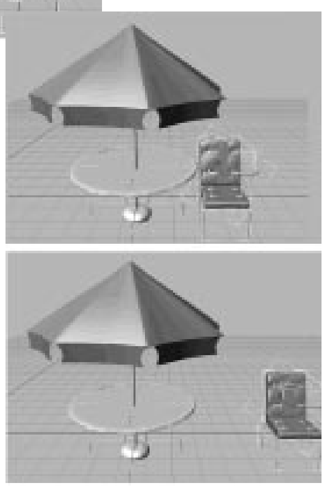
Because axis tripods cannot be scaled, Absolute Scale fields turn gray when in sub-object mode.

See chapter 6, "Basics of Creating and Modifying Objects," for information on sub-object selection and gizmos.

Moving and Rotating Objects



Rotating an object



Moving an object

Moving *translates*, or repositions, an object in the 3D world. Rotation changes the *orientation* of an object.

To move an object:

- Click Select and Move in the toolbar. The button highlights green. You can now select objects, move previously selected objects, or both select and move objects in a single mouse action.

The direction of the movement is determined both by your mouse and by the current Transform Coordinate System. In addition, the Axis Constraint buttons limit movement along one or two axes. See later topics in this chapter for details.

To rotate an object:

- Click Select and Rotate in the toolbar. The button highlights green. You can now select objects, rotate previously selected objects, or both select and rotate objects in a single mouse action.

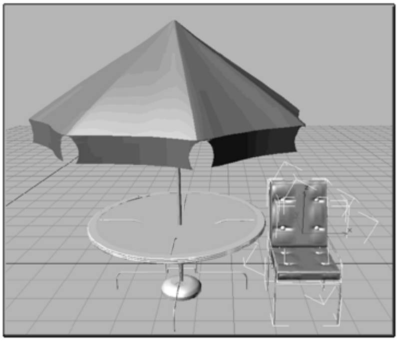
The axis about which the rotation takes place is determined by the Axis Constraint buttons. The center of rotation is determined by the Transform Center button. See later topics in this chapter for details.

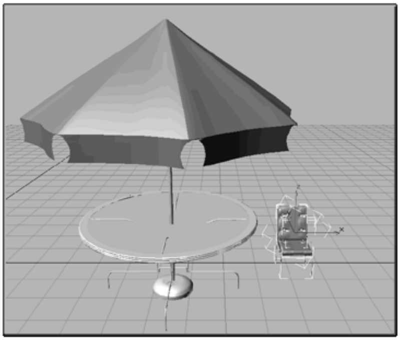
Failure to Move or Rotate

In some cases, an object might fail to move or rotate, even when the proper button is on and the object is selected. This could be due to one of the following reasons:

- The object is frozen. See chapter 3, “Selecting Objects.”
- Axis constraints don’t agree with the viewport. See topics later in this chapter.
- A transform controller has been assigned to the object. See chapter 27, “Working with Controllers.”
- Inverse Kinematics mode is on and the preference called Always Transform Children of the World is off. See chapter 24, “Using Inverse Kinematics.”

Scaling Objects





Uniformly scaling an object

Scaling changes the size of an object.

The Scale transform button is a flyout that provides three types of scale. The type of scale visible in the toolbar is the type that appears in the object shortcut menu, accessed by right-clicking a selected object.

To scale an object:

- Set the Select and Scale button to the scale type you want to use, then click it. The button highlights green.

  You can now select objects, scale previously selected objects, or both select and scale objects in a single mouse action.

Uniform Scale

Use Uniform Scale to scale the selection equally along all three axes.

The center of the scale is determined by the Transform Center button. The axis constraint buttons and the Transform Coordinate system have no effect on uniform scaling. See topics later in this chapter.

Non-Uniform Scale

Use Non-Uniform Scale to scale the selection differently along the three axes.

The Axis Constraint buttons determine the axis or axes along which the scaling occurs. The Transform Coordinate system determines the direction of the scaling, and the Transform Center button determines the center from and to which the scaling takes place.

Squash

Use Squash to scale the selection in one direction along one axis and in the opposite direction along the other axes. Squash gives the appearance of maintaining the volume of the selection.

The Axis Constraint buttons specify the axis of scale, while the remaining axes scale in the opposite direction. If you use a double-axis constraint, the single remaining axis scales in the opposite direction.

Notes on Scaling

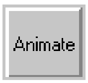
If you scale an object and later check its base parameters in the Modify panel, you see the dimensions of the object *before* it was scaled. The base object exists independently of the scaled object that is visible in your scene.

You can use a tape measure helper object to measure the current dimensions of an object that has been scaled or changed by a modifier. See chapter 5, “Precision and Drawing Aids.”

Animating Transforms

You can animate your transforms by turning on the Animate button and then performing the transform at any frame other than frame 0. This creates a *key* for that transform at the current frame.

To animate an object between three points:

1. 

Click the Animate button to turn it on.

   The Animate button and the highlight border around the active viewport both turn red.

2. Drag the time slider to frame 25, for example.

3. Move the object from point A to point B.

   A Move key is created at frames 0 and 25. The establishing key at frame 0 describes the object's position at point A. The key at frame 25 describes the object's position at frame 25.

4. Drag the time slider to frame 50.

5. Move the object from point B to point C.

   A Move key is created at frame 50 that describes the object's position at point C.

6. 
Click the playback button.

   The object moves from point A to point B over frames 0 to 25, then proceeds to point C over frames 26 to 50.

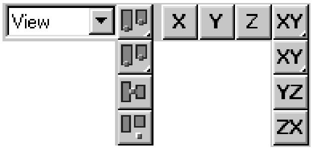
   The playback button has turned into a stop button—click this to stop playback.

You can combine different transforms in a single animation sequence, so that an object appears to move as it rotates and changes in size.

See chapter 22, "Animation Concepts and Methods," for more information on animation techniques.

Transform Managers

To the right of the Move, Rotate, and Scale transform buttons in the toolbar are six controls, collectively referred to as the *transform managers*. These controls modify the action of the three transform buttons.



Transform managers in the toolbar

You use the transform manager controls to specify:

- The *transform coordinate system*.
- The *transform center* about which you apply the transform.
- The *axis constraints* that restrict the transform along one or two axes.

Definitions

Certain terms are used in the description of transforms and the transform managers.

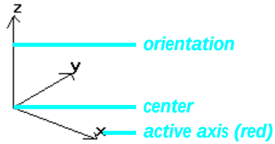
- An *axis* is a straight line along which an object is moved or scaled, or about which an object is rotated. When you work in 3D, you use three axes, labeled X, Y, and Z, that are oriented 90 degrees from each other.
- A *transform coordinate system* specifies the orientation of the three X, Y, Z axes used by a transform.

  For example, in the World coordinate system, as seen from the Front view, the X axis runs horizontally from left to right, the Y axis runs from back to front, and the Z axis runs vertically, from bottom to top.

  On the other hand, each object carries its own Local coordinate system. If the object has been rotated, its Local coordinate system will be completely different from the world coordinate system.
- The *transform center,* or *pivot point,* is the spot about which a rotation takes place, or to and from which a scale occurs.

Using the transform managers in the toolbar, you can specify any combination of axes, transform coordinate systems, and transform centers.

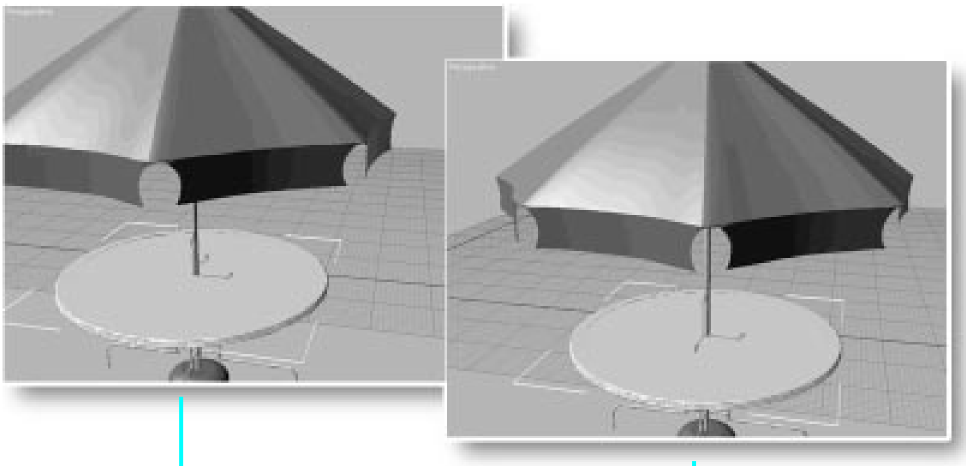
Axis Tripod Icon



An axis tripod appears in the viewports when you select one or more objects, to assist you visually in your transforms. This tripod consists of three lines, labeled X, Y, and Z, and shows you three things:

- The orientation of the tripod reveals the orientation of your coordinate system.
- The location of the junction of the three axis lines shows you where your transform center is.
- The highlighted red axis lines show you the axis or axes to which the transform is constrained. For example, if only the X axis line is red, you can move objects only along the X axis.

You can toggle the display of the axis tripod in all viewports by choosing Views > Show Axis Icon.



world axis local axis

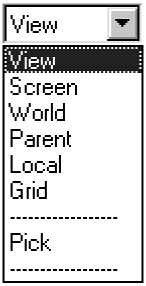
Transform Manager Settings

The state of the three transform managers—coordinate system, center, and axis constraints—is stored with each type of transform. When you switch from Move to Rotate to Scale, the transform managers change to whatever combination they were in when you last used that transform.

For example, if you select Rotate and set the transform managers to Screen, Selection Center, and Y constraint, when you choose Move, the controls might shift to View, Pivot Point, and XY constraint—whichever combination was set the last time you used the Move button. When you go back to Rotate, the controls revert to Screen, Selection Center, and Y constraint.

Tip: To avoid surprises, always click the transform button *first*, and then set the transform managers. If, instead, you first set the transform managers, their settings are likely to change as soon as you choose a new transform button. One way to remember this is always to set the transform and managers by working from left to right in the toolbar.

Specifying a Transform Coordinate System



The transform coordinate system determines the orientation of the X, Y, and Z axes used by the transform. The type of transform system you use affects position, rotation, and scale transforms.

You specify your transform coordinate system using the Transform Coordinate System list.

The following coordinate systems are available:

View—A hybrid of World and Screen coordinate systems. Using View, all orthographic views use the Screen coordinate system, while User, Perspective, and Camera views use the World coordinate system.

Screen—Uses the active viewport screen as the coordinate system. X is horizontal, running in a positive direction toward the right; Y is vertical, running in a positive direction upward; Z is depth, running in a positive direction toward you.

Note: Since the Screen mode depends on the active viewport for its orientation, the X, Y, and Z labels on an axis tripod in an inactive viewport show the orientation of the *currently active* viewport. As a result, the labels on that tripod will change when you activate the viewport it is in.

World—Uses the World coordinate system. Seen from the front, the X axis runs in a positive direction to the right, the Z axis runs in a positive direction upward, and the Y axis runs in a positive direction away from you.

Parent—Uses the coordinate system of the parent of the selected object. If the object is not linked to a specific object, then it's a *child of the world*, and the parent coordinate system is the same as the World coordinate system.

Local—Uses the coordinate system of the selected object. An object's local coordinate system is carried by its pivot point. You can adjust the position and orientation of the local coordinate system, relative to its object, by using the Hierarchy command panel. See chapter 23, "Building Hierarchies."

When Local is active, the Transform Center button is inactive and all transforms use the local axis as the center of transformation. If you have a selection set of several objects, each uses its own center for the transform.

Pick—Uses the coordinate system of another object in the scene.

When you choose Pick, you then need to select the object whose coordinate system the transforms will use. Click to select a single object. The object's name now appears in the Transform Coordinate System list.

Saving the object's name in the list lets you pick an object's coordinate system, change the active coordinate system, and then use the object's coordinate system again at a later time. The list can save up to four object names.

Grid—Uses the coordinate system of the active grid. This can be either the home grid, or a helper grid object you've created. If you change the active grid while the Grid option is active, the coordinate system changes accordingly.

See chapter 5, "Precision and Drawing Aids" for information about grid objects.

Creating a Local Axis

While modeling, it's often helpful to have a temporary, movable local axis so you can rotate or scale about an arbitrary center.

Note: This technique does *not* work for animation. See the following topic, "Choosing a Transform Center," for animation tips.

To create an adjustable local axis:

1. Create a Point helper object.

You create helper objects in the Create panel. Click the Helpers button, click Point, and then click in a viewport.

2. Click Pick in the Transform Coordinate System list, and then click the point object.

The name of the point object appears in the list.

Now you can use the point object's coordinate system as an adjustable axis.

To use the adjustable axis:

1. Place the point object where you want the rotate or scale transform to be centered.

2. Select the object you want to transform.

3. Choose the point object's name in the Transform Coordinate System dropdown list.

4. Click Use Transform Coordinate Center in the Transform Center flyout. (This flyout is discussed in the next topic.)

5. Proceed with the transform.

Choosing a Transform Center

The transform center affects scale and rotation transforms, but has no effect on position transforms.

There are three types of transform center, which you select using the Transform Center flyout on the toolbar. As you change the transform center, the junction of the axis tripod icon moves to the location you specify.

Transform Center flyout

You can choose from these transform centers:

 **Use Pivot Point—** Transforms occur about the center of the object's pivot point.

 **Use Selection Center—**Transforms occur about the center of a bounding box surrounding the current selection.

Note: The selection center of a single object is seldom the same as its pivot point, which might be somewhere other than at the center of the object.

 **Use Transform Coordinate Center—**Transforms occur about the center of the active transform coordinate system.

Coordinate Center Options

When Use Transform Coordinate Center is the active transform center, the active coordinate system affects the transform, as follows:

World—Transforms occur about the origin of world space.

Screen—Transforms occur about the center of the active viewport. Since panning, zooming, or rotating a view can change the position of an object relative to the center of the viewport, these functions also change the relative position of the transform center when you're in Screen mode.

View—In orthographic views, the transforms occur about the center of the active viewport. In User, Perspective, and Camera views, the transforms occur about the world origin.

Local—The Transform Center button is deactivated when Local is active, and all transforms occur about the local axis—or pivot point—of the selected object. If two or more objects are selected, the transform occurs about each of their local axes.

Parent, Pick Object, Grid—The transforms occur about the pivot point of the referenced object: either the parent, the pick object, or the active grid.

See the previous topic for more information about these coordinate system options.

Multiple Objects and the Transform Center

The position of the transform center changes depending on whether you've selected a single object or multiple objects.

By default, when you select multiple objects the transform center changes to Use Selection Center, because selection sets have no pivot point. However, the type of center you choose while the selection is active determines the center that will appear when you next make a selection—either multiple or single.

For example, if you select a single object and choose Transform Coordinate Center, then select multiple objects and choose Pivot Point, when you next select a single object, the center switches to Transform Coordinates. Then, when you select multiple objects, the center switches back to Pivot Point.

Animation and the Transform Center

Because of the nature of keyframing, rotation and scale transforms can only be properly keyframed using an object's local pivot point. For example, while modeling, you can rotate an object that's offset from the world origin around the world center coordinate system. The object sweeps around the origin in a large arc. However, if you attempt to animate this, the object rotates about its local axis and moves in a straight line from one end of the arc to the other.

Note: To avoid this discrepancy, if the Animate button is on and either the Rotate or Scale buttons is active, the Transform Center flyout is inactive and set to Use Pivot Point. When the Animate button is off, all transforms use the center settings as previously described.

You can override this behavior by turning off the Local Center During Animate parameter in the Preferences dialog.

Keep in mind that this only affects the *center* of the transform. The orientation of the selected transform coordinate system is still in effect.

Animating "Off-Center"

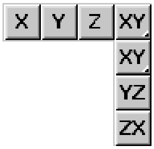
You can animate a rotation or scale about an off-center point by linking your object as child to a dummy helper object, and then rotating or scaling about the dummy.

Another technique is to offset the pivot point of your object using the Hierarchy command panel.

See chapter 23, "Building Hierarchies," for information about linking, dummy objects, and the Hierarchy panel,

Using the Axis Constraints

To the right of the Transform Center flyout on the toolbar are the Axis Constraint buttons. These let you specify one or two axes about or along which the transform takes place.



Axis Constraint buttons

Only one of the four Axis Constraint buttons is active at a time. When a button is turned on, transforms are constrained to the axis it specifies. For example, if you turn on the X button, you can only rotate an object about the X axis of the current transform coordinate system.

The axis or axes to which you're constrained are highlighted in red on the axis tripod icon in viewports.

The first three buttons constrain the transform to a single axis. The third button is a flyout that specifies double-axis combinations.

For example, if the YZ button is active, you can move an object only along the YZ plane, you can scale it only along the Y and Z axes, and you can rotate it about the Y and Z axes but not about the X axis.

Keyboard Shortcuts for Axis Constraints

As an alternative to clicking the buttons in the toolbar, you can use the four function keys F5 through F8 to select the four axis constraint buttons. F5 is the X button, F6 is Y, F7 is Z, and F8 is the double-axis flyout. Press F8 repeatedly to cycle through the three double-axis buttons.

In addition, you can cycle through the buttons by pressing the accent grave key ('). On most keyboards the grave accent is on the same key as tilde (~), and is directly above the TAB key. When the double-axis button is active, pressing SHIFT+' cycles through the three double-axis combinations.

Chapter

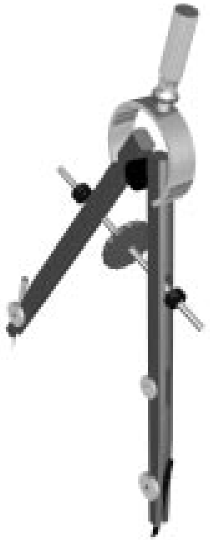
5

Precision and Drawing Aids

3D Studio MAX gives you control over the positioning and alignment of objects in 3D space. This chapter describes the tools that make precision possible. With these tools, you can do the following:

- Choose units from the most common real-world measuring systems or define your own.
- Use the home grid as a construction plane, or use grid objects to position custom construction planes.
- Select different options to align objects with grids, points, and normals.
- Use 3D object snaps on a modeless dialog as you build and move geometry in your scene. Grid points and lines are among the many snap options.
- Use “helper objects” in your work. Grid objects are in this category, along with a number of objects used for positioning and measurement.

Tools for Precision



A set of interrelated tools in 3D Studio MAX gives you precise control of the scale, placement, and movement of objects in your scene. These are especially important tools for those who build precise models in real-world units of measurement.

Basic Tools

The tools for precision are grouped as follows:

Units—Define different measurement systems. Besides the generic unit, you have your choice of feet and inches in both decimals and fractions. Metric units range from millimeters to kilometers. You can also define other units.

Grids—Include the home grid and special *grid objects*. Both types of grid can act as construction planes. 3DS MAX constructs objects using the orientation and position of the active grid. While the home grid is fixed in world space, you can rotate grid objects and place them anywhere in a scene, and align them to other objects and surfaces. You can also give each grid object its own spacing, and display any grid as a dedicated viewport.

Object alignment—Matches an object with the position, orientation, or normal of another object, or to a point in space.

Object snaps—Ensure precise placement when creating and rearranging objects. Keyboard shortcuts let you change object snaps as you work. You can also set snaps to find grid lines and intersections. An angle snap sets the increment for rotation, and a percent snap sets the increment for scaling.

Helpers—Provide useful assistance, as the name implies. These are specialized tools in the same category as grid objects. A Tape object measures distances in current units, and a Protractor object measure angles. A Point object marks a particular spot in 3D space.

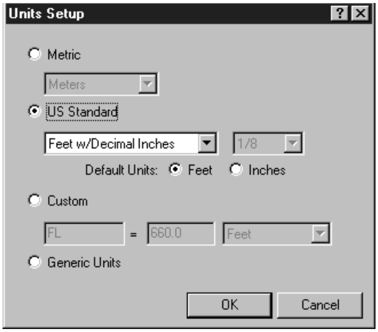
How the Tools Work Together

The tools themselves establish a general order of use and interaction, although you can always change settings as needed without following this sequence.

- Choose a measuring unit. The default is generic units, sufficient for many purposes.
- Set grid spacing (the size of the smallest square), based on the measuring unit. The home grid and grid objects can have their own spacing.
- Move and align grid objects to a useful orientation.
- Set or vary snap settings as needed in your work.
- Use other helper objects like Point and Tape as part of the precision process.

As you work, you can change your settings—including the measuring unit—without losing any precision.

Using Units



Units are the key to connecting the three-dimensional world of 3D Studio MAX with the physical world. You define the units you want to use from a single dialog, Units Setup.

To display the Units Setup dialog:

- Choose Views > Units Setup from the menu bar.

Changing Units

When you change units, the program *displays* measurements in the new unit for your convenience. All dimensions in 3D Studio MAX read out in the new unit. Essentially, you're using a new "measuring stick." No object is changed in this process. As in the physical world, objects in 3DS MAX maintain their absolute size, regardless of how you measure them.

Type-In Entry

When you enter a number, 3DS MAX always assumes the current unit. You can also enter a series of numbers, and the software then converts the sum into the current unit. Here are some examples that assume you're working in centimeters:

- When you enter a dimension of **1'** (one US foot), the software converts it to **30.48CM**.
- If you enter a series of numbers like **14 286 175** (separated by spaces), the software totals the series to **475.0CM**.
- If you enter **1' 1** (one US foot and 1 centimeter), the software converts and combines the two into **31.48CM**.

When you use US Standard, you can select either feet or inches as the default for type-in entry. If you select feet and enter **12**, the result is **12'0"**. However, if you enter **1'2**, the software identifies the second digit as inches, producing **1'2"** as the result.

In any unit system, you can enter fractional amounts. Assume you're working in US Standard with feet as the default:

- If you enter **18/3**, the result is **6'0"**.
- You can change units on the fly. If you enter **18/3"**, the result is **0'6"**. Similarly, you can change the fraction to centimeters. If you enter **18/3CM**, the result is **0'2.362"**.

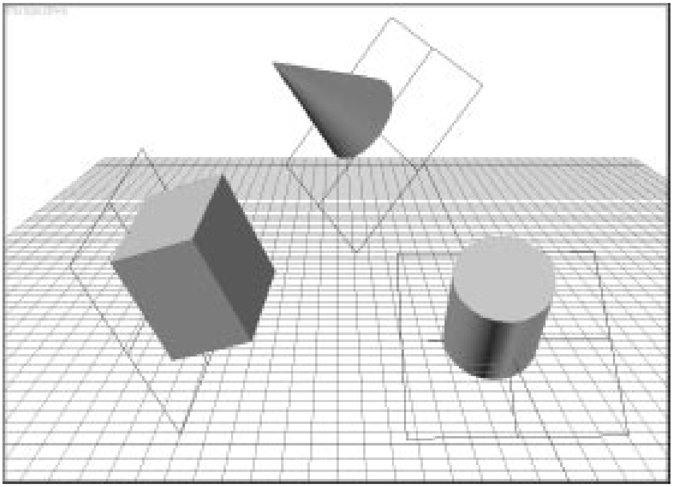
Understanding the System Unit

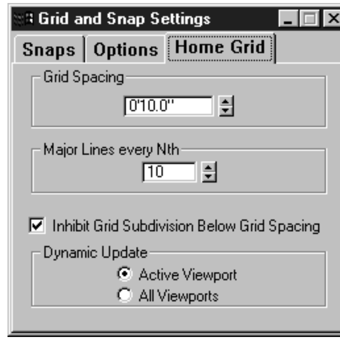
3DS MAX keeps track of all measurements in its own internal *system unit*. No matter what kind of units you use, the program maintains measurements in this absolute unit for storage and computation. This means you can merge a model created with any standard unit into your scene at true scale. 3DS MAX converts the model to the units currently active in your scene.

The default system unit in 3DS MAX is defined as 1.000 inch. As long as the system unit is left at one inch, you can freely share models and change units on the fly with no effect on the underlying geometry. Except in rare circumstances, you never need to change this default scale.

Warning: Do not reset the system unit unless you have a clear need to do so. If the need arises, see instructions in online reference.

Using Grids





Grid Objects used as construction planes

Grids are two-dimensional arrays of lines similar to graph paper, except that you can adjust the spacing and other features of the grid to the needs of your work.

In 3D Studio MAX, grids have these primary uses:

- As an aid in visualizing space, scale, and distance.
- As *construction planes* where you create and align objects in your scene.
- As a reference system for using snap.

Home Grid and Grid Objects

3DS MAX provides two kinds of grids: the home grid and grid objects.

Home grid—The basic reference system, defined by three fixed planes on the world coordinate axes.

- The home grid is present by default when you start 3DS MAX, but can be turned off.
- You can use any view of the home grid as a construction plane.

See chapter 2, "Viewing and Navigating 3D Space" for a complete introduction to the home grid.

Grid objects—A type of helper object you can create whenever you need a local reference grid or construction plane somewhere other than the home grid.

The following are features of grid objects:

- You can have any number of grid objects in your scene, but only one can be active at a time.
- When active, a grid object replaces the home grid in all viewports.
- Each grid object has its own set of XY, YZ, and ZX planes. You can freely move and rotate grid objects, placing them at any angle in space, or attach them to objects and surfaces.
- You can change viewports to display a plan or top view of any active grid object.
- Grid objects can be named and saved like other objects, or used once and deleted.

Using the Home Grid

The home grid provides ready-to-use construction planes, much like a leveled building site marked with stakes and strings. When you create an object in a viewport, the new object is placed on the home grid plane of that viewport.

To use the home grid effectively for construction, you often need to change the defaults to the job at hand—analogous to moving the stakes and strings to match your own site plan.

Changing Home Grid Settings

The home grid is a single system—its three planes use the same settings for grid spacing and major line divisions. You change these settings from a single panel of the Grid and Snap Settings dialog.

To access the Home Grid panel:

- Choose Views > Grid And Snap Settings, then click the Home Grid tab.

Setting Grid Spacing

Grid spacing is the size, in current units, of the grid's smallest square. The basic idea is to choose a grid spacing that corresponds to your unit of measurement, then choose a larger spacing for multiple units.

For example, if you have units set to centimeters, you might make one grid space equal to 1.000 (one unit, or one centimeter in this case).

Setting Major Grid Divisions

The home grid displays heavier or "major" lines to mark groups of grid squares. You can use these to represent larger units of measurement.

For example, if you use a grid spacing of one centimeter, you might use a value of 10 so the major grid divisions represent one decimeter.

In perspective viewports, the displayed size of the home grid adjusts for different major grid divisions. The home grid itself is unchanged.

Setting Color and Intensity

To improve grid visibility, you can change the intensity or color of the home grid. Choose Files > Preferences > Color panel. See online reference for details.

Using Grid Objects

Grid objects let you bypass the home grid and work on separately defined grids to create and position objects. You can use as many grid objects as you like, each serving as a custom construction plane with its own grid settings.

Grid objects are 2D parametric objects, with adjustments for overall size and grid spacing. You can adjust their orientation in world space, and match them to a particular surface or object.

Creating and Modifying Grid Objects

Grid objects are in the category of helper objects on the Create panel.

To create a grid object:

-  Click the Helpers button, then Grid, and drag in a vierwport.

  This creates and selects a grid object, which appears in white wireframe, divided into four quadrants with coordinate axes at the center.

Grid objects are named when you create them and are saved with the scene. They can be deleted at any time.

Like other objects created in 3D Studio MAX, grid objects are placed on the grid of the current viewport. By default, this is a plane of the home grid, but can also be another *activated* grid object.

On the Modify panel, parameters for a selected grid object include:

Grid Size—Sets overall size of the grid object. This size determines the extents of a viewport set to the grid object. It does not affect the useful limits of the grid, which continues as a plane beyond its visible size.

Grid Spacing—Sets the size of the smallest square in the visible grid. This setting appears on the status line when the grid is activated.

Transforming Grid Objects

Like other objects in 3D Studio MAX, grid objects can be moved anywhere in your scene. You can also rotate them freely in space.

Scaling Not Advised—As a rule, don’t use scaling to resize a grid object. Scaling enlarges or reduces the *apparent* size of the grid object but has no effect on grid spacing. A sphere 20 units in radius created on a grid object appears smaller than another 20-unit sphere created on a scaled-up version of the same grid.

If you want to increase or decrease the *actual* size of the grid object, select it and go to the Parameters rollout on the Modify panel.

Activating Grid Objects

Grid objects require *activating* before use. Only one grid object can be active at a time. The home grid is switched off when a grid object is activated.

To activate a grid object:

- Right-click a selected grid object and choose Activate Grid Object from the shortcut menu, or choose this option from Views > Grid on the menu bar.

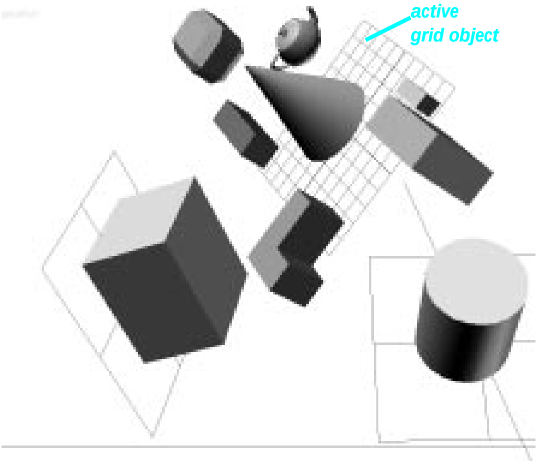
  The grid object changes to show its internal grid structure. Except for its main axes, the home grid disappears in all viewports. The grid object is now an active construction plane.

To return to the home grid:

- Choose Activate Home Grid from the shortcut menu, or from Views > Grid. This deactivates the grid object and returns the home grid in all views.

- If you delete an activated grid object, the home grid is also reactivated.

Viewing Grid Objects



An activated grid object creates a true plane in 3D space. No matter how small an activated grid object appears on screen, its plane is effectively infinite, just as if it were a plane of the home grid. However, you can view a given grid object in a number of different ways.

Setting the Display Plane

You can adjust the visible plane of a selected or activated grid object on the Modify panel.

To set the display plane:

1. Open the Modify panel to display the Parameters rollout for the selected grid.

2. Under Display, select any of the three planes: XY, YZ, or ZX.

   The grid rotates in world space to display the corresponding plane, always based on the grid's local axis.

You can move or rotate a grid, either before or after setting its display plane.

Setting Grid Views

You can set the view in any viewport to an orthographic view as well as the display view.

To set grid view:

1. Activate the grid object.

2. Right-click a viewport label and choose Views > Grid to display a menu of possible views.

3. Choose Front, Back, Top, Bottom, Right, Left, or Display Planes. Display Planes corresponds to the current setting on the Parameters rollout (XY, YZ, or ZX).

   The viewport now shows this view of the grid.

You can set different viewports to different views of the grid. Objects are always created on a view's orthographic grid, even if you rotate the angle of view. In other words, you can set up a convenient view and still construct on the chosen plane.

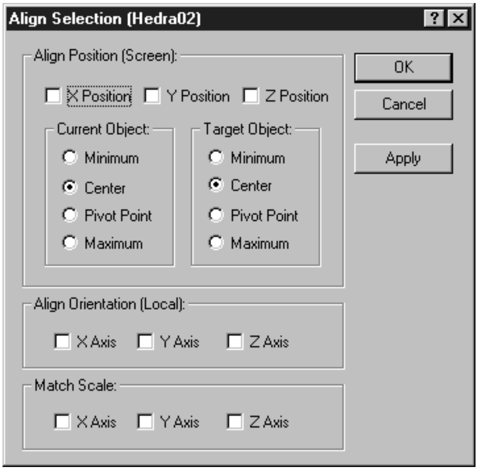
Deactivating Grid Objects

You can deactivate a grid object in any of the following ways:

- Right-click a selected grid object and choose Activate Home Grid from the shortcut menu.
- Select and activate another grid object.
- Delete the grid object.

When you deactivate a grid object, any viewports based on that grid switch to the corresponding orthographic view. For example, a Grid (Front) viewport becomes a Front viewport. A Grid (Display Planes) viewport always switches to Top view, regardless of the currently displayed plane.

Aligning Objects



Any two objects can be aligned, with one as the source object that takes on the alignment of the target object. There are many uses for this feature. For precision, an important use is grid alignment. You can create a new grid object and align it to an existing object. Or conversely, you can move an object onto a grid anywhere in your scene.

Source and Target Objects

Alignment involves two objects: one is the *source* object, where the process begins; the other is the *target* object, where the process ends.

Source object—Object you want to move into alignment with another object. You select a source object to begin the alignment process.

Target object—Object used as the center of alignment. You select the target object during the alignment process. It cannot be selected beforehand.

Setting a Coordinate System

The effects of alignment depend on the current reference coordinate system, such as View, World, Local, and so on. You should decide what system you want to use before beginning alignment.

Reference Coordinate System—Determines the axes used for positional alignment and the size of the bounding box for maximum and minimum positions.

To align objects using active grid axes:

- Choose Grid as the reference system from the list in the toolbar.

To align two objects using their own axes:

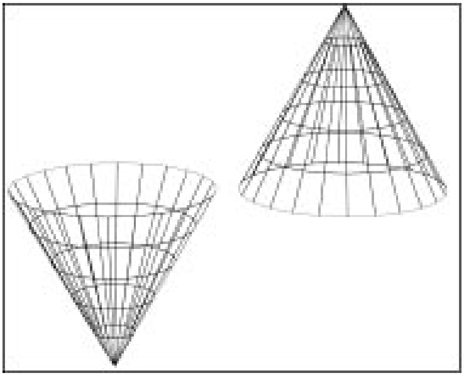
- Choose Local as the reference system. Alignment is then strictly between the two objects. Object bounding boxes determine maximum and minimum positions.

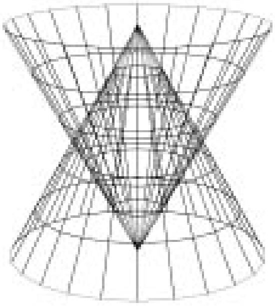
As a reminder, the current reference system appears in parenthesis following the Align Position label in the Align Selection dialog.

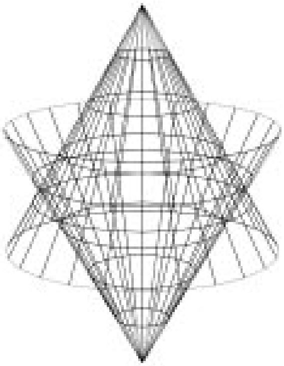
Basics of Aligning Objects

Alignment controls are on a single dialog. As you make a setting, the object being aligned moves immediately to the new position. This lets you experiment with alignment until you get what you want. Settings can be made in any order.

You can also work step-by-step, by applying position choices, for example, before deciding on final orientation. You can cancel at anytime, returning the scene to its original state. You can also undo any alignment and start over.


Original alignment of cones


Cones aligned X, Y and Z center to center


Cones aligned X, Y and Z pivot to center

Aligning Multiple Objects

When you select multiple objects for alignment, the same settings apply to all of them. However, the effect on each individual source object is different. In practice, you're aligning separate objects at the same time with the same settings.

To align a collection of objects as a single unit, select the objects and group them. The alignment now takes place relative to the pivot and bounding box of the entire group.

Sub-Object Alignment

You can use Align with any selection that can be transformed. The tripod axis becomes the source for alignment. You do this by accessing the gizmo level of the object before clicking Align. See online reference for specific steps.

Align Settings and Options

These settings and options are available on the Align Selection dialog.

Current Object—The currently selected source object that is moved during alignment.

Target Object—The object used as the center of alignment.

Center—Geometric center of either object.

Pivot Point—Local pivot of either object.

Minimum, Maximum—Refer to intersections of either object's bounding box with the reference coordinate axes. Maximum occurs in the positive direction along a given axis, Minimum in the negative direction.

If the local axes of an object are skewed to the reference system, the program draws a larger (invisible) bounding box to enclose the object and uses it to calculate Maximum and Minimum.

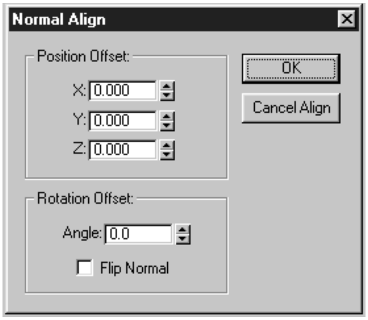
X,Y,Z Position—Offsets for current object along the axes of the reference coordinate system.

X,Y,Z Axis—Local axes of current and target objects. The current object aligns its local axes with the local axes of the target.

Apply—Accepts alignment based on current settings and clears all X,Y,Z settings.

Match Scale—Set any of three axes to match the scale axis values between the two selected objects. There is no change in size if neither of the objects has been previously scaled.

Aligning Normals



In 3D Studio MAX, you can align the normals between any two objects. In the case of mesh objects, the alignment is between individual faces, because each face has its own normal. See chapter 19, “Adjusting Normals and Smoothing.”

For objects without face normals (such as helper objects, space warps, and particle systems), Normal Align uses the Z axis and origin of the object as a normal.

Normal Align also works with sub-object selections and loft objects. For smoothing groups, this command uses the interpolated normal. See online reference for more information.

Basics of Aligning Normals

Before you begin, select a view that lets you see both objects you want to align. If necessary, you can navigate the view after selecting the first normal. To see face normals clearly, work in a wireframe viewport. These are the basic steps:

- Select a source object. This is the object that moves during alignment.
-  Click Align Normals in the default toolbar, or choose Edit > Align Normals.
- Drag across the surface of the source object.The Align Normal cursor appears attached to a crosshair. A blue arrow at the crosshair indicates the current normal.
- Move the crosshair and blue arrow until you locate the normal you want to use, then release. The blue arrow remains as reference to the source normal.
- Drag across the surface of the target object.

  A green arrow at the crosshair indicates the current normal.
- Move the crosshair and green arrow until you locate the normal you want to use as a target, then release. The source object moves into alignment with the target normal, and the Normal Align dialog appears

Modifying Normal Alignment

When the Normal Align dialog appears, alignment between normals is complete. The source object is moved and rotated to conform to the target normal.

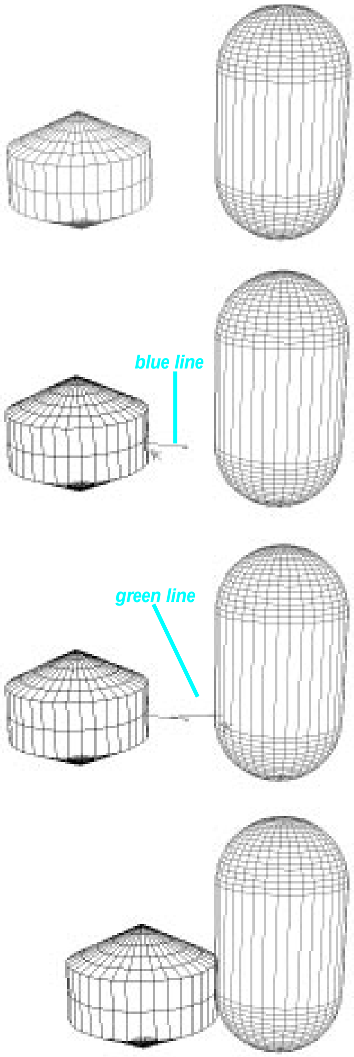
Before accepting this alignment, you can make adjustments in position, rotation, as well as the direction of the normal. These settings are interactive. You see their effect in the viewport as you make them. None are final until you close the dialog.

Position Offset—Moves the source object relative to the coordinates of the target normal.

Setting Z moves the source object in and out along the target normal. Setting X and Y moves the source object back and forth along the corresponding axes.

Rotation Offset—Includes settings for Angle and Flip Normal.

- *Angle* rotates the source object around the Z axis of the target normal. The default alignment produces centered rotation, since the two normals are in the same position. If you reposition the source object, the rotation is off-center.



Normal alignment process

- *Flip Normal* reverses the Z direction of the target normal. The source object, its normal aligned with the target normal, is flipped end for end.

Other Align Options

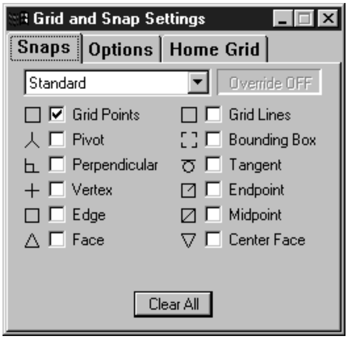
The Align flyout has other buttons that provide specialized alignments.

Align to Camera—Orients a camera viewport to a selected face normal, with the normal in the center of the viewport and on the axis of the camera.

Align to View—Orients a local axis of an object or sub-object selection with the current viewport. Options on the dialog are interactive, as they are on the Normal Align dialog.

Place Highlight—Orients a face normal to a light. See chapter 17, "Lighting Your Scene."

Setting Standard Snaps



Snap panel

Standard snaps give you control in creating, moving, rotating, and scaling objects. You access the snap features in 3D Studio MAX from buttons in the prompt line.

You make most snap settings on Grid And Snap Settings, a modeless dialog with three panels. You can move this dialog to any convenient location on your screen and turn options on and off as you work. Panels are reset to defaults for each new session.

Snap Panel

The most commonly used grid and object snaps appear on the Snap panel. This is the general sequence for using these snaps:

- Click the 3D Snap button to turn it on, then right-click this button to display the Snap Panel. Standard snaps are the default. The list contains NURBS snap options, discussed in chapter 11.
- By default, only Grid Points is cleared as an active snap. Select other snap types to activate them. When you create or move objects, these snaps are now in effect, anywhere in 3D space and unaffected by the current transform coordinate system.
- As you move the cursor, each snap type is marked by a distinctive icon shown on the Snap panel. The current icon indicates the type and position of the next snap.

  Grid points and grid lines are secondary to the other snap types. For example, if Grid Point and Vertex are both active, the software makes a vertex snap in preference to an equally close grid point.
- Turn snaps on and off as needed.

Snap Types

These are the snap types available on the Snap panel. Some types work only on mesh objects, some only on splines, and some on both.

Grid Points—Snaps to grid intersections.

Grid Lines—Snaps to any point on a grid line.

Pivot—Snaps to pivot points of objects.

Bounding Box—Snaps to any of the eight corners of an object's bounding box.

Perpendicular—Snaps to the perpendicular point on a spline, relative to the previous point.

Tangent—Snaps to a tangent point on a spline, relative to the previous point.

Vertex—Snaps to vertices of mesh objects, or object that can be converted to editable meshes.

Endpoint—Snaps to the end points of edges on meshes or splines.

Edge—Snaps anywhere along edges, whether the edges are visible or invisible.

Midpoint—Snaps to the middle of edges on meshes or splines.

Face—Snaps anywhere on the surface of a face. Back faces are culled, so they have no effect.

Center Face—Snaps to the center of triangular faces.

Other Snap Toggles

3D Snap is the default toggle on this fly-out button. Two other toggles limit the range of snaps.

2D Snap—Snaps only to the active construction grid, including any geometry on the plane of that grid. The Z axis, or vertical dimension, is ignored.

2.5D snap—Snaps to the vertices or edges of the *projection* of an object. Projection is relative to the current construction plane.

Suppose you create a grid object and make it active. You then position the grid object so you can see through the grid to a cube further off in 3D space. Now, with 2.5D set, you can snap a line from vertex to vertex on the distant cube. The line is drawn on the active grid. The effect is like holding up a sheet of glass and drawing the outline of a distant object on it.

Snap Override

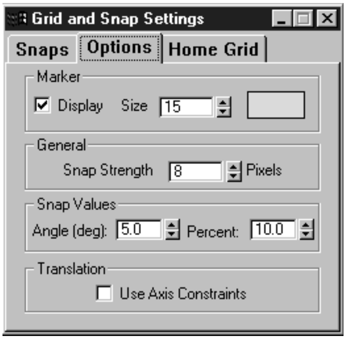
Snap Override bypasses currently selected snaps. You use a keyboard-mouse combination to define a new snap for the next click. You can override on the fly, one snap at a time.

For example, while creating a spline between grid points, you might need to snap to a vertex or midpoint of an object. This is the general procedure:

- As you use preset snaps to create or move an object, hold down SHIFT and right-click. This pops up a small Snap Override dialog.
- From the Standard option on this dialog, choose the snap type you want to use. The cursor switches to this type.
- Make the snap. After the snap, currently set snaps are back in effect.

The Snap Override dialog also contains an option for the last override you used, as well as an option for None. None turns off snaps entirely on the next click.

Setting Snap Options



You access a number of snap features on the Options panel of the Grid And Snap Settings dialog.

- Right-click any of the snap buttons in the prompt line to display the Grid And Snap Settings dialog, then click the Options tab.

Marker settings—Determine the color and size of the snap cursor. Clear Display to turn off the snap cursor entirely.

Snap strength—Determines how close the cursor needs to approach a snap point before the snap takes place. This is a global setting, affecting all snap interactions. Possible values range from 1 to 20, representing the pixels in a “search region” around the active point of the cursor. Default=8.

Settings for Angle and Percent Snap

The following Option settings are for two snap buttons that operate independently of standard snaps.

 **Angle snap—**A global setting, in degrees, that determines the *angle of rotation* for a number of features in 3D Studio MAX, including the standard Rotate transform. As you rotate an object (or group of objects), the object moves around a given axis in the increment you set. Angle snap also affects the following:

- Pan/Orbit camera controls
- FOV and Roll camera settings
- Hotspot and Falloff spotlight angles

 **Percent snap—**Sets a percentage increment during a scaling operation.

Rotating and Scaling with Snaps

The effect of rotating and scaling with snaps depends on whether the Animate button is turned on of off.

Animate Off—Rotations and scales occur around the snap point. For example, using Vertex snap, you can rotate a box about any of its corners.

Animate On—Snap toggles are disabled, while Angle and Percent snaps remain active. Rotation and scaling occur around the pivot point of the object.

Translation Option

By default, the Use Axis Constraints option is clear. The current toolbar setting for axis constraint (XY, for example) has no effect. Setting this option lets you use snaps in conjunction with axis constraints.

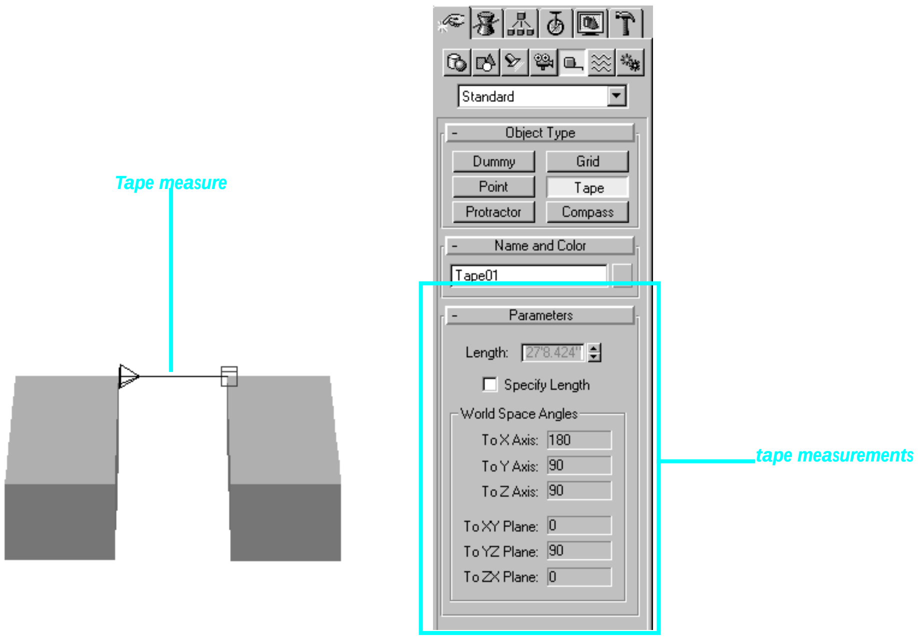
Settings for Spinner Snap

Spinner Snap is set on the General panel of the Preferences dialog. Right-click the button to display this panel.

 **Spinner snap—**Sets a numerical increment for spinner fields. If you’re using generic units of 1 inch, a setting of 12 would let you resize objects by one foot with every click, or add 12 segments to a sphere.

The same setting applies to all spinner fields. Since spinner snap is a toggle, you can easily turn it on when needed and use the default at other times. Spinner snap has no effect on dragging a spinner, only on single clicks.

Measuring Distances



3DS Studio MAX provides a number of options for measuring distances, angles, direction, and object data. Four helper objects create a tape measure, protractor, point, and compass rose. A Measure utility has a floater to display various measurements of any selected object.

Helper Objects for Measurement

Helper objects are located on the Create panel.

To create a helper object:

1.  Click the Helpers button, then click the button for the object you want to create.
2. Either drag or click in a viewport to create the object.

Tape

Using 3D snaps, you can create a Tape object by dragging between any two points in 3D space, much like using a physical tape measure. You then read the length on the Parameters rollout.

If you set Specify Length, the length field lets you enter a value in current units. This is like locking a tape measure to a set length. You can position the tape object and snap to its ends.

- To display length and angle settings, select only the tape end of the object—a small yellow pyramid.
- To move the entire tape including its target, a blue cube, select the connecting line.

Two sets of World Space Angles give you current readouts to the three world axes (X,Y,Z) and the three world planes (XY, YZ, ZX).

Protractor

The protractor reads angles between pivot points of two objects, as measured from the center, the protractor object.

To use Protractor:

1. Click to position it as a protractor object.
2. On its Parameter rollout, click Pick Object 1 and select an object in your scene. The name of the selected object appears above the pick button. A line connects the protractor to the center of the object.
3. Repeat for a second object, then read the angle on the rollout.
4. To move either object while watching the Angle readout, turn on the Pin Stack button in the Modifier Stack rollout. This locks the stack to the Protractor's Parameters rollout.

Point

You can create a point object with a single click anywhere in 3D space. This dimensionless object lets you identify a position that has no other geometry. You might snap a point object to a corner or another object, and use the point as one pick for the protractor.

Compass

The compass object establishes an orientation for your scene. You create this object with a single click, usually in a Perspective or Top view. On its parameters rollout, you can adjust the apparent radius of the compass rose. This object, for reference and display only, does not render.

Measure Utility

This utility provides various measurements of a selected object or shape.

To use Measure:

1. On the Utility panel, click Measure to display the Measure rollout.
2. Select any shape or object and read out the measurements. If you select multiple objects, the sum of the measurements is displayed.
3. Click New Floater to display a modeless Measure dialog with all the same information.
4. Use the Measure dialog to display the length of a spline, like a line or circle, as you create it.

Chapter

6

Basics of Creating and Modifying Objects

In 3D Studio MAX, you model basic parametric objects into more complex ones by:

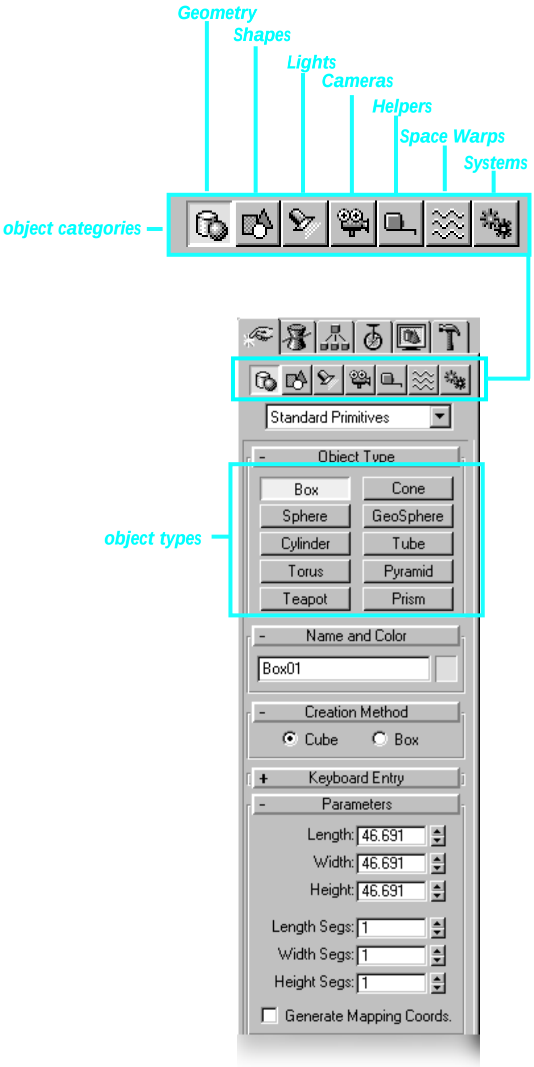
- Changing parameters.
- Applying modifiers.
- Directly manipulating sub-object geometry.

This chapter provides an introduction to creation techniques and modeling possibilities.

The Create panel contains the controls for creating new objects—the first step in building a scene in 3D Studio MAX. Despite the variety of object types, the creation process is generally the same for all objects.

The Modify panel provides the controls to complete the modeling process. Any object can be reworked, from its creation parameters to its internal geometry. Both *object-space* and *world-space modifiers* let you apply a wide range of effects to objects in your scene. The Modifier Stack allows editing of the modifier sequence.

Using the Create Panel



The Create panel provides the controls for creating objects and adjusting their parameters.

To access the Create panel:

1. Click the Create tab in the command panels. By default, this panel is open when you start a new session of 3D Studio MAX.

2. Click an object type to display its Parameters rollout. Box is the object type illustrated here.

The Creation Process

The actual creation of objects is almost effortless, accomplished with a single click of the mouse, a drag, or some combination, depending on the object type. This is the general sequence:

- Choose an object type.
- Click or drag in a viewport to create an object of approximate size and location.
- Adjust the object's parameters and position—either immediately or later.

See "Creating an Object" in this chapter.

Create Panel Interface

Controls in the Create panel vary depending on the kind of object you are creating. However, certain controls are always present, and others are shared by nearly all object types.

Category—Buttons at the top of the panel access the seven main categories of objects in 3DS MAX. Geometry is the default category.

Subcategory—List for selecting subcategories. For example, subcategories under Geometry include Standard Primitives, Extended Primitives, Compound Objects, Loft Object, Particle Systems, Patch Grids, NURBS Surfaces, Doors, and Windows.

Each subcategory contains one or more object types. If you've installed plug-in components for additional object types, these might be grouped as a single subcategory.

Object Type—Rollout containing labeled buttons for creating objects in a particular subcategory.

Name and Color—Text field lists the automatically assigned name of the object. You can edit this name or replace it with another, including duplicate names. Clicking the square color swatch brings up a color selector for the object.

Creation Method —Rollout provides a choice of how you use the mouse to create an object. For example, you can use either the center (radius) or edge (diameter) to define the size of a Circle shape.

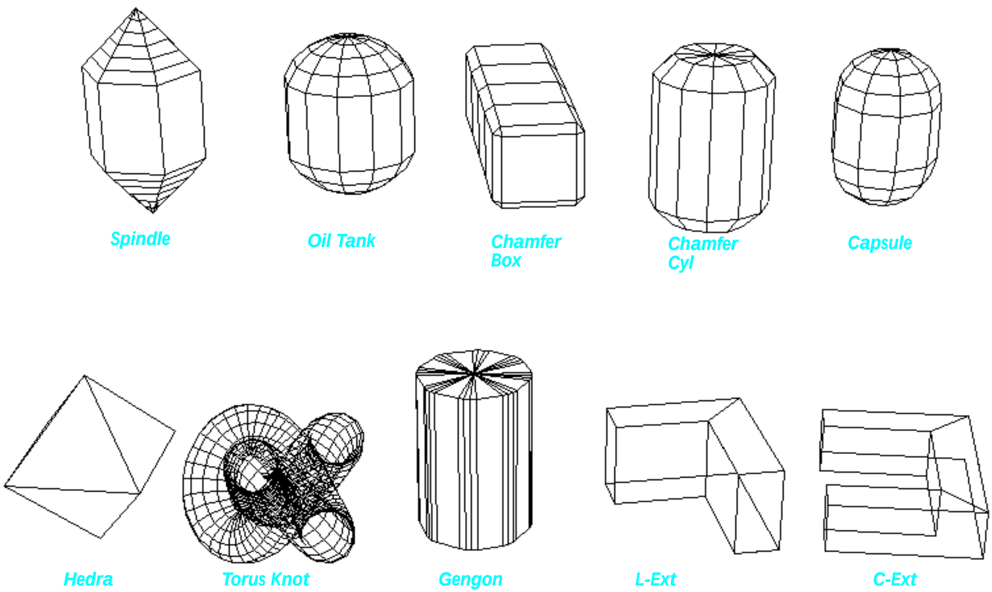
You always have a default creation method. If you want to use an alternate method, choose the option *before* you create the object. The creation method has no effect on a finished object; the options are for your convenience during creation.

Parameters—Rollout shows *creation parameters*—the defining values for that object. Some parameters can be preset, while others are only for adjustment after an object has been created.

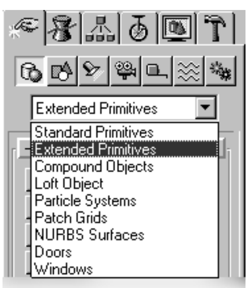
Keyboard Entry—Rollout lets you enter creation parameters from the keyboard for geometric primitive and shape objects.

Other rollouts—Additional rollouts can appear on the Create panel, depending on what kind of object you create.

Identifying the Basic Building Blocks



Extended Primitives



On the Create panel, the categories for Geometry and Shapes supply the "building blocks" to combine and modify into more sophisticated objects. These parametric objects are ready to use. By adjusting values and turning some buttons on or off, you can create dozens of "new" building blocks from the ones listed here.

Geometry Types

See online reference for details about types of objects and individual objects.

Standard Primitives—Relatively simple 3D objects such as a Box, Sphere, and Cylinder, as well as a Torus, Teapot, and Prism.

Extended Primitives—More complex 3D objects such as a Capsule, OilTank, Spindle, Hedra, and Torus Knot.

Compound Objects—Compound objects include Morphs and Booleans. Booleans combine the geometry of two objects using union, intersection, and difference operations. Morphs are animated objects that change one geometric shape into other shapes over time. Shape Merge lets you embed a spline shape into a geometric mesh.

Loft Object—A *super object* that uses shapes as cross sections along a path to produce a 3D object. See chapter 11, "Creating Lofts."

Particle Systems—Animated objects that simulate spray, snow, blizzard, and similar collections of small objects.

Patch Grids—Simple 2D surfaces ready for modeling or repairing existing meshes.

NURBS surfaces—Analytically generated surfaces especially suited for modeling surfaces with complicated curves.

Doors—Parametric doors include Pivot, Sliding, and Bifold styles.

Windows—Parametric windows include Awning, Casement, Fixed, Pivoted, Projected, and Sliding styles.

Shape Types

See online reference for details about types of shapes and individual shapes.

Splines—Common 2D shapes such as a Line, Rectangle, Circle, Ellipse, Arc, Donut, NGon, and Star. Text shapes support True Type fonts. Section creates a spline from the cross-section of an object. Helix is a 3D shape.

NURBS Surfaces—A Point Curve and CV Curve provide the starting points for complex surfaces. See chapter 10, “Creating NURBS Surfaces.”

Varying the Parameters

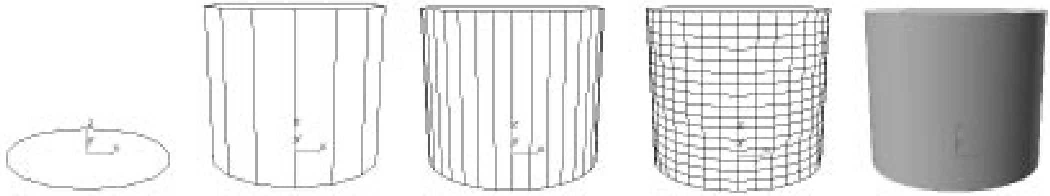
Unlike physical building blocks, with fixed shape and size, you can change the parameters of objects and shapes to dramatically alter topology. Here are some examples of changes you can make:

- Turn a cone into a four-sided pyramid by reducing the number of sides and setting the Smooth option off.
- Slice any circular object as if it were a pie.
- Animate almost all creation parameters, and interactively change their settings during animation playback.
- Render splines directly at any assigned width.

Mapping Coordinates

Most Geometry objects have an option for generating mapping coordinates. Objects need these mapping coordinates if you plan to apply a mapped material to them. Mapped materials include a wide range of rendered effects, from 2D bitmaps to reflections and refractions. See chapter 21, “Designing Mapped Materials.”

Creating an Object



diameter defined | height defined | sides increased | height segments defined | smooth & shadows

With some variations, these steps apply to creating any type of object on the Create panel. For a concrete example, see “Creating Boxes and Cubes.”

Choose a category of object:

1. Click the Create tab to view the Create panel.

2. Click the Geometric button at the top of the Create panel.

3. Choose the subcategory Standard Primitives from the list.

A number of buttons appear on the Object Type rollout.

Choose an object type:

- Click the button for the type of object you want to create.

  The button turns green, showing that it is active. Four rollouts appear: Name and Color, Creation Method, Keyboard Entry, and Parameters.

Choose a creation method (optional):

You can accept the default method and skip this step.

- Choose a method in the Creation Method rollout.

Preset the creation parameters (optional):

You can adjust *all* creation parameters *after* you create an object. Skip this step if you prefer.

In the Parameters rollout, you can set parameters before you create an object. However, the values of parameters you set by dragging the mouse—for example, the Radius and Height of a cylinder—have no effect until *after* you create the object.

Create the object:

1. Press at a point in any viewport where you want to place the object, and *hold the mouse button down*—do not release the button.

2. Drag the mouse to define the first parameter of the object—for example, the circular base of a cylinder.

3. Release the mouse button. The first parameter is set with this release.

4. Move up or down *without touching the mouse button*. This sets the next parameter—for example, the height of a cylinder.

- *If you want to cancel:* Until you complete the next step, you can cancel the creation process with a right-click.

5. Click when the second parameter has the value you want, and so on.

The number of times you press or release the mouse button depends on how many spatial dimensions are required to define the object. (For some kinds of objects, such as Line or Bones, the number is open-ended.)

When the object is complete, it is in a selected state and ready for adjustments.

Name the object (optional):

- Highlight the default object name in the Name and Color rollout, and then enter a name you choose.

Adjust the parameters:

- Change the creation parameters *immediately after you complete an object*, while it's still selected. Otherwise you can change them later on the Modify panel.

You can also change the object's display color by clicking the color swatch in the Name and Color rollout, and then using the Object Color dialog to choose a new color.

While making adjustments, you can use viewport navigation controls like Zoom, Pan, and Arc Rotate to change your view of the selected object. You can also adjust the time slider.

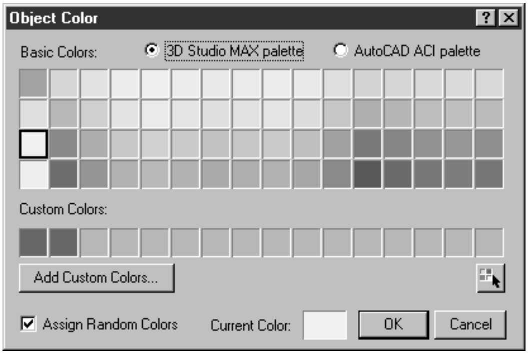
When you end the creation process, the Parameters rollout disappears from the Create panel. Once this rollout is no longer visible in the Create panel, you must go to the Modify panel to adjust the object's creation parameters. See later topics in this chapter.

Ending the creation process:

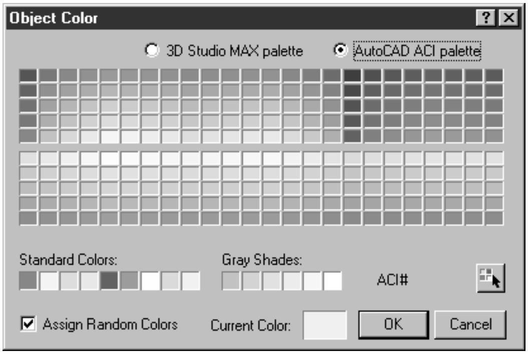
While the object type button remains active, you can continue creating objects of the same type until you do one of these actions:

- Select an object other than the one you created most recently.
- Transform an object.
- Change to another command panel.
- Use commands other than viewport navigation or the time slider.

Assigning Colors to Objects



Object Color dialog, 3DS MAX version



Object Color dialog, AutoCAD version

3D Studio MAX is a true color program. When you pick a color in 3D Studio MAX you are specifying 24 bits of color data which provides a range of over 16 million colors.

Object wireframe colors are used primarily as an organizational tool in 3D Studio MAX. Object naming strategies, named selection sets, and object wireframe color strategies provide a rich set of tools for organizing even the most complex scenes.

You can use two dialogs to specify colors:

- The Object Color dialog contains a preset palette of colors that you use to set an object's wireframe color. This is also the surface color you see in a rendered viewport.
- The Color Selector is a generic dialog that you use to define any color in the 24-bit color range.

Switching Between Palettes

You can alternate between two versions of the Object Color dialog at any time by clicking the appropriate toggle:

- A 3D Studio MAX version contains a fixed palette of 64 colors, plus a custom palette of 16 user-defined custom colors.

  Use this version when you want to work with a smaller palette of colors or when you want to define custom object wireframe colors.

- An AutoCAD-compatible version contains a fixed palette of 256 colors matching the colors in the AutoCAD Color Index (ACI).

  Use this version when you want to assign object colors that match the AutoCAD Color Index. Using ACI colors is useful if you plan to export objects to AutoCAD and want to organize them by object color.

Setting Object Color

This is the general procedure for selecting object color.

To select object color:

1. Select one or more objects.
2. Click the color swatch to the right of the Object Name field to display the Object Color dialog.
3. Click a color swatch from the palette and click OK to apply the color to the selection.

Using Random Color Assignment

By default, 3D Studio MAX assigns colors randomly as objects are created. The colors are chosen from the current palette in the Object Color dialog.

To create objects of the same color:

- Choose the color you want to use and clear Use Random Colors.

  Newly created objects appear in this color until you change the setting.

Defining Custom Colors

The 3DS MAX version of the Object Color dialog contains a palette of 16 custom color swatches. You can define any color for each of the 16 color swatches. See “Using the Color Selector.”

To define a custom color:

1. On the 3DS MAX version of the Object Color dialog, click one of the 16 custom color swatches.
2. Click Add Custom Colors to display the Color Selector.
3. Define a custom color and click Add Color.

   The custom color is stored in the selected color swatch of the Object Color dialog and is set as the current color.

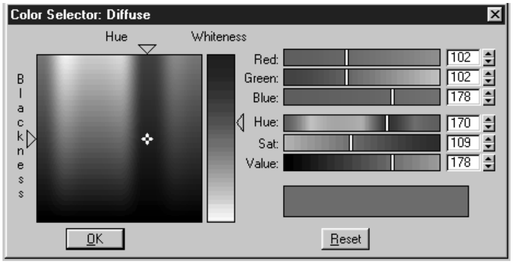
You can also get a custom color from an object in your scene and drag that color into one of your custom color swatches.

- Drag the Current Color swatch up to one of the custom color swatches.

Selecting Objects by Color

Click the Select By Color button to display the Select Objects dialog. You can use this dialog to select objects based on their wireframe color. See chapter 3, “Selecting Objects”.

Using the Color Selector



You use the Color Selector whenever you specify a custom color parameter in 3D Studio MAX. You can work simultaneously with three different color models to help you zero in on the exact color you want.

The Color Selector is used to specify many color parameters such as light colors, material colors, background colors, and custom object colors.

To display the Color Selector:

1. On the Object Color dialog, click the Current Color swatch. You can also click the color swatch of any color parameter, such as a light color or a material color.

2. Make a color selection and click Close.

3. To keep the original color, click Reset.

The Color Selector is divided into three different color selection models. You can use the controls for any model to define a color. The three color models are:

- Hue|Blackness|Whiteness (HBW)
- Red|Green|Blue (RGB)
- Hue|Saturation|Value

As you adjust the controls of one color model, the controls of the other two models change to match. The color defined by the color model is displayed in the right swatch of the Color Output box. The original color, before you began making changes, is displayed in the left swatch.

Changing the HBW Color Model

The most prominently displayed and intuitive color model is the HBW model. This model represents a natural, pigment-based way of mixing color by starting with a pure color (*hue*) and then making it darker by adding black, or lighter by adding white.

The main feature of this model is a large square box displaying the color spectrum. Across the top of this box you have the spectrum of pure colors, or hue. Down the side of the box you see increasing levels of *blackness*, making the color darker as you approach the bottom.

Hue—Define a pure color by dragging the hue pointer across the top of the box.

Blackness—Drag the blackness pointer down the side to darken the pure color by adding black.

You can also click or drag inside the box to change hue and blackness at the same time.

Whiteness—Vertical bar to the right controls the amount of whiteness. The color set by the hue and blackness pointers is displayed at the top of the bar and pure white at the bottom. Drag the whiteness pointer down to lighten the color by adding white.

Changing the RGB Color Model

The RGB model adjusts the mix of Red, Green, and Blue to define a color. This model represents the way colored light is mixed.

The controls for the RGB are three horizontal sliders with matching numeric fields, one for each color component.

The position of the slider sets the amount of Red, Green, or Blue used in the color mix.

- When a slider is all the way to the left, its field reads 0. None of the color controlled by that slider is being used.
- If the slider is all the way to the right, the field reads 255. The maximum amount of that color is being used.

Changing the HSV Color Model

HSV represents the traditional color model of Hue, Saturation, and Value. You operate these sliders the same as the RGB sliders, but the range of 0 to 255 has a different meaning.

Hue—Sets the pure color. Locating the slider all the way to the left gives you pure red. As you drag the slider to the right you move through the spectrum or Red, Yellow, Green, Cyan, Blue, Magenta, and back to Red again.

Hue is more accurately represented as a color wheel rather than a linear slider. That is why the Hue slider is red at both ends. Think of the hue range from 0 to 255 as being points on a circle where the numbers 0 and 255 are right next to each other.

Saturation—Sets the purity or strength of the color. A weak color, with saturation near 0, is dull and gray. A strong color, with a saturation near 255 is very bright and pure.

Value—Sets the lightness or darkness of a color. Low values darken the color towards black. High values lighten the color towards white. A value in the middle, a setting of 127, gives you the color defined only by hue and saturation.

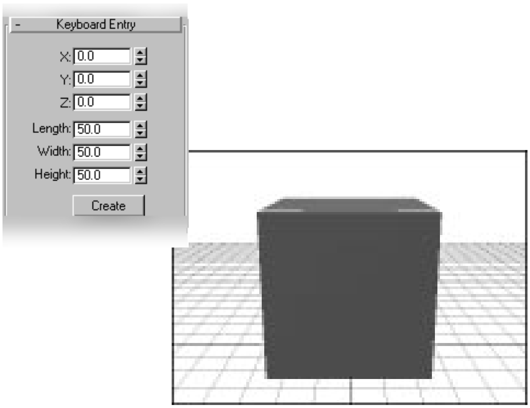
Reading the Slider Background Colors

As you change the color settings in any color model, the backgrounds of all the sliders, except hue, change color as well. The background of a slider displays the current color at a slider's location and shows what will happen to the current color if you drag the slider in a given direction. When you release a slider, the backgrounds update to show the new current color.

For example, imagine your current color is a bright orange. Looking at the Green slider you see orange directly underneath the slider, yellow to the right, and red to the left. This indicates that if you increase the amount of green, by dragging to the right, the color will change from orange to yellow.

Tip: A rule of thumb when working with color sliders is to drag in the direction of the color you want.

Creating Objects from the Keyboard



You can create most geometric primitives and splines from your keyboard using the Keyboard Entry rollout. In a single operation, you define both the initial size of an object and its three-dimensional position. Name and color are automatically assigned.

This method is generally the same for all primitives; differences occur in the type and number of parameters.

To use keyboard creation:

1. Click any of the object types for Standard or Extended Primitives, or for Splines.
2. Click the Keyboard Entry rollout to open it. This rollout is closed by default.

   In the illustration, see the Keyboard Entry rollout for the Box primitive.

   Note: The buttons on the Creation Method rollout have no effect on keyboard entry.
3. On the Keyboard Entry rollout, select a numeric field with the mouse and then enter a number.
4. Press TAB to move to the next field. You do not have to press ENTER after entering a value. Press SHIFT+TAB to reverse direction.
5. When you have all fields set, press TAB to move to the Create button. Press ENTER.

   The object appears in the active viewport.

Once created, a new object is unaffected by the numeric fields in the Keyboard Entry rollout. You can adjust parameter values on the Parameters rollout, either immediately after creation or on the Modify panel.

Settings on the Keyboard Entry rollout are saved during a session. Among other possibilities, this lets you create the same or similar object at different locations in your scene.

Coordinates and Parameters

The Keyboard Entry rollout contains a common set of position fields, labeled X, Y, and Z. The numbers you enter are offsets along the axes of the active construction plane—either the home grid or a grid object. Plus and minus values correspond to positive and negative directions for these axes. The defaults are 0,0,0—the center of the active grid.

The location set by X,Y is equivalent to the first mouse-down position in the standard method of creating objects.

Here are examples of parameters that appear on the Keyboard Entry rollout for some of the Standard Primitives.

Primitive	XYZ point	Parameters
Box	Center of base	Length, Width, Height
Sphere	Center	Radius
Cylinder	Center of base	Radius, Height
Torus	Center	Radius 1, Radius 2
Tube	Center of base	Radius 1, Radius 2, Height
Cone	Center of base	Radius 1, Radius 2, Height
Teapot	Center of base	Radius

Creating Boxes and Cubes

The Standard Primitive Box produces either a rectangular box or a cube. You can vary the scale and proportion of this simple primitive to make many different kinds of rectangular object—from large, flat panels and slabs to long columns and small blocks. Box also provides a concrete example of how many objects are created in 3D Studio MAX.

Creating a box:

1. On the Object Type rollout, click Box.
2. In any viewport, drag to define a rectangular base, then release to set length and width.
3. Move the mouse up or down to define the height.
4. Click to set the finished height and create the box.

Creating a box with a square base:

- Hold down CTRL as you drag the base of the box. This keeps length and width the same. CTRL has no effect on height.

Creating a cube:

1. On the Object Type rollout, click Cube.
2. In any viewport, drag to define the size of the cube.

   As you drag, a cube emerges with the pivot point at the center of its base.
3. Release to set the dimension of all sides.

   You can adjust parameters to vary the length of one or more sides of the finished cube.

Box Settings

The defaults produce a box with one segment on each side.

Length, Width, Height—Set interactively during creation. These fields act as “readouts” while you drag the sides of the box. After creation, you can revise these values. Default=0,0,0.

Length, Width, Height Segments—Sets number of divisions along each axis of the object. Can be set before or after creation. By default, each side of the box is a single segment. When you reset these values, the new values become the default during a session. Default=1,1,1.

Increasing the number of segments creates a more complex geometry, useful if you plan to edit the mesh of the box. For some lighting set-ups, more segments can improve highlights on the finished rendering. Aside from highlights, box sides render the same with single or multiple segments.

Generate Mapping Coordinates—Sets up required coordinates for applying mapped materials to an object. See chapter 21, “Designing Mapped Materials.” Default=cleared.

Creating Circular Primitives

The geometric primitives based on the circle—Sphere, Cylinder, Torus, OilTank, Capsule, and Spindle, for example—share a number of common or closely related settings.

Creation Methods

On the Creation Method rollout, you set the method for creating circular primitives—either Edge or Center. The choice determines where the first point appears when you hold the mouse button down to begin creating an object.

Edge—First point on the diameter of the circle. Useful for creating objects tangent to one another.

Center—First point at the center of the circle. Useful when the exact location of the center is important.

Rotating into Alignment

When you create a circular primitive, you can use the keyboard to rotate it around the center determined by the Diameter or Radius setting. This is especially useful for aligning the sides of a prism-like primitive with another object or the axes of the construction plane.

- To adjust both the diameter and rotation at the same time, hold down CTRL as you make the first drag in a viewport.
- To make the effect obvious, use a limited number of sides.

Other Common Parameters

Radius—Sets the distance from the center of the primitive to a circumference. All circular primitives have settings for one or two radiuses.

Sides—Sets number of sides around the circular primitive. Higher numbers shade and render as true circles when Smooth is selected. Lower numbers create regular polygonal objects when Smooth is cleared.

Smooth—Smooths the edges between segments for shading and rendering. When cleared, faces of the circular primitive are visible. See “Default Smoothing Groups.” Default=selected.

When the scene is animated, Smooth toggles on or off at the frame where you made the toggle.

Height Segments—Sets the number of divisions along the object’s major axis.

Cap Segments—Sets the number of concentric divisions around the center of an object’s top and bottom. Cap segments only appear in wireframe when the setting is 2 or higher.

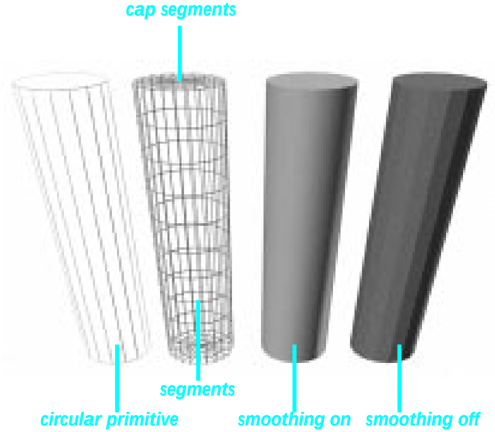
Increase cap segments, for example, when you need to alter the geometry of a cap through subobject selection.

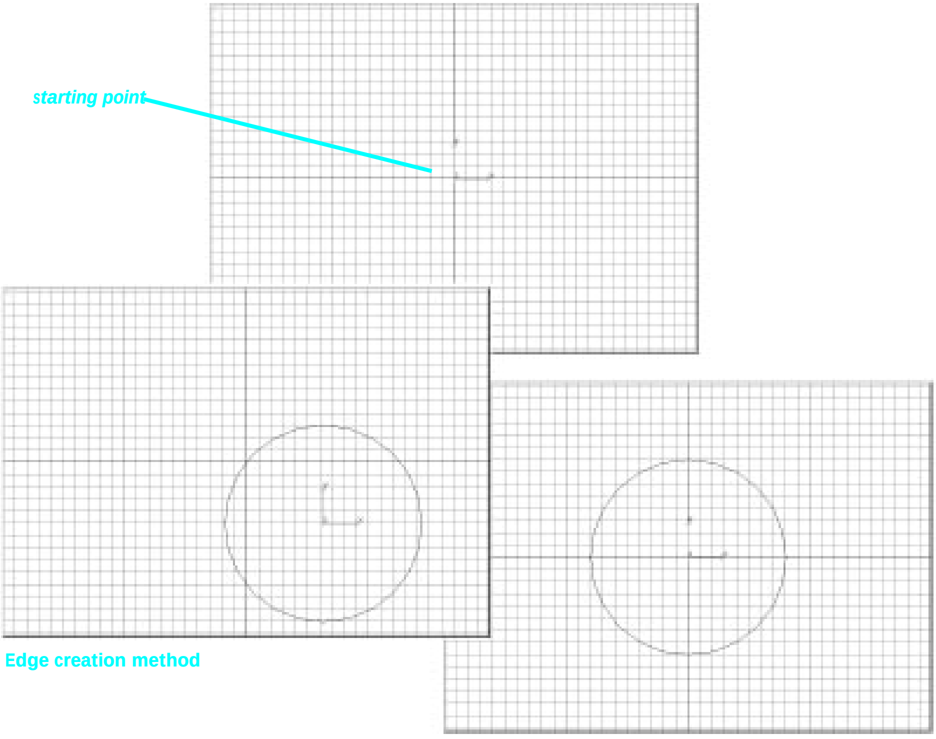
Default Smoothing Groups

The circular geometric primitives all have a smoothing option. When selected (the default), certain angular faces of the object are blended together, creating a smooth appearance in rendered views.

The Smooth settings toggle default *smoothing groups*, which determine where edges appear on a rendered object. For example, a default cylinder has two smoothing groups, one along its length, and one for its two ends. Edges appear between smoothing groups, and you see a cylinder with circular ends. A default sphere has only one smoothing group and is therefore uniformly smooth when rendered.

Default smoothing groups can be changed or adjusted. See chapter 19, “Adjusting Normals and Smoothing.”

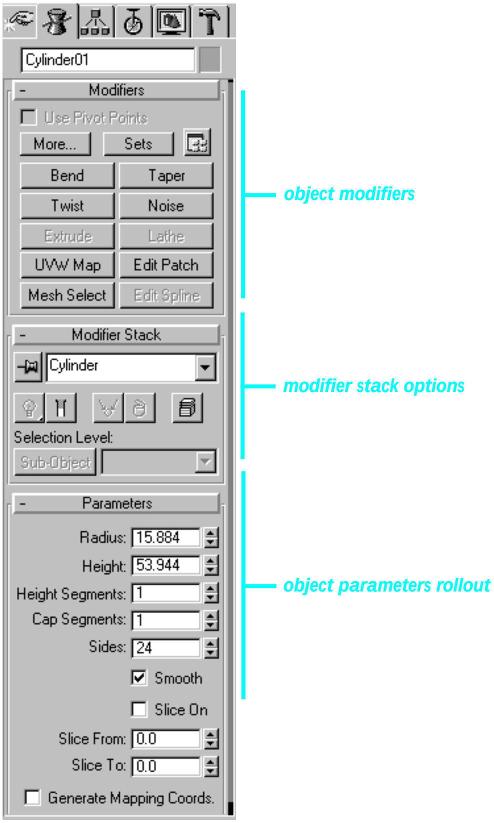




Edge creation method

Center creation method

Using the Modify Panel



Modify panel for a selected cylinder

After adding objects to your scene from the Create panel, you move to the Modify panel to change an object's original creation parameters and to apply *modifiers*. Modifiers are the basic tools for modeling and reshaping objects in 3D Studio MAX.

To use the Modify panel:

1. Select an object in your scene.

2. Click Modify to display the Modify panel.

The name of the object appears at the top of the Modify panel, and various fields change to match this object.

A Parameters rollout is at the bottom of the panel. This is where you change creation parameters for an object. As you change them, the object updates in the viewport.

The Modify panel stays in view until you dismiss it by clicking the tab of another command panel. The panel updates to show whatever options and controls are available for the currently selected object or modifier.

Applying Modifiers

All modifiers are applied in the same way. With the Modify panel open for a selected object, do either of the following:

- Click a button on the Modifiers rollout. This applies the modifier to the selected object.

  You can configure custom button sets to replace the default set of 10. See online reference for instructions.

- Click More in the Modifiers rollout to display a list of additional modifiers. Highlight one and click OK, or double-click the modifier name. This applies the modifier to the selected object.

  The Parameters rollout now shows the settings for the modifier. As you change these settings, the object updates in the viewport.

Using Modifiers

Once you've applied modifiers to an object, you can use the Modifier Stack rollout to find a particular modifier, change its parameters, edit its sequence in the modifier stack, copy its settings to another object, or delete it entirely.

See "Modifier Types Defined" for an overview of available modifiers, and see online reference for details about a specific modifier and its parameters.

General Rules

In 3DS MAX, you can generally do the following:

- Modify anything you can select. This includes any object or set of objects, or any part of an object down to the sub-object level. For example, you can select a single face and apply a Taper modifier.
- Apply an unlimited number of modifiers to an object or part of an object.

  The *order* or sequence in which you make modifications is important. Each modification affects those that come after it. See "Using the Modifier Stack."

How Modifiers Differ from Transforms

Modifiers are a distinct class in 3DS MAX. They are *not* transforms.

- Modifiers, as the name implies, change an object's geometrical structure, deforming it in some way.

When you apply a Taper modifier to the end of a cylinder, the vertices near the end move closer together. Modifiers make changes in the geometry that stay in effect until you adjust or delete the modifier.

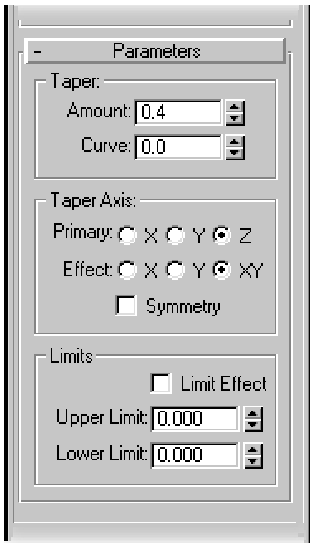
- Most transforms produce equal displacement along one or more axes of an object, or part of an object.

For move, rotate, and uniform scale transforms, the displacement is equal along all three axes. When you rotate a box, all sides remain parallel. In general, all vertices keep the same relative position to one another. The exceptions are Squash and Non-Uniform Scale, which displace axes by different amounts.

- An object can carry any number of modifiers, but only one set of transforms.

Although you can change transform values over time, each object has only one position, one rotation, and one scale transform. Transforms are evaluated on the modifier stack *after* all the object-space modifiers, but before the world-space modifiers. See "Using the Modifier Stack."

Modifier Types Defined



Parameters rollout for the Taper modifier

In 3D Studio MAX, *modifiers* are the basic tools for modeling and reshaping objects. There are two basic types: *object-space* modifiers and *world-space* modifiers.

- Click More on the Modifiers rollout to see the main list of both types.

Note: A large number of modifiers ship with 3D Studio MAX software. Only selected modifiers are mentioned here as examples of their type. See online reference for detailed documentation on individual modifiers.

Object-Space Modifiers

Object-space modifiers are by far the largest set of modifiers. They generally affect one or more selected objects, or part of an object. You can bend a single tree or a whole forest of trees with the same modifier. You can also bend just the branch of a tree.

The following list is a rough grouping of the different kinds of object-space modifiers.

Basic geometric—Apply easily understood, real-world effects to objects. Examples are Bend, Taper, and Twist, all in the default set of buttons, and Skew and Stretch on the More list.

Editable mesh—Supply separate functions found in editable meshes and the Edit Mesh modifier. Examples are Mesh Select, Affect Region, Delete Mesh, Face Extrude, Tessellate.

Free-Form Deformation (FFD)—Used in animation for wildly mutable forms and for general modeling. These modifiers surround the selected object with a lattice box. By adjusting the control points of the lattice, you deform the enclosed geometry.

General—Apply an overall effect to object geometry. Examples are Mesh Smooth, Optimize, and Relax.

Material—Supply functions for material use. Examples include Materials, Smooth, Normal, Camera Map, UVW XForm, and Unwrap UVW.

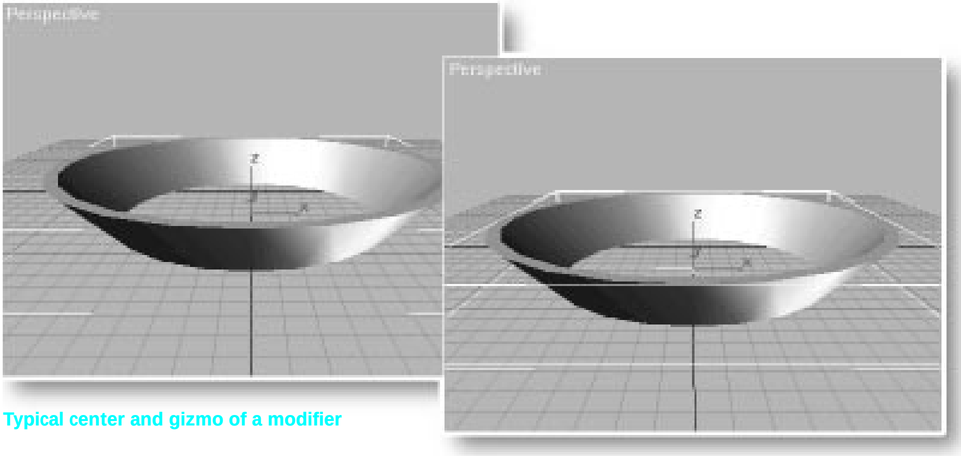
Space Warp—Parallel in object space the effects of space warps in world space. Examples are Ripple and Wave.

Specialized Deformation—Supply deformation functions for particular kinds of geometry. Examples are Patch Deform, Path Deform, and Surf Deform.

Transform—Allow use of transform effects in the modifier stack. Examples are Mirror, XForm.

World-Space Modifiers

World-space modifiers are applied like other modifiers and are carried with the object. The primary difference is that they base their effects on world space instead of object-space. As one example, Map Scaler, a world-space modifier, maintains the scale of a map applied to the object. As you increase the object's size, the map



Typical center and gizmo of a modifier

(of brick or tile, for example) maintains a constant size based on world coordinates. Without Map Scaler, the brick or tile pattern would enlarge with the object.

World-space modifiers include Camera Map, Map Scaler, Patch Deform, Path Deform, and Surface Deform.

Common Features

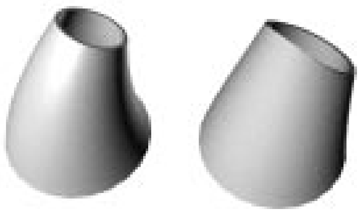
Most modifiers share a number of common features. A modifier typically consists of a *parameter* set and one or more "sub-modifier objects" like a *gizmo* and a *center*.

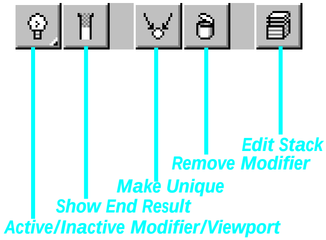
Gizmo—Displays in viewports as a wireframe that initially surrounds the selected object. A gizmo acts like a mechanical apparatus that transfers the modification to the object it's attached to. You can move, scale, and rotate a gizmo as you would any object, altering the effect of the modifier on the object.

Center—Marked with a 3D pivot point like other objects in 3DS MAX. You can move a modifier's center, altering the effect of the modifier on the object.

Gizmos and centers are sub-object features you access from the Modifier Stack rollout. They are discussed in later topics in this chapter.

Using the Modifier Stack





Result of reversing stack order of two modifiers

The modifier stack and its editing dialog are the keys to managing all aspects of modification in 3D Studio MAX. You use these tools to:

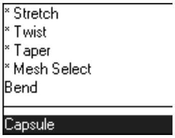
- Find a particular modifier and adjust its parameters.
- View and manipulate the sequence of modifiers.
- Copy, cut, and paste modifiers between objects, or sets of objects.
- Inactivate the effect of a modifier in either the stack or the viewport display, or both.
- Select a modifier's components, such as gizmo or center.
- Delete modifiers.

Examining the Modifier Stack

The modifier stack (or *stack* for short) is a list on the Modifier Stack rollout of the Modify panel. It contains the accumulated history of a selected object and any modifiers you have applied to it.

Internally, 3D Studio MAX "evaluates" an object beginning at the bottom of the stack and applies changes to the object by moving sequentially to the top of the stack. You should therefore "read" the stack from bottom up to follow the sequence used by the software in displaying or rendering the final object.

Here is an example or stack entries for a capsule object:



- At the bottom of the stack, the first entry always lists the object type (in this case Capsule). You click this entry to display the object's creation parameters so you can adjust them. If you haven't applied any modifiers yet, this is the only entry in the stack.
- Object-space modifiers are in the next section of the stack, above the line indicating the division between object and modifiers. You click a modifier entry to display the modifier's parameters so you can adjust them, or delete the modifier.

  If you've only applied object-space modifiers, the stack has only two sections.
- The final, top section of the stack lists any world-space modifiers and space warps bound to the object.

  A double line appears between the object-space and world-space sections. This double line is a placeholder for transforms. It is a reminder that 3DS MAX applies transforms *after* it applies modifiers but *before* it applies world-space information.

Basics of Using the Stack

With the stack feature, no modification you make is "cast in concrete." By clicking any entry in the stack, you go back to the point where you made that modification. You can then rework your decisions or discard the modification entirely by deleting it. You can also insert a new modifier in the stack at that point. The changes you make ripple upward through the stack, changing the current state of the object.

Adding Multiple Modifiers

You can apply any number of modifiers to an object, including repeated applications of the same modifier. As you start applying object modifiers to an object, the modifiers "stack up" in the order they're applied. The first modifier appears just above the object type at the bottom of the stack.

- 3DS MAX inserts a new modifier in the stack *just above the current selection*, but always within the proper section. If you try to insert a world-space modifier between two object-space modifiers, the program automatically places it in the top section.
- If you select the object type or the division line at the bottom of the stack, and then insert a new object-space modifier, it appears at the *bottom* of the other modifiers and becomes the first modifier evaluated.

Effect of Stack Sequence

Since the software applies modifiers in their *stack order* (beginning at the bottom and carrying the cumulative change upward), *where* you put a modifier in the stack is critical. See the figure at the top of this topic. It shows the difference between the objects based entirely on a reversal in the stack order of two modifiers. On one cylinder the bend is first, on the other the taper is first.

Using the Buttons

The Modifier Stack rollout has the following buttons to help you manage the stack:


Pin Stack—Locks the stack and all Modify panel controls to the selected object. You can continue to edit the object even if you select a different object in the viewports.


Active/Inactive Modifier/Viewport—Deactivates the effect of the current modifier. This toggle lets you choose either to the turn off the effect of the modifier on the stack, the viewport display of the modifier effect, or both.

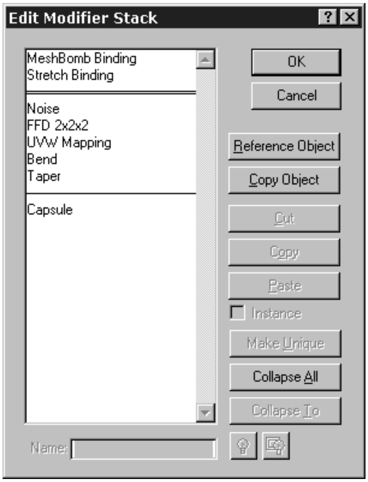
Show End Result—When on, shows the effect of the entire stack on the selected object. When off, shows the effect of the stack only up to the current modifier.

Make Unique—Makes an instanced modifier unique to a selected object. See details in the next topic.

Remove Modifier—Deletes the current modifier from the stack, eliminating all changes caused by that modifier.

Edit Stack—Click to display a dialog where you can edit the stack, as described in the following topic.

Editing the Stack



The Edit Modifier Stack dialog provides tools for copying, cutting, and pasting modifiers within an object's stack, or into the stacks of other objects. Among other features, you can give modifiers explicit names to help you remember the intended effect.

To edit the stack:

1. Select an object.

2.  Click Edit Stack in the Modifier Stack rollout to display the Edit Modifier Stack dialog.

If the object has no modifiers, a menu appears when you click Edit Stack. Choose the Edit Stack option.

Rearranging and Sharing Modifiers

The stack in this dialog is identical to the one on the Modifier Stack rollout. However, you can select any modifier, or group of modifiers, to rearrange or share with other objects. Standard selection techniques apply, including the use of SHIFT and CTRL keys to select multiple modifiers.

To rearrange the list:

1. Select one or more modifiers and click Copy or Cut.

2. Select a new location in the list (either a modifier or a dividing line) and click Paste. The paste occurs immediately above the new location.

To share modifiers with other objects:

1. Select one or more modifiers and click Copy or Cut.

2. Click OK to dismiss the dialog and open the Edit Modifier Stack dialog for another object, or group of objects.

3. Select a location and click Paste.

Tip: Select world-space and object-space modifiers separately. Cut, Copy, and Paste are disabled if both types are selected. If you try to paste a world-space modifier into the section for object-space types, the paste occurs at the top of the world-space section. The reverse is also true.

Unique and Instanced Modifiers

By default, pasted modifiers are *unique*—they lose all connection with the modifier from which they were copied.

- To paste an instanced modifier, set Instance *before* pasting.

Any instance of a modifier controls all other instances. Use this feature when you want a number of objects to take on the same feature. For example, copying an instanced Bend modifier to a number of trees would make them all bend identically. Changing Bend parameters on any one tree would change all the others.

- To remove the instance, select the instanced modifier and click Make Unique.

Using Make Unique with Multiple Items

Assume you have a group of trees all sharing the same instanced Bend modifier. If you select two of them and click Make Unique, a message asks, “Do you want to make the selected objects unique with respect to each other?”

- If you click Yes, the two trees become independent of one another. Each has a unique copy of the modifier and can be bent separately.
- If you click No, the two trees continue to share the same instanced modifier, but separate from the instance in the original group. The two trees can be bent together.

See “How Instanced Modifiers Work.”

Making Copies and References

By selecting the object level of the stack list, you can convert the object to either an independent copy or a reference. The option is the same as Shift-Clone. In this case, the original object is replaced with the new version.

- Select an object and click Reference Object.

  A dividing line appears above the current object-space modifiers. When you add modifiers above this line, they affect only this one object. A referenced object can therefore share some modifiers with other objects and have others that are unique.

- Select an object and click Copy Object.

  The object is now an independent copy, with an identical stack. However, any connections to the stacks of other objects are removed.

See “Appendix” for a detailed discussion of copies, instances, and references.

Collapsing a Stack

You can collapse all or part of a stack into a mesh that preserves the effect of the collapsed modifiers on the base object. You might choose to do this in these cases:

- You’ve finished modifying an object and want to keep it as is.
- You want to discard animation tracks.
- You want to simplify a scene and save some memory. However, some modifiers such as Bevel increase file size and memory use.

After you collapse an object’s stack, you can no longer parametrically adjust either its creation parameters or the individual modifiers affected by the collapse. Animation tracks that were assigned to such parameters also disappear.

- Collapsing a stack does *not* affect the object’s transforms or world-space bindings. Collapsing a stack also has no effect if the stack contains no modifiers.

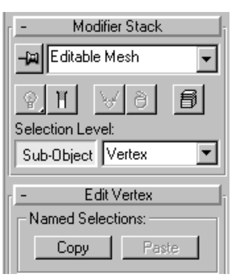
Tip: Before you use either Collapse option, use File > Save Selected to preserve a copy of the original parametric object.

Collapse To—Collapses the stack—up to and including the chosen modifier—into an editable mesh. Modifiers on the stack *above* the chosen modifier are not affected, and you can still adjust them individually.

Collapse All—Collapses the entire stack into an editable mesh. The stack list shows a single entry, Editable Mesh.

See topics later in this chapter on sub-object selection for information on editable meshes.

Modifying at Sub-Object Level



To achieve highly detailed modeling effects, you can directly transform, modify, and align the geometry of objects at a *sub-object* level.

You can also access and transform the sub-object components of modifiers.

The particular geometry available at sub-object level depends on the object type, as shown in the following table. See referenced chapters for details on each object type.

Object Type	Sub-Object Geometry	See Chapter
Meshes	Vertex, Face, Edge	12
Splines	Vertex, Segment, Spline	8
Patches	Vertex, Edge, Patches	11
NURBS	Surfaces, Curves	10

Making a Sub-Object Selection

These are the general steps in setting up an object for sub-object selection. See the chapters referenced above for more information.

- Convert the object into an editable mesh or editable spline. You can also apply any of various modifiers to the object, such as Mesh Select or Spline Select. Work in a wireframe viewport so you can see the geometry.
- On the modifier stack, select the modifier or editable object. Click Sub-Object. The button

turns yellow and stays on until you turn if off or switch to a different command panel. You can only work in sub-object mode while the button is on.

- From the Selection Level list, choose the kind of sub-object geometry you want to work with—Vertex, Face, or Edge, for example. Each selection level has a rollout with its own set of options.
- Use standard selection techniques to select any part of the geometry, from a single component to the entire object. By default, the sub-object selection highlights in red.

Once you make a sub-object selection of geometry, you can do any of the following:

- Apply any options supplied for the kind of object and the selection level.
- Apply standard transforms (move, scale, rotate). See "Transforming a Sub-Object Selection."
- Apply object-space modifiers (Bend, Taper, and Twist, for example, to perform useful modeling operations).
- Bind a space warp to the selection. The rest of the object is unaffected by the warping.
- Use the toolbar commands Align, Normal Align, and Align To View with face selections.

Transforming a Sub-Object Selection

Using an editable mesh or spline, you can directly transform any sub-object selection. However, Mesh Select and Spline Select only perform selection.

To transform a sub-object selection made with a Select modifier:

1. Add an XForm modifier to the stack, following (or somewhere above) the Select modifier.
2. In the stack, open the Select modifier and make a sub-object selection.
3. Choose XForm in the stack. You then transform the XForm gizmo, which applies the transform to the selection.

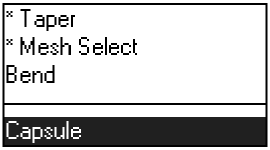
Selecting and Transforming Modifier Components

Most modifiers have sub-object components, such as a gizmo and center. Like sub-object geometry, these components can be accessed and transformed at sub-object level, directly modifying the object's shape.

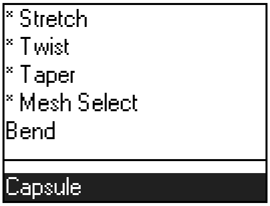
Other modifiers, like those for free-form deformation, have control points and lattices at a sub-object level. Moving these components creates the modeling effects of the modifier.

Using the Stack at Sub-Object Level

Stack with no asterisks



Stack with asterisks on Mesh Select and Taper



Before a second Mesh Select

After a second Mesh Select

With editable meshes and splines, and the Mesh Select and Spline Select modifiers, you can continue to model a single sub-object selection by applying any number of other modifiers. When you go back and change the original selection, the new selection is "passed up the stack" to the modifiers that follow.

Editable meshes and splines have "built-in" sub-object selection at their base level. But the selections you make with Mesh Select and Spline Select work exactly the same way on the stack.

This topic uses meshes for its examples. You can apply the same concepts to splines. Refer to the illustration of the various stack examples.

Working at Two Levels

When you select an object and apply modifiers (Bend and Taper in the illustration), you're working with the object as a single unit, or "whole object"—at the *object* level.

When you make a *sub-object* selection, the stack changes to show you are no longer working with the whole object. In the illustration, you see the effect of a Mesh Select applied between the Bend and Taper. An *asterisk* (*) appears in the stack next to both Mesh Select and Taper to indicate sub-object selection is now in effect.

When applied, Mesh Select automatically turns on sub-object selection for faces. The asterisk next to Taper indicates that the effect of Taper is now applied to the *sub-object selection defined by Mesh Select* instead of to the whole object, as it was originally.

- Sub-object selection carries upward through the stack. If you add more modifiers, each is marked with an asterisk to indicate this state.

By learning to "read the stack," you can move back and forth between the object and sub-object levels while you model an object.

Returning to Whole-Object Level

When you finish modeling a certain sub-object selection, you can return to work on the whole object.

To return the stack to object level:

1. At any point in the stack, apply another Mesh Select modifier.

2. Turn off Sub-Object for this Mesh Select. Any modifiers already on the stack above this modifier no longer show asterisks. Any modifiers you add above the second Mesh Select now apply to the whole object.

3. To continue sending the sub-object selection up the stack, delete the second Mesh Select.

Naming Sub-Object Selections

Sub-object selections are often quite complex, involving a great many small elements that would be impossible to select a second time. For this reason, it's a good idea to name important selection sets using the list on the toolbar. See chapter 3, "Selecting Objects," or online reference.

- Named sub-object selections only appear at the level where they were first named.

  If you have a named set of vertices, you can only access that set at the vertex level of the editable mesh or Mesh Select where you first named the set. This is the only place the selection exists.

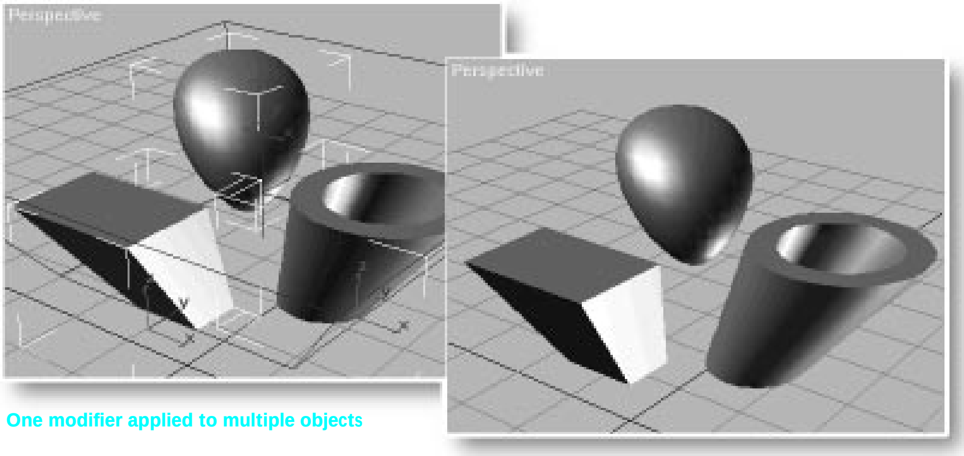
Copying Sub-Object Selections

Once named, sub-object selection sets can be copied between modifiers in the same stack, or to the stack of another object of the same type. Editable meshes and Mesh Select (and their spline counterparts) have buttons for Copy and Paste at each level of geometry.

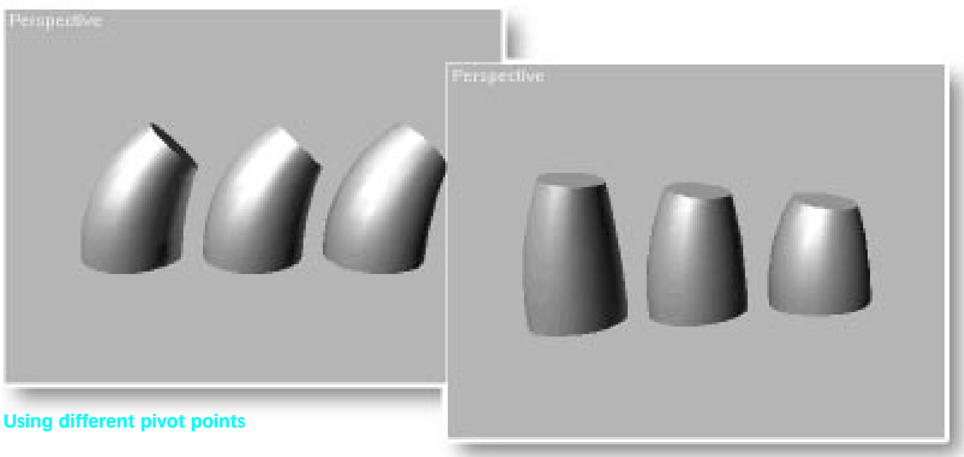
To copy named sub-object selections:

1. Click Copy. From the dialog, choose any available named selection set for that level of geometry.
2. Go to another Mesh Select modifier or to an editable mesh—at the same level of geometry. Click Paste to complete the copy.

Modifying Multiple Objects



One modifier applied to multiple objects



Using different pivot points

3D Studio MAX lets you apply modifiers to multiple objects. In general, the process is parallel to modifying a single object. You make a selection set and apply an available modifier. The modifier then appears on a special stack that refers only to the *commonality* of that selection set.

The Principle of Commonality

When you select multiple objects, 3DS MAX determines what the particular selected set of geometry has in common, if anything.

Given any "commonality" among objects, 3DS MAX presents the options as available modifiers.

- Highlighted buttons in the Modifiers rollout show what modifiers you can apply to the selection.
- Grayed-out buttons represent areas where commonality does not hold.

You can apply modifiers to different categories of objects, depending on the modifier. For example, you might apply a Bend to both a 3D object and a 2D shape. You can apply Mesh Select to a spline primitive and convert it to a mesh, but the reverse is not true—Spline Select is restricted to objects of like category.

To modify multiple objects:

1. Select two or more objects.

   For selection sets, the name at the top of the Modify panel changes to read "Multiple Selected." If the objects are grouped, the group name appears.

2. Decide on the kind of pivot point you want to use. See "Using Pivot Points."

3. Click a modifier button and adjust its parameters.

   If you apply a Mesh Select modifier, you can select geometry on one or more of the objects to use as a sub-object selection set.

Using Pivot Points

The Modifiers rollout has an option called Use Pivot Points, which is grayed out unless multiple objects are selected. Default=cleared.

- When set, the program uses the pivot point of each object as the center of a modifier's operation. If you bend a line of trees around the Z axis, they would all bend along their trunks.

- When cleared, the program calculates a central pivot point for the entire selection set, and modifies the selection as a whole. For a Z-axis bend, trees at the end of a line would deform more than those at the center where the pivot is located.

Tip: Choose the pivot setting *before* you apply the modifier. You can't change the pivot point afterward, although you can delete the modifier and start over without deselecting the selection set.

Instanced Modifiers

When you apply a modifier to multiple objects, each object receives an identical version of the modifier. These are called *instanced modifiers*. They are interchangeable. A change to any one of the instances affects all the other instances. The next topic covers instanced modifiers in detail.

Tip: Sometimes you might apply modifiers to a selection set, perform some other operations, and select the set again, only to find its modifier stack is empty. This happens if you applied a modifier to an *individual* object in the original selection set. When you select the set again, the modifier stack is empty because all members of the set no longer have all modifiers in common. You can still access the instanced modifiers by selecting a single object in the set. The individual object's stacks still contain the modifiers you applied to the set as a whole.

How Instanced Modifiers Work

When you apply a modifier to a selection set, the same modifier is carried on the stack for each individual object. These are *instanced modifiers*—they are all exactly the same, and a change to the instance for any one object will change all the others.

Identifying Instanced Modifiers

You can quickly lose track of which objects share the same modifier. An option on the Views menu highlights those objects.

To identify objects sharing instanced modifiers:

1. Select an object with an instanced modifier. Choose the instanced modifier in its stack.

2. Choose Views > Show Dependencies from the menu bar. Other objects with instances of the same modifier appear in bright green.

Adjusting Instanced Modifiers

You can make changes to an entire set of objects from a single instance. This is a major advantage of instanced modifiers.

To adjust instanced modifiers:

1. Select any object in a set of objects with instances of the same modifier.

2. Choose the instanced modifier in its stack.

   The single object highlights and the appropriate gizmos appear for the entire selection set. Adjustments to this modifier now affect the entire set.

Making Instanced Modifiers Unique

At some point in your work, you might want to turn a modifier instance into a local copy that affects only a single object. The Modifier Stack rollout has a dedicated button. You can also use the Edit Modifier Stack dialog

To make an instanced modifier unique:

1. Select an object with an instanced modifier.

2. Choose the instanced modifier in its stack.

3.  Click Make Unique in the Modifier Stack rollout. If Show Dependencies is set, the green highlight disappears from the other objects.

   The modifier is now separate from the set of instanced modifiers. Adjustments you make to this modifier no longer affect other objects. Its parameters and gizmo remain unchanged from their original, instanced settings until you adjust them.

To make multiple modifier instances unique:

1. Select two or more objects with the same instanced modifier. The stack now shows what the objects have in common.

2. Choose the instanced modifier in the stack.

   There can be more than one instanced modifier in this stack. Click the one you want to make unique for each of the selected objects.

3.  Click Make Unique.

   A message asks, “Do you want to make the selected objects unique with respect to each other?”

4. Click Yes to make the two objects become independent of one another. Click No for the two objects to continue to share the same instanced modifier, but separate from the instance in the original group.

The parameters for this modifier disappear, because the objects no longer share the modifier. For each object, the modifier is now separate from the set of instanced modifiers.

As with a single object, the parameters and gizmo are unchanged in the now unique modifiers.

- To access the unique modifiers, select the objects individually.

Chapter

7

Creating Copies and Arrays

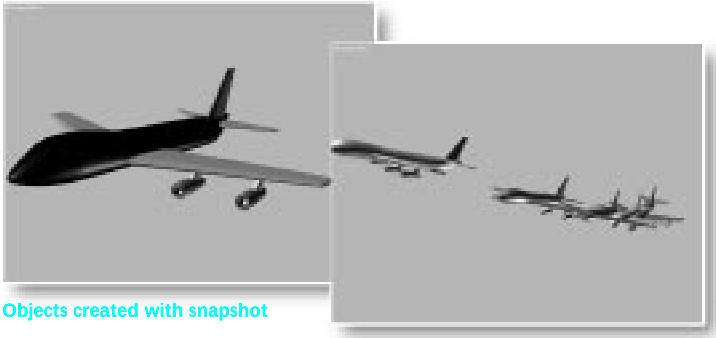
With 3D Studio MAX, you can quickly create multiple versions of one or more selected objects. You do this while you move, rotate, or scale the selection.

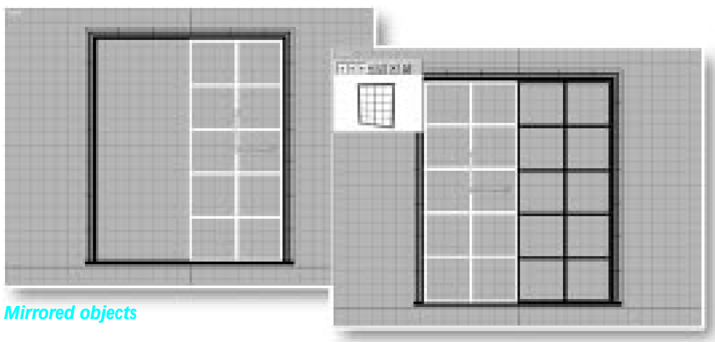
The general term for duplicating objects is *cloning*. This chapter presents all the methods and choices available for cloning objects in 3DS MAX. In addition to the transform method, the tools include the following:

- *Array* lets you set all three transforms, in all three dimensions, at the same time. The results are precise linear and circular arrays in 2D or 3D space.
- *Mirror* produces a “reflected” clone about one or more axes. If you mirror an object without cloning, the result is a “flip” to a new location.
- *Snapshot* lets you create clones equally spaced over *time or distance*, based on an animation path.

Any of the cloning techniques can be animated. This chapter supplies specifics of animation for each of them.

Techniques for Cloning Objects


Objects created with snapshot


Mirrored objects

3D Studio MAX provides a number of different techniques for copying or duplicating objects—*cloning* is the general term for this process.

There are four distinct techniques. They can be used to clone any selection set.

- Shift-Clone
- Snapshot
- Array
- Mirror

Shared Features

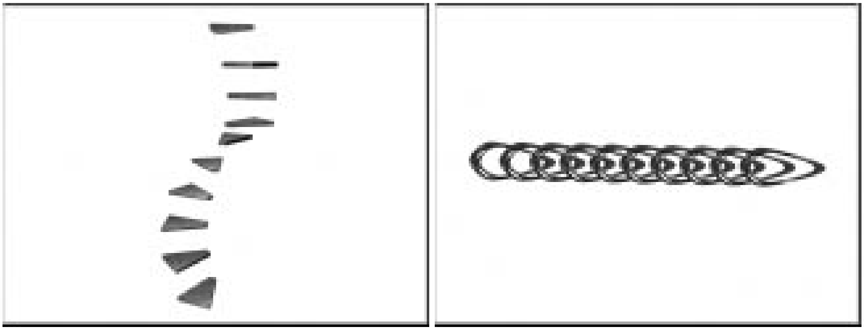
While each technique has distinct uses and advantages in cloning objects, they share some similarities in how they work:

- You can apply a transform when you clone. New objects are moved, rotated, or scaled as they are created.
- The transform is relative to the current coordinate system, axis constraint, and transform center.
- When cloning creates new objects, you have the choice of making them copies, instances, or references.

Each of these items is discussed later in this chapter.

Shift-Clone

In 3DS MAX, you can clone an object as you transform it. The process is referred to as *Shift-Clone*—a technique of holding down the SHIFT


Objects created with array

key while transforming a selected object with the mouse.

Quick and versatile, this technique is probably the one you'll use most often to duplicate objects. Snap settings give you precise results.

How you set the center and axes for the transforms determines the arrangement of the cloned objects. Depending on the settings, you can create both linear and radial arrays.

You need a working knowledge of transform features to take full advantage of Shift-Clone. See chapter 4, "Using Transforms."

Snapshot

Snapshot clones an animated object over time. You can create a single clone on any frame, or space multiple clones along the animation path. The spacing is a uniform time interval; it can also be a uniform distance.

Array

Array creates repeating design elements—for example, the gondolas of a ferris wheel, the treads of a spiral stair, or the battlements along a castle wall.

Array gives you precise control over all three transforms and in all three dimensions, including the ability to scale along one or more axes. It is the combination of transforms and dimensions, coupled with different centers, that gives you so many options with a single tool. A spiral stair, for example, would be a combination of Move and Rotate around a common center. Another array using Move and Rotate might produce the interlocked links of a chain.

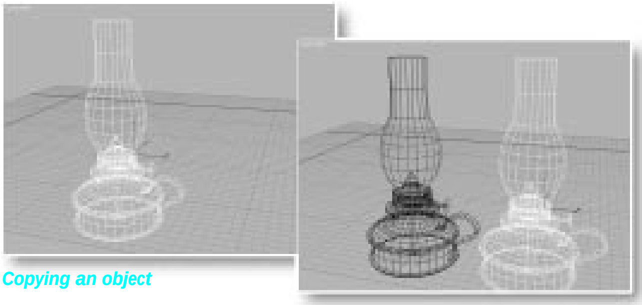
Mirror

Mirror produces a symmetrical copy around any combination of axes. There is also a "No Clone" option that performs the mirror operation without copying. The effect is a flip or move of the object to a new orientation.

Mirror has an interactive dialog. As you change settings, you see the effect in the active viewport—in other words, a preview of what the mirror will look like.

There is also a Mirror modifier that gives you parametric control of the mirror effect.

Overview of Copies, Instances, and References



Copying an object

Every time you duplicate an object in 3D Studio MAX, you choose one of three methods for cloning the original object. For all three methods, the original and clone are identical at the geometry level. Where the methods vary is in the way they handle modifiers (for example, Bend or Twist).

Copy method—Creates a completely separate clone from the original. Modifying one has no effect on the other.

Instance method—Creates a completely interchangeable clone of the original. Modifying an instanced object is the same as modifying the original.

Reference method—Creates a clone dependent on the original. Modifying the original affects the referenced object, while modifying the referenced object has no effect on the original.

Depending on the method used to create them, cloned objects are called *copies*, *instances*, or *references*.

The following discussion focuses on how you might use these methods. For the underlying theory, see “Appendix.”

Copies

Copies are the most familiar kind of clone object. Copied objects are completely independent of the original object, and therefore analogous to a precise physical copy.

Example of Using Copied Objects

If you modeled a basic head shape and wanted to create a group of individual characters, you would probably make a *copy* of the basic head shape each time you started a new character. You could then model an individual nose, mouth, and other features.

Instances

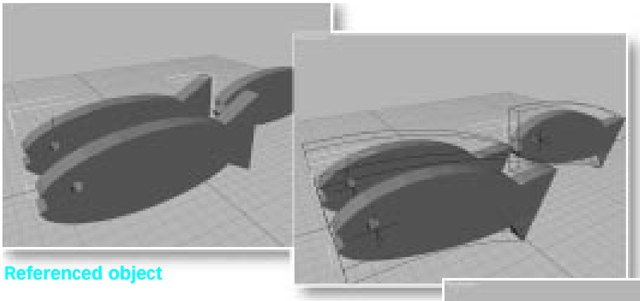
Instances are not only alike in geometry, but also in every other way as well.

When you change one instance by applying a modifier, for example, all the other instances change with it.

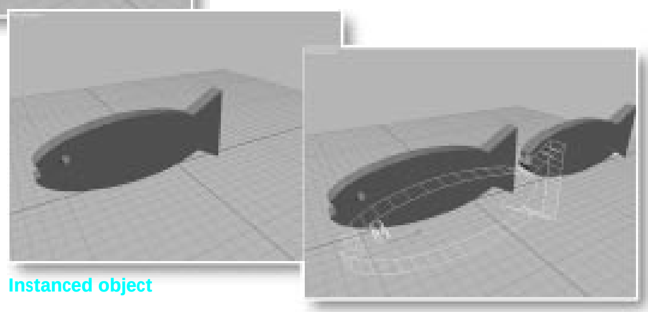
Within the program, instances derive from the same master object. What you’re doing (“behind the scene”) is applying a single modifier to a single master object. In the viewport, what you see as multiple objects are multiple instances of the same definition.

Example of Using Instanced Objects

If you wanted to create a school of swimming fish, you might begin by making many instanced copies of a single fish. You could then animate the swimming motion by applying a ripple modifier to any fish in the school. The



Referenced object



Instanced object

whole school would swim with exactly the same motions.

References

References are like "one-way" instances. Referenced objects are based on the original object, as are instances, and can also have their own unique modifiers.

Any modification made to the original object is passed on to its references, but any modification made to a reference is *not* passed back to the original.

The "one-way" effect is useful, since you can maintain an original that will affect all its references, while the references themselves can take on individual characteristics.

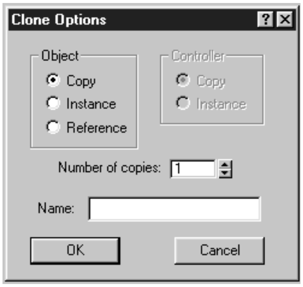
Example of Using Referenced Objects

In the example of modeling heads, you might want to keep a family resemblance in your characters. You could model basic features on the original, then model specifics on each reference.

At some point, if you wanted to see what your characters would look like as "cone-heads," you could apply a Taper modifier to the original head and have all the other characters take on the same feature. You could give the original character a very pointed head, then apply a separate Taper to some referenced characters to reduce the point toward normal.

In the swimming fish example, you might choose to make the school all referenced objects based on a single, original fish. You could still control the swimming motion from the original fish, and also add modifiers to individual fish in the school to vary their behavior.

Using Shift-Clone



Clone Options dialog

Shift-Clone is the primary way to duplicate objects in 3D Studio MAX. You hold down the SHIFT key and drag during any of the standard transform operations: Move, Rotate, or Scale.

To Shift-Clone an object:

1. Click a Move, Rotate, or Scale button.

2. Select a transform coordinate system and constraints. Each transform carries its own settings. To avoid surprises, always click the transform button *first*, and then set your transform coordinate system and constraints.

3. Select the object or set or objects you want to clone. The selection can be a single object, multiple objects, a group, or a sub-object selection.

4. Hold down the SHIFT key and drag the selection to apply the transform.

As you drag, a clone is created and selected; it is now the object being transformed. The original object is no longer selected and is unaffected by the transform.

5. When you release the mouse button, the Clone Options dialog appears. Change settings in this dialog or accept the defaults, and then click OK.

The Clone Options dialog is common to Shift-Clone for any transform you choose.

Object—Set choice of method for copy, instance, or reference 3DS MAX retains your choice as the default during a session.

Number of copies—Set the number of copies. This is the total number of copies plus the original selection. Default=1.

Name—Enter a name for the first clone. 3DS MAX always supplies a default name. It also appends numbers in sequence to the name in this field.

For example, you might model a basic head and name it “Conehead.” If you then make five clones of Conehead, they will be named “Conehead01” through “Conehead05.” You can change any of these names after cloning is complete.

Controller—Set to make either copies or references of the original objects’ transform controllers. For this option to be active, the selection you’re cloning must include two or more linked (child-parent) objects. See chapter 27, “Working with Controllers.”

Animating Shift-Clone

Any Shift-Clone operation can be animated. See “Animating Shift-Rotate and Shift-Scale.”

Cloning Without Transforming

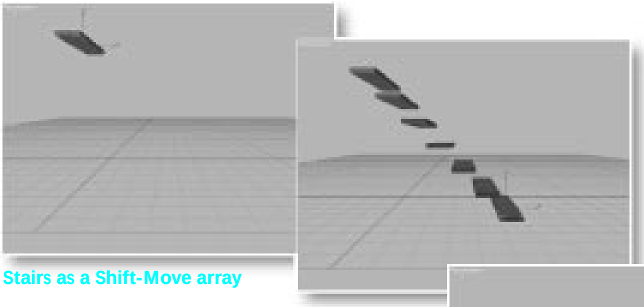
Cloning objects with Shift-Clone requires transforming them at the same time—by moving, rotating, or scaling them. In some cases, you might want to clone an object without transforming it in any way. A menu item gives you this option. Only one clone can be created.

To clone objects without transforming:

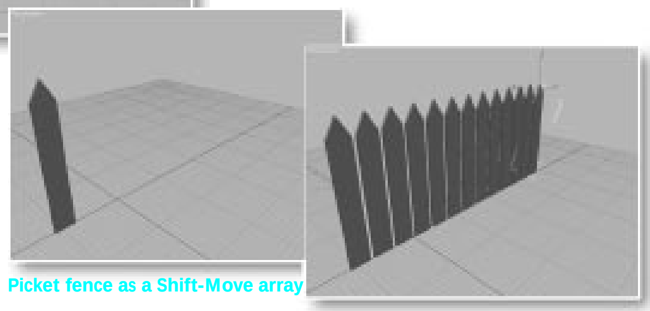
1. Select the object, or set of objects, to clone.
2. Choose Edit > Clone on the menu bar. The Clone Options dialog appears. This is the same dialog used with Shift-Clone except that you can't set a multiple number of clones.
3. Change settings in this dialog or accept the defaults, and click OK.

Note: The cloned object occupies exactly the same space as the original, and is selected when cloning is complete. Use Select By Name to select the original or reselect the clone.

Cloning with Shift-Move



Stairs as a Shift-Move array



Picket fence as a Shift-Move array

Cloning objects while moving them is quick and easy. It produces a linear array of two or more objects.

To clone with Shift-Move:

1. Click the Move button on the toolbar.

2. Choose a coordinate system and axis constraint.

3. Make the selection you want to Shift-Move.

4. Hold down SHIFT and drag to move a clone of the selection away from the original.

5. Choose the number of copies you want to make on the Clone Options dialog, and whether you want them to be copies, instances, or references.

About Shift-Move Arrays

Multiple clones produced by Shift-Move form an equal-spaced linear array with these characteristics:

- The line of the array runs from the center of the original through the centers of the clones.
- The spacing between copies is determined by the distance between the original and the first clone.

By using snaps as you move the selection, you can make exact arrays.

Examples of Shift-Move Arrays

A clear example of the Shift-Move array is the picket fence. From a single picket, you can generate long runs of fencing. The fence can be arrayed along a major axis of the home grid, then grouped, rotated to a particular angle, and moved into position.

You can also make three-dimensional arrays with Shift-Move. The main choice is the combination of axes to allow movement off the construction plane. In the following figure, a box forms the top step of a stairway. The step is then copied and arrayed to create a downward flight.

Cloning with Shift-Rotate

Cloning objects while rotating them produces a variety of effects, depending on how you set up the transformation.

To clone with Shift-Rotate:

1.  Click the Rotate button on the toolbar.
2. Choose a coordinate system, transformation center, and axis constraint.
3. Make the selection you want to clone.
4. Hold down SHIFT and drag to rotate the selection.
5. Choose the number of copies you want to make on the Clone Options dialog, and whether you want them to be copies, instances, or references.

Effects of Transform Settings

Where you locate the transform center determines how 3D Studio MAX positions clones during Shift-Rotate.

- For all settings, the direction of rotation is constrained by the active axis or axes of the viewport's coordinate system.
- Each clone is rotated the same amount as the first clone from the original.

Local Pivot at Center

An object's default pivot point is often located at its center or its base. When you Shift-Rotate around an object's default pivot point, the clones evenly overlap as each one is rotated the same amount. This is true for multiple objects with a local-pivot setting, since each object uses its own local center.

Clones of circular objects, like a sphere or cylinder, are exactly overlaid on the original. You need to move them away from the original to see them.

With angle snap set to divide a circle evenly, you can produce complex symmetrical objects from simple ones. For example, a tetrahedron can be cloned around one axis, then the new set cloned about another axis to produce a faceted star.

Local Pivot at a Distance

When the local pivot is separated from the original, clones create a wheel-like arrangement. Long shapes like petals or blades, cloned with the center near one end, create flowers or propellers.

You can move the local pivot any distance from the object, creating large circular arrays. Since direct animation is limited to the local pivot, this is a key technique in animating circular arrays. See "Animating Shift-Rotate and Shift-Scale."

Selection Center

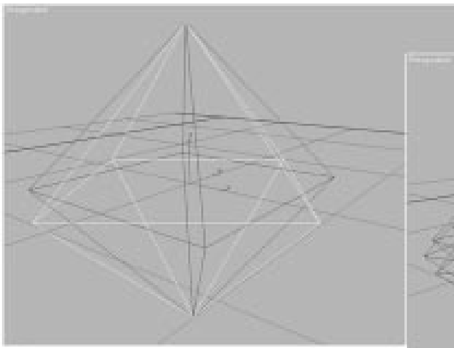
For either single or multiple objects, the selection center is the geometric center of the bounding box enclosing the entire selection. Clones are arrayed around this center, forming wheel-like arrays.

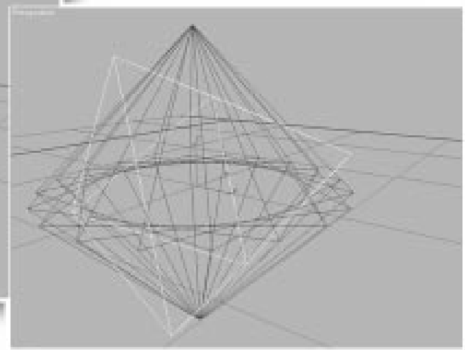
For a single object, this center is usually different from its local center, but effects are similar to those based on a local pivot.

Coordinate Center

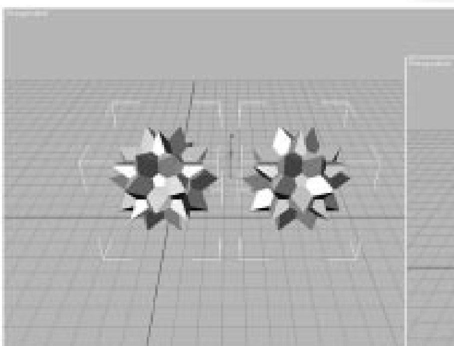
Using the coordinate center, Shift-Rotate can produce circular arrays of any size.

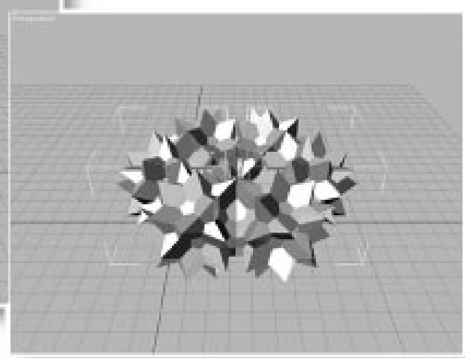
The rotation takes place around the center of the home grid, screen, or whatever coordinate system you choose. While clones can be created this way, the process cannot be directly animated. See "Animating Shift-Rotate and Shift-Scale" for details on overcoming this limitation.



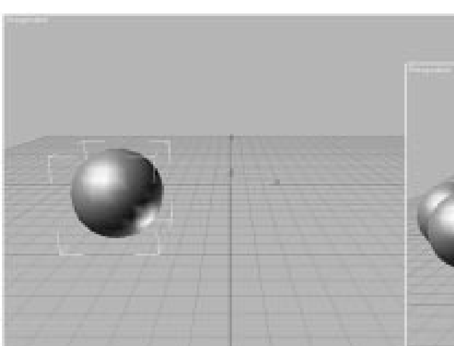


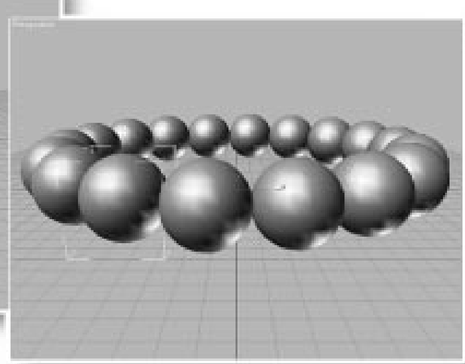
Examples of Shift-Rotate around center pivot





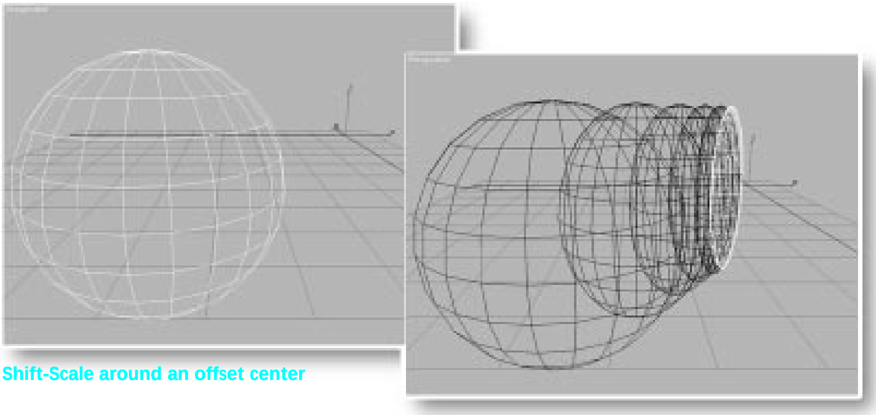
Shift-Rotate around
transform coordinate center



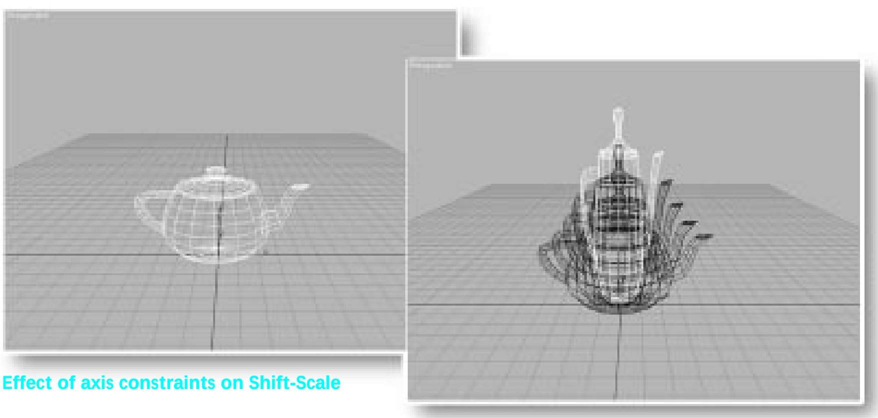


Shift-Rotate around selection center

Cloning with Shift-Scale



Shift-Scale around an offset center



Effect of axis constraints on Shift-Scale

Cloning objects while scaling them can produce a variety of nested objects and arrays, depending on the center you choose.

To clone with Shift-Scale:

1.  Click a Scale button on the toolbar.
2. Choose a coordinate system, axis constraint, and transform center.
3. Make the selection you want to clone.
4. Hold down SHIFT and drag to scale the selection.
5. Use the Copy Options dialog to choose the number of clones you want to make and whether you want them to be copies, instances, or references.

Effects of Transform Settings

Transform settings determine how 3D Studio MAX distributes clones of a selection during Shift-Scale. In all scaling operations, the transform center acts as the center of scaling:

- When clone objects decrease in size, they shrink toward the transform center.
- When clone objects increase in size, they expand away from the transform center.

The distance between cloned objects is scaled like the clones themselves, based on the initial distance from the original to the first clone. The spacing increases or decreases proportionately with respect to the transform center.

Nested Copies

When the selection center is used as the transform center for a single object, scaling occurs symmetrically around that center, producing nested copies.

- As you scale in toward the center, smaller and smaller copies are created.
- In the other direction, the original object is enclosed by increasingly larger copies.

Variations are possible, depending on the type of scale and axis limitations. For example, you can scale a flat box into a progressively stepped pyramid by using Squash and cloning inward on the Z axis.

Offset Centers

For Shift-Scale, any center other than the local pivot has the effect of creating an array of progressively scaled objects. Again, objects scale down in size toward the center, while increasing in size further away. However, this effect is limited by the particular scale option and the axis constraints, as discussed below.

Axis Constraints

Uniform Scale is unaffected by axis constraints. Copies are always arrayed in or out from the center of the current coordinate system.

For Non-Uniform Scale and Squash, scaling occurs only along the axis or axes set with the Axis Constraint buttons.

Animating Shift-Rotate and Shift-Scale

When the Animate button is on, the transform center defaults to local pivot, and the flyoff on the toolbar is grayed out. If you choose one of the other centers and turn on the Animate button, the center returns to the local pivot. This means you can't directly animate Shift-Rotate and Shift-Scale effects such as creating clones in an arc or circular array around a common center.

Using Non-Local Centers

To use a center separate from the object you're cloning, you can do any of the following:

- Use a dummy object.
- Offset the local pivot.
- Change the default.

Using a Dummy Object as Center

In this procedure, you use the axis tripod of the dummy object as the center for rotation or scale.

To use a dummy object as center:

1. Create a dummy object at the center of rotation or scaling.
2.  Link the object you want to clone to the dummy object, which becomes the parent.
3. Select both the dummy and the object, then Shift-Rotate or Shift-Scale the *dummy* when animating.

   For Shift-Rotate, the dummy's center becomes the pivot.

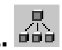
   For Shift-Scale, the dummy and selected object scale together toward a common center.

For details of dummy object use, see chapter 23, "Building Hierarchies."

Offsetting the Local Pivot

In this procedure, you move the object's pivot to the center of rotation or scale. This works much like using a dummy object.

To offset the local pivot:

1.  On the Hierarchy panel, choose Pivot and then Affect Pivot Only.
2. Select and move the local pivot of the original object to another location in your scene.

   Shift-Rotate or Shift-Scale now animate around the offset center. This works with the default setting for local center.

Note: Moving the local pivot can adversely affect linking and inverse kinematics. If this is a possibility, consider changing the default instead of moving the local pivot.

Changing the Default

In this procedure, you change the default of animating only around the local axis.

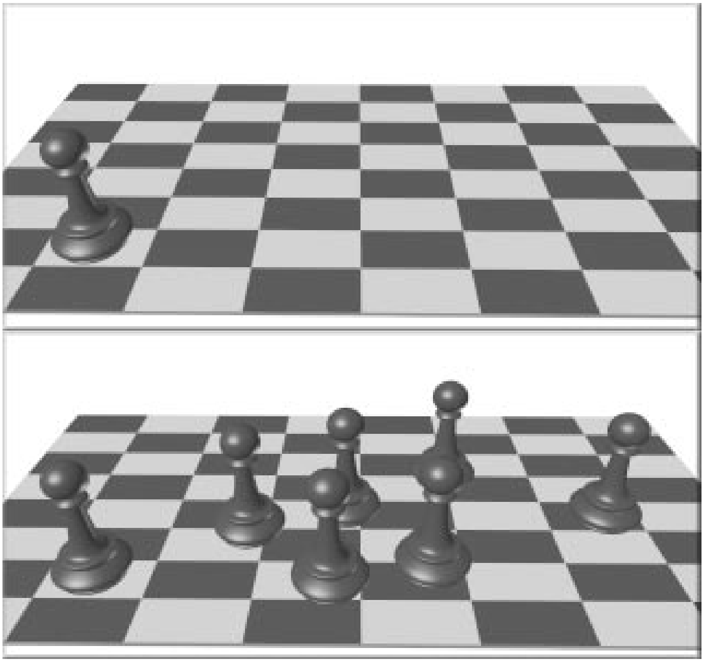
To change the default:

1. Choose File > Preferences and open the Animation panel of the Preference Settings dialog.
2. Clear Local Center During Animate.

   This changes the default and activates all the transform center buttons. You can now animate around either the selection or coordinate center, as well as local pivot.

Note: Changing the default animates the rotation you see in viewports as a rotation plus translation, which might not be the effect you wanted.

Cloning Objects Over Time with Snapshot



The Snapshot tool lets you clone an object along its animation path. You can make a single clone at any frame, or multiple clones spaced over a selected number of frames.

Snapshot spaces the clones equally in *time*. Adjustments in Track View let you space the clones equally along the path instead.

Like other clone techniques, Snapshot creates copies, instances, or references. You can also choose a mesh option.

To clone an object with Snapshot, the object must already be animated. You can use Snapshot from any frame on the path. The Animate button has no effect on Snapshot, since Snapshot creates static clones, not animation. This is the general procedure:

To clone an object with Snapshot:

1. Select an object with an animation path. The animation can result from applying transforms, controllers, or any combination of effects.
2.  Click Snapshot, a flyout from the Array button. The Snapshot dialog appears.
3. Set parameters in the dialog, and click OK.

Snapshot Parameters

These are the Snapshot parameters:

Single—When chosen, a single “snapshot” is made at the current frame.

Range—When chosen, the following parameters become available:

- *From* sets the frame where the first clone appears.
- *To* sets the frame where the last clone appears.
- *Copies* sets the total number of clones in the defined range. By default, the clones are distributed evenly over the time period represented by the range of frames.

Clone method—Sets choice for *copies*, *instances*, *references*, or *mesh*.

- *Mesh* creates objects that are no longer parametric. Mesh objects lose all their modification and animation data, but accurately clone the geometry at each frame.

  If you try to use the mesh method on non-mesh objects like lights and cameras, the result is a copy.

Spacing Clones Evenly on Path

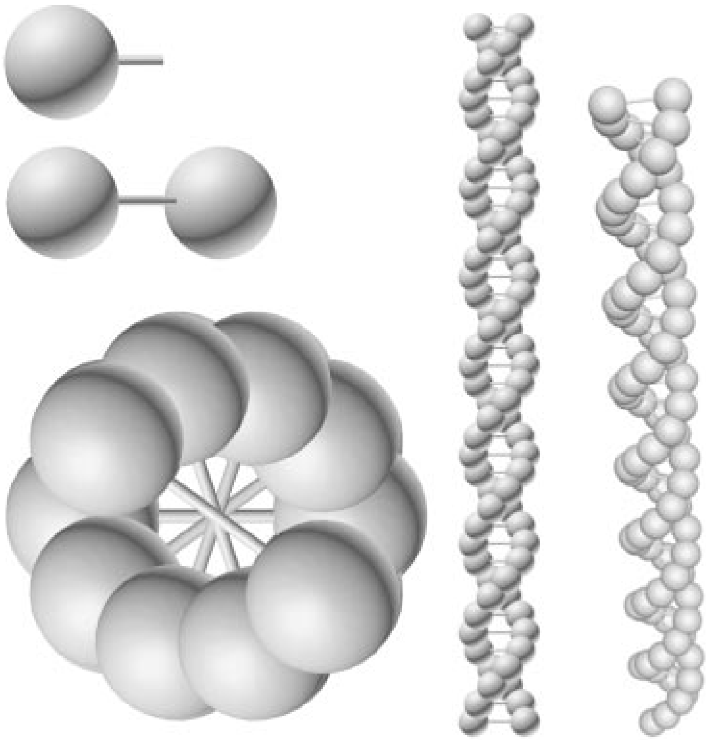
In cases where you want the clones to have equal *space* between them on the animation path, use this procedure.

To space clones evenly on the path:

1. Begin by cloning an object over time.
2.  Open Track View and find the transform tracks for the original object.
3. Select all the transform keys, then right-click one of the selected keys to bring up the Key-Info dialog. Click Advanced to expand the dialog.
4. Click Normalize Time, then select Constant Velocity.

   The clone objects are now spaced evenly along the animation path of the original.

Arraying Objects



Array is a dedicated tool for cloning and precisely transforming and positioning groups of objects in one or more spatial dimensions. For each of the three transforms (move, rotate, and scale), you can specify parameters for individual objects in the array, or for the array as a whole. Many results would be laborious or impossible using Shift-Clone techniques.

Creating an Array

This is the general procedure. For more details, see the next topic, “Using the Array Dialog,” and the topics that follow.

To create an array:

1. Select one or more objects to be in the array.

2. Choose a coordinate system and transform center.

3.  Click the Array button on the full toolbar to display the Array dialog.

4. Set array parameters on this dialog and click OK.

Reuse of Array Settings

Generally you should approach Array creation as an iterative process. The dialog settings are not interactive, so you get feedback only after creating the array. By revising the current settings and repeating the array, you develop a solution that meets your needs.

- After creating an array and checking its result, undo the array using Edit > Undo Create Array or CTRL+Z. This leaves the original selection set in place.

Repeating an Array

When you create an array, object selection moves to the *last* copy in the array. By simply repeating current settings, you create a seamless continuation of the original array.

During a session, 3D Studio MAX maintains all the dialog settings for your current array.

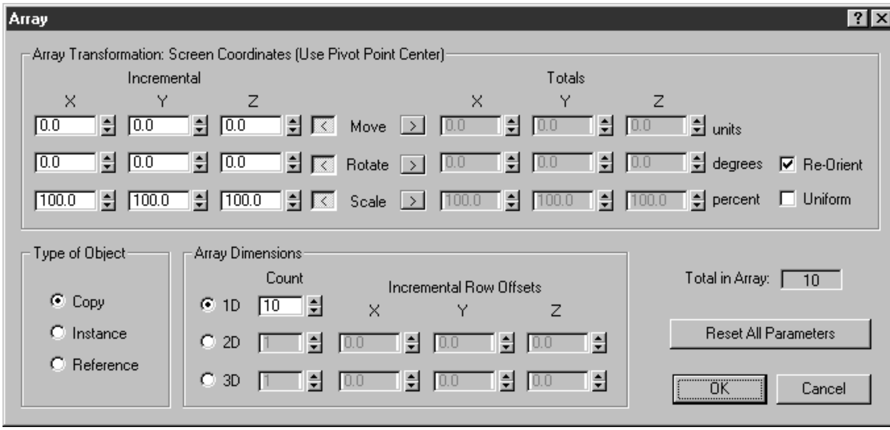
Array settings are saved only during the current session, not with the file. Be sure you've finished an array before you quit 3DS MAX.

General Considerations

When you create an array, keep these points in mind:

- Array is relative to the current viewport settings for coordinate system and transform center.
- Axis constraints do not apply, because Array allows you to specify transforms along all axes.
- Arrays can be fully animated. By changing the default, you can activate all the transform center buttons, allowing direct animation around either the selection or coordinate center, as well as local pivot. To change the default, see “Animating Shift-Rotate and Shift-Scale.”
- To make an array of a hierarchically linked object, select all the objects in the hierarchy before you click Array.

Using the Array Dialog



Array dialog

The Array dialog appears rather complex. However, there are two main control areas where you set the important parameters: Array Transformation and Array Dimensions.

You can set parameters in any order, but in practice it's useful to start with Array Transformation. This creates the basic building block for the larger array, as defined by Array Dimensions.

Later topics discuss specific strategies for using these controls.

Array Transformation

This area lists the active coordinate system and transform center. It's where you set the transforms that define the first row of the array. You decide here on the distance, rotation, or scale of individual elements, and along what axes. You then repeat this row in other dimensions to produce the finished array.

Move, Rotate, and Scale Transforms

You set Move, Rotate, and Scale parameters along any of the three axes of the current coordinate system.

- Move is set in current units. Use a minus value to create the array in the negative direction of the axis.
- Rotate is set in degrees. Use a minus value to create the array in a clockwise direction around the axis.
- Scale is set as a percentage. 100 percent is full-size. Percentages below 100 decrease the size.

Incremental and Totals

For each transform, you have the choice of whether to apply the transforms to each element in the array or to the overall array.

- Click arrows on either side of the transform labels to choose between Incremental or Totals.

Incremental and Totals settings are toggles for each transform. When you set a value on one side, the other side is grayed out. However, the grayed-out value updates to show the equivalent setting.

Incremental—Parameters set on this side apply to individual objects in the array. Here are examples:

- An Incremental Move X setting of 25 specifies a spacing of 25 units on the X axis between centers of arrayed objects.
- An Incremental Rotate Z setting of 30 specifies a progressive rotation of 30 degrees on the Z axis for each object in the array. In the finished array, each object is rotated 30 degrees farther than the one before it.

Totals—Parameters set on this side apply to the overall distance, number or degrees, or percentage scale in the array. Here are examples:

- A Totals Move X setting of 25 specifies a total distance of 25 units on the X axis between the centers of the first and last arrayed objects.
- A Totals Rotate Z setting of 30 specifies a combined rotation of 30 degrees on the Z axis divided equally among every object in the array.

Array Dimensions

The other main control area is Array Dimensions. These controls determine the number of dimensions used in the array and the spacing between the dimensions.

Count—The number of objects, rows, or layers in each dimension.

1D—One-dimensional arrays form a single line of objects in 3D space, like a line of columns.

1D Count=number of *objects* in a row. Spacing for these objects is defined in the Array Transformation area.

2D—Two-dimensional arrays form a single layer of objects in 3D space, like the rows of squares on a chess board.

2D Count=number of *rows* in the array.

3D—Three-dimensional arrays form multiple layers of objects in 3D space, like boxes stacked neatly on a palette in a warehouse.

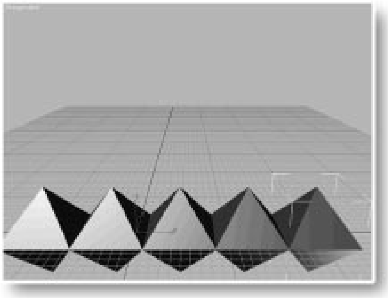
3D Count=number of *layers* in the array.

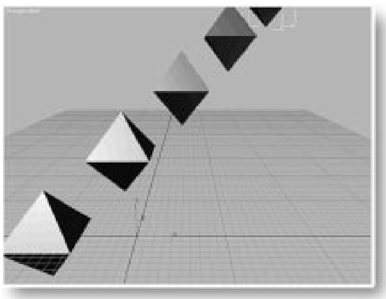
Incremental Row Offsets

These parameters become available when you choose a 2D or 3D array. These are distances along any of the three axes of the current coordinate system.

- If you set a Count value for 2D or 3D, but no row offsets, the array is created with overlapping objects. You need to specify at least one offset distance to prevent this.
- If some objects appear missing in the array, use Select By Name on the toolbar to see the full listing of objects in your scene.

Creating Linear Arrays





Linear array and diagonal array

A linear array is a series of clones along one or more axes. A linear array can be anything from a line of trees or cars to a stairway, a picket fence, or a length of chain. Any design requiring repeated objects or shapes is a candidate for a linear array.

See the previous topic, "Using the Array Dialog," for an explanation of interface terms used here. See "Creating Arrays" for the basic steps in making an array.

Creating Simple Linear Arrays

The simplest 2D linear array is based on moving a single object along a single axis. These are the basic choices to make on the Array dialog.

Make these choices in the Array Transformation area:

- Use *Incremental* Move settings where you know the spacing you want between objects.
- Use *Totals* Move settings when you know the overall space or volume you want the array to occupy.
- For either kind of Move, enter a value for *one* axis. Leave the other transforms at their defaults.

Makes these choices in the Array Dimension area:

- Set 1D (2D and 3D are toggled off automatically when you set 1D).
- Enter a Count value for the number of objects in the array. The field for Total In Array updates to show you the current total of objects in the array you are designing.
- Click OK to create a linear array along the chosen axis, with the number of objects specified by Count.

2D and 3D Linear Arrays

Arrays in 2D and 3D have the same Array Transformation setup as 1D.

- Set 2D or 3D and enter a Count value.

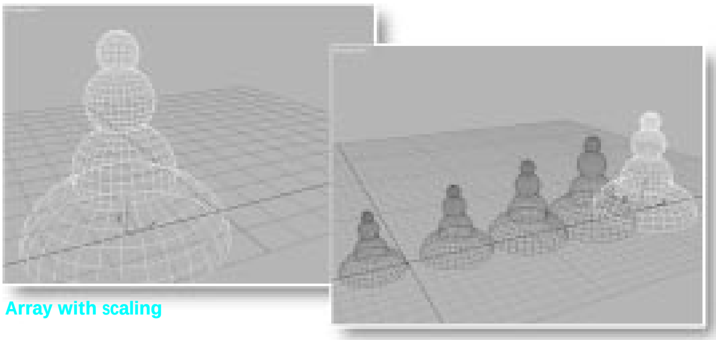
  If you set 3D, 2D is automatically active. Both default to 1, which has the same effect as 1D. Set 2D and 3D Count above 1 to produce a more complex array.

- Set at least one Incremental Row Offset for 2D and 3D. Otherwise, there will no separation between the 1D row and the new clones.

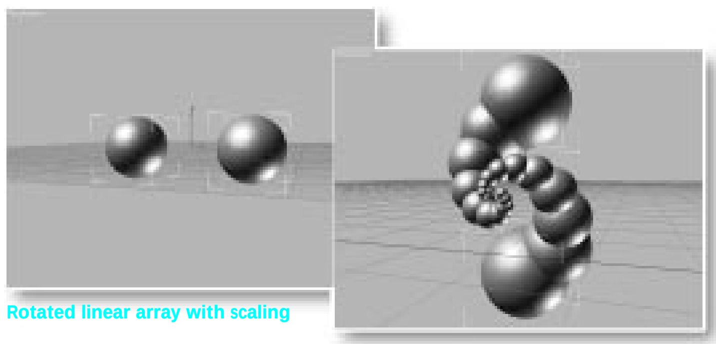
A wide variety of linear arrays are possible. Experiment with moving along all three axes and varying the row offsets in 2D and 3D.

Using Rotation in Linear Arrays

When you add rotation to a linear array, the choice of transform center becomes important.



Array with scaling



Rotated linear array with scaling

Examples shown here are 1D arrays. Using 2D or 3D arrays would add more components to the same pattern.

In the illustration, two spheres are arrayed along a single axis and rotated around their common center. This linear array produces a DNA-like spiral.

Using Scale in Linear Arrays

When you add a scale factor, each copy is scaled from the previous copy. Objects in the array become progressively smaller or larger. In the illustration, the same rotated array is used, but this time is scaled uniformly larger. The result can have a fractal-like effect.

Scale and Movement in Nested Arrays

Using only scale settings and the local pivot of an object produces nested arrays, like Russian dolls, just as it does when you Shift-Scale from the local pivot. However, with the Array tool, you can add movement as well. This means you can create increasingly larger or smaller copies and array them at the same time.

Using Uniform Scaling

By default, all axes are available for scaling.

If you set Uniform, only the Incremental Scale X field is active; the Y and Z fields are grayed out. The X value is applied as a uniform scale on all axes of the arrayed objects.

Creating Circular and Spiral Arrays

Circular and spiral arrays typically involve some combination of moving, scaling, and rotating copies along one or two axes and around a common center. The effects can vary from the uniform radial arrangement of bolts on a wheel hub to the complex geometry of a spiral staircase. Many circular patterns can be modeled with these techniques.

See “Using the Array Dialog,” for an explanation of interface terms used here. See “Creating Arrays” for the basic steps in making an array.

Using a Common Center

Circular and spiral arrays both require a common center for the arrayed objects. This can be the world center, the center of a custom grid object, or the center of the object group itself. You can also move the pivot point of an individual object and use that as the common center.

Circular Arrays

Circular arrays are similar to linear arrays, but based on rotation around a common center rather than movement along an axis. The following procedure makes a circle of objects on the XY plane of the home grid with the Z axis as the center.

To create a circular array:

1. On the toolbar, select a transform center to become the center of the array. In this case, select Use Transform Coordinate Center so the center of the grid becomes the array center.

2. Select an object and position it at some distance from the center of the array. This distance is the radius of the finished circle.

3. On the Array dialog, enter 360 in the Totals Rotate Z field. This is the total rotation for the array, a complete circle. Enter fewer degrees to create a partial circle.

4. Set 1D and enter a Count value. This can be any number. The software arrays that number of clones within the total rotational angle you specify.

Spiral Arrays

The simplest spiral arrays are rotated circular arrays with a movement along the central axis. The same circle is formed, but now the circle rises upward.

- If Z is the central axis, enter a value for Incremental Move Z. Each clone is then moved upward this amount as the circle is formed.

Rotation in Spiral Arrays

In spiral arrays, the direction of rotation determines the direction of the spiral—which way it winds up or down.

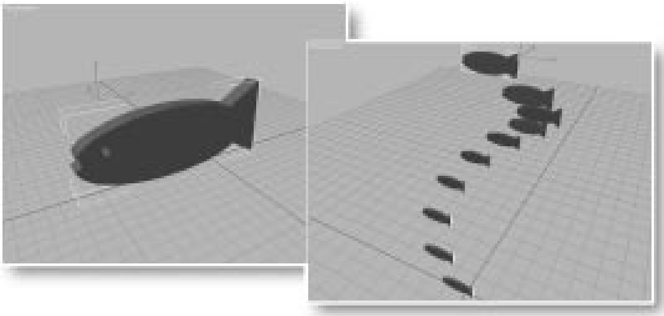
- Enter a positive rotation for a *counterclockwise* spiral.

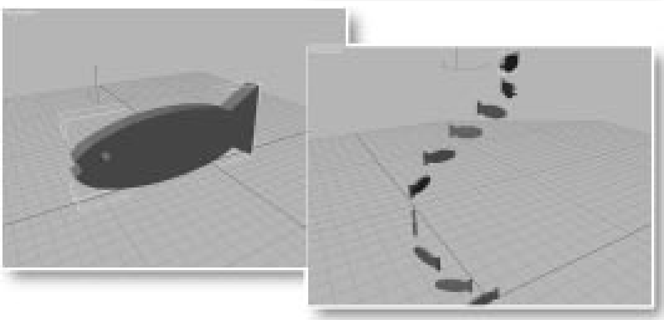
- Enter a negative rotation for a *clockwise* spiral.

Reorienting an Array

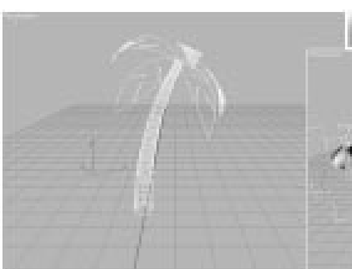
By default, each object, when copied into the array, rotates around its own center to follow the main rotation around the common center. This is controlled by the Re-Orient option.

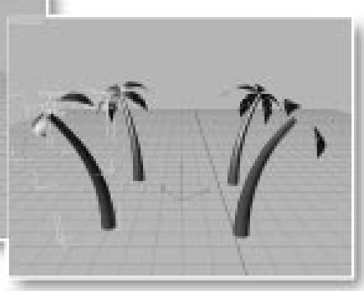
Clear Re-Orient so objects maintain their original orientation while being rotated. In effect, objects remain “facing the same direction” as the original object.



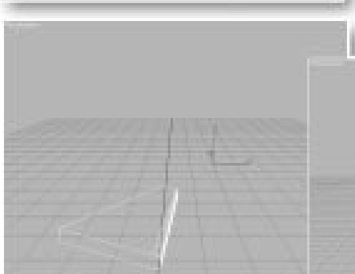


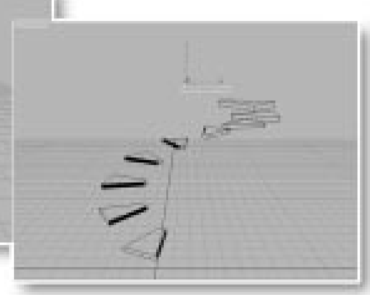
Rotation with and without Re-Orient





Circular array





Spiral array

Mirroring Objects

The Mirror tool in 3D Studio MAX is a dialog that either creates a mirrored clone of a selected object, or mirrors the orientation of the object without creating a clone. You can preview the effects of settings before committing to the operation.

This is the general procedure for mirroring an object. Begin by selecting the object.

-  Click the Mirror button in the full toolbar or choose Edit > Mirror from the menu bar. This displays the Mirror dialog. The title bar indicates the coordinate system currently in use.

Clone Selection

By default, the No Clone option is set on the Mirror dialog. The effect is to flip the object to a new position instead of copying it.

- To see a true mirror effect, set Copy, Instance, or Reference instead.

Mirror Axis

Sets one of six possible axes for the mirror operation: X, Y, Z, XY, YZ, ZX.

- As you set parameters for Mirror Axis, the active viewport changes to show the effect of each parameter as you set it.

- When you click OK, 3DS MAX creates the choice of mirror that you see previewed.

Offset

This option moves the mirrored object on the mirror axis.

- A positive offset moves in a positive direction along this axis; a negative offset moves in the other direction.

- If a combination axis like XY is set, movement is equal along both X and Y.

Mirror IK Limits

When selected, this option allows Inverse Kinematic (IK) constraints to be mirrored about a single axis along with the object's geometry. Mirrored IK hierarchies work properly when this option is selected.

Mirrored Arrays

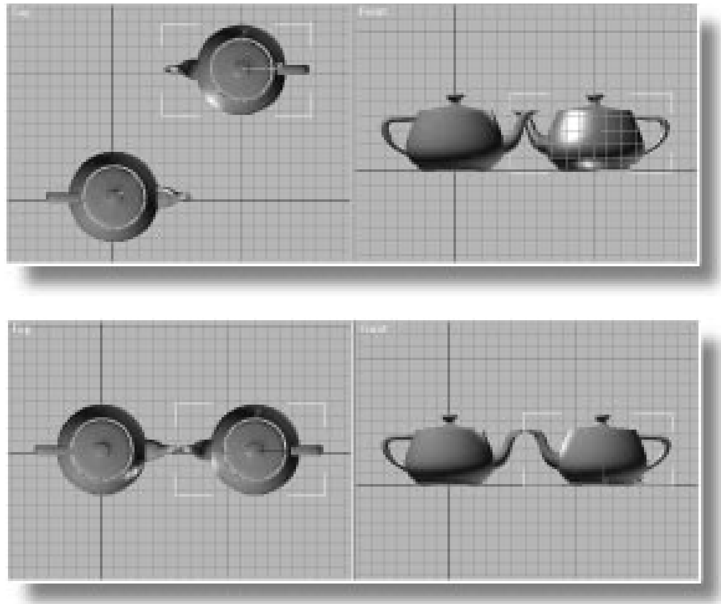
You can combine the Mirror and Array tools. An entire array can be mirrored, or you can set up mirrored objects before creating an array

Animating Mirror

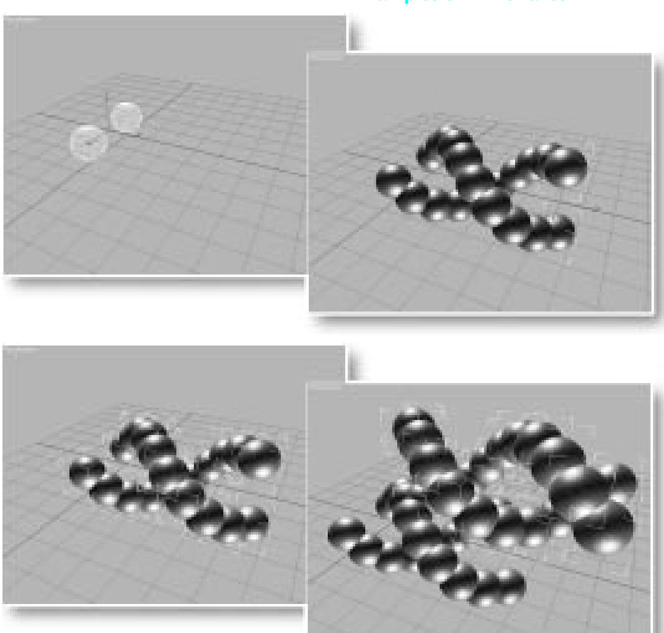
When you animate Mirror, you see the transition occur as the mirrored object moves into place. For example, a cylinder mirrored to the other side of an axis appears to flatten and reshape itself. This is not visible unless animated.

Mirror Modifier

A Mirror modifier provides a parametric method of mirroring an object or sub-object selection within the modifier stack. You can apply the Mirror modifier to any type of geometry. You can animate the mirror effect by animating the modifier's gizmo.



Examples of mirror axes



Mirrored array

Chapter

8

Creating Spline Shapes

Shapes are 2D and 3D objects that you use as components of other objects. The shapes in 3D Studio MAX are made from splines and NURBS curves. This chapter discusses the use of spline shapes. See chapter 10, “Creating and Editing NURBS Models” for information about NURBS curves and surfaces.

You use spline shapes to do the following:

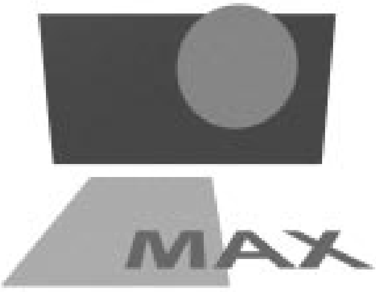
- Generate planar and thin 3D surfaces.
- Define loft components such as paths, shapes, and fit curves.
- Generate surfaces of revolution.
- Generate extrusions.
- Define motion paths.

You can quickly create spline shapes using mouse or keyboard entry, and those shapes can be combined to form compound shapes.

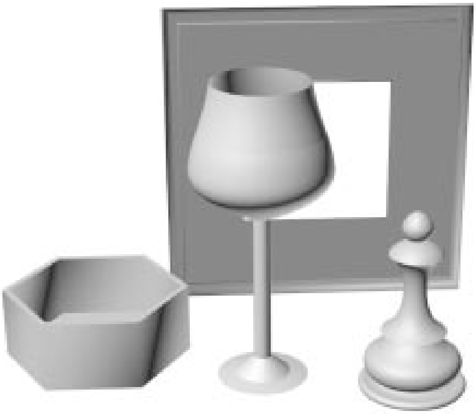
See the Online Reference for details about shape creation parameters and shape editing.

Uses for Shapes

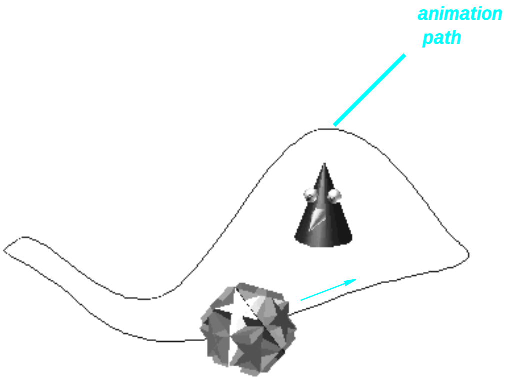
Lofted shapes



Shapes as planar objects



Lathed shapes



Shape used as a motion path

Extruded shapes

The primary use for shape objects is as building blocks to create or control something else. In that role shapes have many uses.

Shapes as Planar Objects

You can use shapes as 2D cutouts or planar objects. Examples include ground planes, text to form signs, and cutout billboards. You create planar objects by collapsing a shape to an Editable Mesh object, or by applying a mesh modifier such as Normal.

Extruded and Lathed Shapes

You can also create 3D objects by applying modifiers to a shape. Two of these modifiers are Extrude and Lathe. Extrude creates a 3D object by adding height to a shape. Lathe creates a 3D object by rotating a shape around an axis.

Lofting Shapes

Lofts are created by combining two or more shapes in special ways. Shapes form the lofting path, loft cross sections, and loft fit curves. See chapter 9, “Creating Lofts.”

Shapes as Animation Paths

Shapes can be used to define the position of an animated object. You create a shape and use it to define a path that some other object follows.

- A Path controller reads a shape to control object motion. See chapter 27, “Working with Controllers.”
- A shape can be converted into position keys using the Trajectory/Convert From function on the Motion panel. See chapter 26, “Function Curve and Trajectory Editing.”

Creating Shapes

Click Shapes on the Create panel to gain access to the shape creation tools. 3D Studio MAX spline shapes are under Splines in the category list.

Creating Single Spline and Multiple Spline Shapes

A shape can contain a single spline or it can be a *compound shape* containing multiple splines. Use Start New Shape on the Object Type rollout to control how many splines are in a shape. The checkbox next to Start New Shape determines when new shapes are created.

- When selected, you create a new shape object for every spline you create.
- When clear, splines are added to the current shape until you click Start New Shape.

To manually control creating new shapes:

1. Clear the Start New Shape checkbox, and click Start New Shape.

2. Create splines. Each spline is added to the compound shape. You can tell you are creating a compound shape because all the splines remain selected.

3. Click Start New Shape to complete the current shape and start another.

- You can change the parameters of a single spline shape after the shape is created.
- You cannot change the parameters of a compound shape. For example, create a compound shape by creating a circle and then adding an arc. Once you create the arc, you cannot change the circle parameters.
- You can add splines to a shape by selecting the shape and then creating splines with the Start New Shape checkbox cleared.

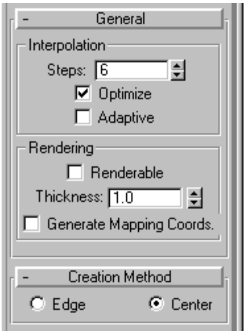
Creating Splines with Keyboard Entry

You can create most splines solely using keyboard entry. The process is generally the same for all splines and the parameters are found under the Keyboard Entry rollout. Keyboard entry for each spline varies primarily in the number of optional parameters.

The Keyboard Entry rollout contains three fields for the X, Y, and Z coordinates of the initial creation point plus a variable number of parameters used to complete the spline. Enter values in each field and click Create to create the spline.

Lines use a slightly different keyboard entry method. See the Online Reference for details about creating lines using the keyboard.

Setting Common Spline Controls



Splines share common parameters that set your creation method and how 3D Studio MAX interpolates splines.

Choosing a Spline Creation Method

On the Creation Methods rollout, you set whether to define splines by their center point or by their edge diagonal.

Center—Press to define the center of the shape and drag a radius or corner point.

Edge —Press to define a point on the side or corner of the shape and drag a diameter or opposite corner.

Line and Arc have unique Creation Methods that are discussed later in this chapter.

Setting Spline Interpolation

The Interpolation parameters on the General rollout set how spline curves are divided to approximate a true curve. The divisions between each vertex on the spline are called *steps*. The more steps used the smoother the curve appears.

Spline steps can be either adaptive or manually specified. The method used is set by the state of the Adaptive check box.

- When Adaptive is selected, spline steps are automatically calculated to produce a smooth curve. Straight segments have zero steps.
- When Adaptive is clear, you manually control interpolation by setting the Steps and Optimize parameters. If Optimize is selected straight segments have zero steps.

The main use for manual interpolation is to create splines for morphing or other operations where you must have exact control over the number of vertices created.

Renderable Shapes

The Rendering parameters on the General rollout set whether a shape appears in a final rendering.

- When Renderable is selected, the shape is rendered using a 12-sided circle as a cross section. The Thickness parameter sets the diameter of the circle.
- When Renderable is clear, splines appear in viewports but are invisible in final renderings.

Creating Lines and Arcs

Lines and Arcs use a slightly different creation method than most other splines.

Creating Lines

Use Line to create a free-form spline made of multiple segments.

You choose Line creation methods to control the type of vertex created when clicking or dragging vertices. These are vertex types:

Corner—Produces a sharp point. The spline is linear to either side of the vertex.

Smooth—Produces a smooth, nonadjustable curve through the vertex. The amount of curvature is set by the spacing of the vertices.

Bezier—Produces a smooth, adjustable curve through the vertex. The amount of curvature and direction of the curve is set by dragging the mouse at each vertex.

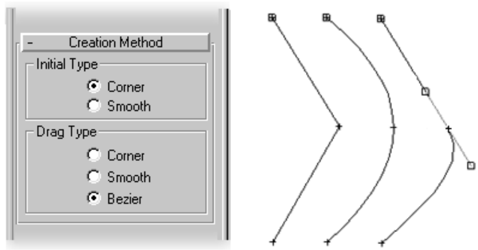
Your choice for Initial Type sets the vertex you get when you click. Your choice for Drag Type sets the vertex you get when you drag.

Creating Arcs

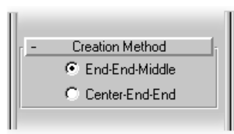
Use Arc to create open and closed circular arcs made of four vertices. Like lines, arcs have their own unique options under Creation Method.

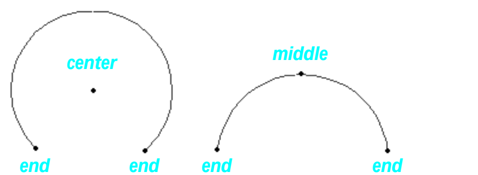
The two arc creation methods are End-End-Middle and Center-End-End. These two methods only affect how you specify an arc. All arcs are created with four vertices along their edge and their pivot point at their center.

Once you have created an arc, you can change its radius and endpoint angles. You can also turn the arc into a closed pie slice.

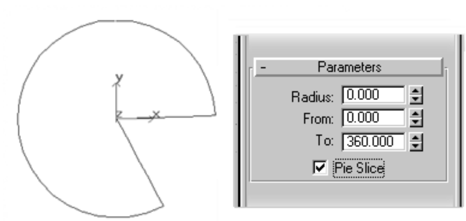


Results of different Line creation method choices



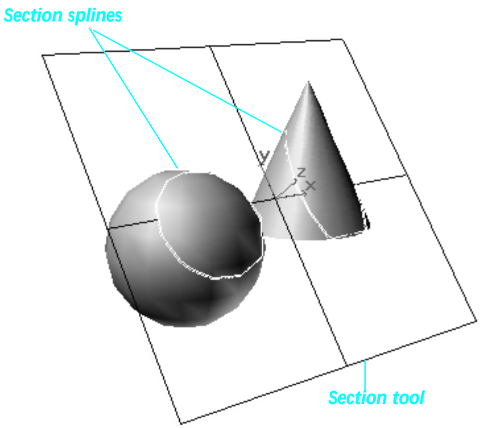


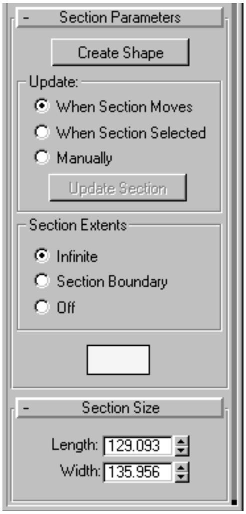
Creating arcs with Center-End-End and End-End-Middle



Result of setting Pie Slice

Creating a Section Shape





The Section tool on the Object Type rollout is really a tool that generates shapes based on a cross-sectional slice through a mesh object. Using the Section tool you can create multiple shapes.

Creating a Section Shape

The Section tool has no Creation Methods rollout. After you click Section on the Object Type rollout, you drag two corners of a rectangular cutting plane.

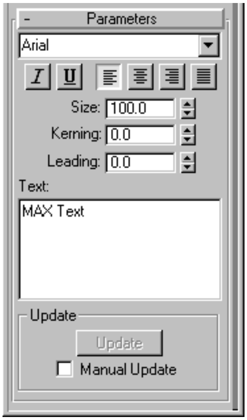
The cutting plane is not a shape. Think of the Section tool as a gizmo that you use to create shapes.

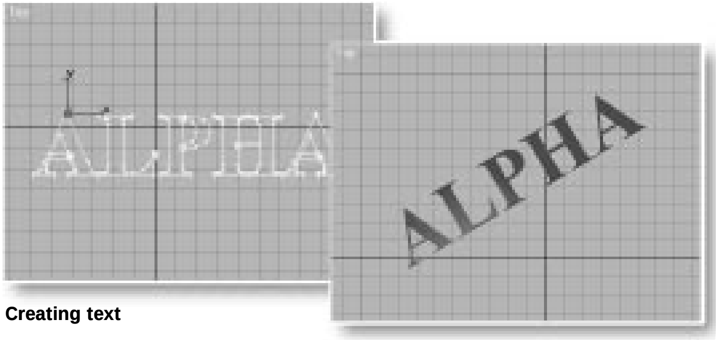
To create shapes using a Section tool:

1. Create a Section tool, then go to the Modify panel.
2. Move and rotate the Section tool to intersect mesh objects in your scene. When the Section tool intersects a mesh object, you see cross sections drawn on the Section plane.
3. Click Create Shape to create a spline shape from the cross section. The created spline is a separate shape object independent of the Section section that created it.
4. Repeat steps 2 and 3.

See the online documentation for details about Section options.

Creating Text





Creating text

Modifying text

Use Text to create editable text splines using Type 1 PostScript® fonts or any font installed in your Windows NT system.

To create text:

1. Click Text on the Object Type rollout.

2. Enter text in the Text edit box.

3. Do the following to define an insertion point:

- Press the mouse button to see the text in the scene.
- Drag the text into position and release.

Text Parameters

Once you have created text, you can change text properties such as font, style, justification, and kerning. You can also edit the text in the Text edit box.

Using Text Shapes

Text shapes maintain editable text as a parameter so you can go back and change the Text at any time. If the font used by your Text gets deleted form the system, 3DS MAX still properly displays the Text shape. However, you cannot edit the text string in the Text edit box until you choose a new available font.

The Text in your scene is just a shape where each letter and possibly pieces of each letter are individual splines. You can apply modifiers like Extrude, Bend, and Edit Shape to Text shapes just like any other shape.

Creating Spline Primitives

The remaining shape primitive use the same basic procedure of dragging a center and radius (or edge to edge) and then setting parameters. The basic set of spline shape primitives included in 3D Studio MAX are listed below.

Circle—Creates a four vertex circle. You can create a circle with any number of vertices by creating an Ngon.

Donut—Creates a compound shape of two concentric circles.

Helix—Creates a flat spiral or 3D helix.

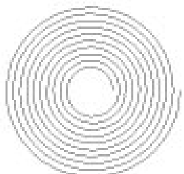
Ngon—Creates a multiple-sided regular polygon with an option to round corners. Select the Circular option to convert it to a multiple vertex circle.

Star—Creates a multiple-pointed star with an option to round points.

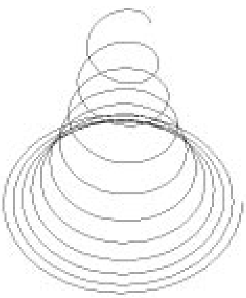
Rectangle—Creates a rectangle or square with an option to round corners.

Ellipse—Creates a four vertex ellipse.

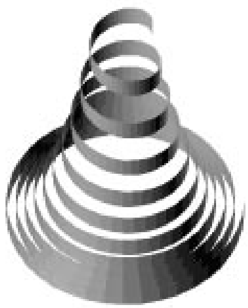
Overview of Editing Shapes



helix in shape view



Helix in perspective view



Extruded helix

Original arc

Bend applied to arc

arc extruded

arc extruded with
Pie Slice checked

You can edit shapes using many techniques:

- Collapse the shape to an Editable Spline.
- Apply an Edit Spline modifier.
- Apply deformation modifiers.
- Apply surface generation modifiers.

Collapsing Shapes

Collapse a shape to an Editable Spline object when you no longer need access to the original shape parameters. Editable Spline parameters include tools for editing and animating shape sub-object splines, segments, and vertices.

Editing Shapes

Apply an Edit Spline modifier to select sub-object splines, segments, and vertices within a shape object and pass them up the modifier stack to other modifiers.

Edit Spline also includes many of the editing tools as the Editable Spline object but does not have the ability to animate the edits. Use Edit Spline only when you want to edit spline sub-objects while retaining the original spline parameters.

Converting Shapes to Surface Geometry

Use modifiers such as Extrude and Lathe to convert a 2D or 3D shape into a 3D surfaced object.

You can also convert shapes to 3D geometry by creating Loft objects from the shapes. See chapter 9, “Creating Lofts.”

Applying Shape Modifiers

Applying deformation modifiers—such as Bend or Twist—to a shape works the same as applying modifiers to a mesh object. See chapter 6, “Basics of Creating and Modifying Objects.”

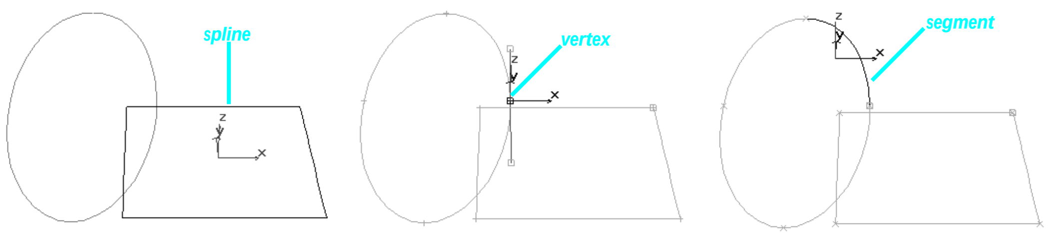
Using the Shape View

You can use the Shape view type to assist you during shape editing. The Shape view is always aligned with the local XY plane of a selected shape. You might find Shape views useful when adding splines to a shape as described above.

To assign Shape view to a viewport:

- Right-click a viewport label, then choose Views > Shape from the menu.

Collapsing Shapes to Editable Splines



Shape with various subobject selections

You convert a parametric shape to an Editable Spline to perform the following tasks:

- Combine multiple shapes into a single shape.
- Extract splines from a shape to form independent shapes.
- Change, transform, and animate shape sub-object splines, segments, and vertices.

Collapsing a Spline Shape

You can collapse a shape by selecting the shape and clicking Edit Stack on the Modify panel.

- If you have not applied any modifiers to the shape, you can choose to collapse the shape to an Editable Mesh or an Editable Spline object.
- If you have applied modifiers you see the Edit Modifier Stack dialog where you click Collapse All.

Selecting Shape Sub-Objects

You can work with parts of shapes and splines using shape sub-object selection of the Editable Spline object.

- *Vertices* define points and curve tangents.
- *Segments* connect vertices.
- *Splines* are a combination of one or more connected segments.

To select shape sub-objects:

1. Turn Sub-Object on and choose one of the sub-object selection levels.
2. Click a selection or transform tool and select sub-objects using standard click or region-selection techniques.

Transforming Shape Sub-Objects

You can use the transform buttons on the toolbar to move, rotate, and scale shape sub-object selections. You can animate sub-object transformations by turning on the Animate button.

Because sub-object selections can be complex you might consider using one of the following techniques to prevent clearing the sub-object selection by accident:

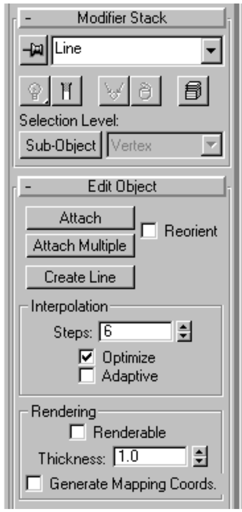
- Use Lock Selection.
- Name the sub-object selection.

See chapter 3, “Selecting Objects”, for details about locking selections and naming selections.

Cloning Sub-Object Selections

Hold down the SHIFT key to clone shape segment and spline sub-objects while transforming them. Shape vertices cannot be cloned.

Editing Shapes at the Object Level



You can perform some functions at the base object level of an Editable Spline object.

Attaching Shapes

Click the Attach or Attach Multiple at the object level of an Editable Spline to combine shape objects. This is the general method to add more splines to a shape.

You can also choose the Reorient option so attached shapes are moved and rotated to match the local coordinate system of the current shape.

Attached shapes lose their identity as individual shapes, with the following results:

- The attached shape loses all access to its creation parameters. For example, once you attach a circle to a square you cannot go back and change the radius parameter of the circle.
- The modifier stack of the attached shape is collapsed. Any edits, modifiers, and animation applied to the attached shape are frozen at the current frame.

Adding Lines to a Shape

Click Create Line at the Editable Spline object level to quickly add a new spline to the selected shape. This command behaves similarly to the Line command on the Create panel.

Unlike the Line command in the Create panel, you cannot set what type of vertex is created by clicking or dragging. With vertex creation, these actions are restricted, as follows:

- Click to place a *corner* vertex.
- Drag to place a *Bezier* vertex.

Editing Spline Vertices

Spline vertices define the ends and curvature properties of spline segments. Choose Vertex as the Editable Spline sub-object level to display the Edit Vertex rollout.

Adding Vertices to a Spline

You can use Insert and Refine to add vertices to a spline. The main difference is how the curvature and location of the new vertex is defined.

- Click Insert to add new vertices to a segment between two connected vertices. You have complete control over vertex location and curvature.

- Click Refine to add more detail to a segment without changing its curvature. This command is useful for smoothing out curves that appear rough due to low step settings. Refine is also available in the Edit Segment rollout.

Removing Vertices

You can remove vertices from splines by deleting or welding vertices.

- You can delete all but the last two vertices from a spline by selecting vertices and clicking Delete.

Welding Vertices

- Use Weld to combine two or more selected vertices into a single vertex. You use the Weld Threshold field to specify how close the vertices must be to each other in order to be welded.

Breaking Vertices

Select a vertex and click Break to split a spline at a vertex location. The result is two unconnected vertices at the same location.

Connecting Vertices

You can connect the end points of open splines in the same shape by clicking Connect and then dragging from one vertex to another.

Setting the First Vertex on a Spline

The first vertex of a spline is indicated as a vertex with a small box around it. You select a vertex and click Make First to specify which vertex is considered the first vertex.

The first vertex on a spline has special significance. The following table defines how the first vertex is used.

Shape Use	First Vertex Meaning
Loft Path	Start of the path. Level 0. See chapter 9, “Creating Lofts”.
Loft Shape	Initial skin alignment. See chapter 9, “Creating Lofts”.
Path Controller	Start of the motion path. 0% location on the path. See chapter 27, “Working with Controllers”.
Trajectory	First position key. See chapter 26, “Function Curve and Trajectory Editing”.

The following rules apply to setting a first vertex:

- The first vertex of an open spline must be one of the end vertices.
- Any vertex can be the first vertex of a closed spline.
- If more than one vertex is specified on the same spline, the first vertex is not changed.
- If multiple vertices are selected on multiple splines, only vertices meeting the above rules are assigned as first vertices. The other selected vertices are ignored.
- If no vertices are valid first vertices, an alert appears: “No valid vertices selected.”

Changing Vertex Properties

Each vertex on a spline has a specified curve property. The property sets the behavior and appearance of the spline at that vertex.

Smooth—Non-adjustable vertices that create smooth continuous curves. The curvature at a smooth vertex is determined by the spacing of adjacent vertices.

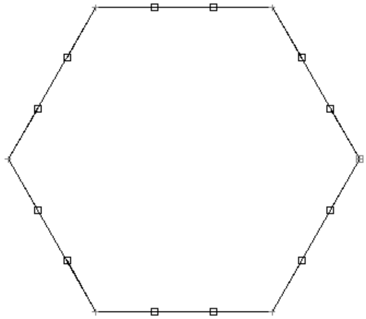
Corner—Non-adjustable vertices that create sharp corners.

Bezier—Adjustable vertex with locked continuous tangent handles that create a smooth curve. The curvature at the vertex is set by the direction and magnitude of the tangent handles.

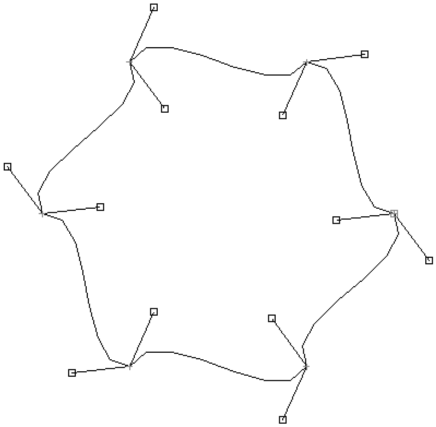
Bezier Corner—Adjustable vertex with discontinuous tangent handles that create a sharp corner. The curvature of the segment as it leaves the corner is set by the direction and magnitude of the tangent handles.

To set a vertex property:

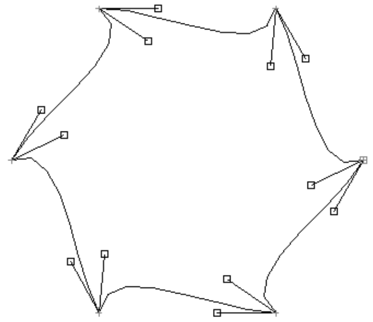
- Right-click any vertex in a selection, then choose a property from the shortcut menu.



Original N-gon



All vertices locked



Like vertices locked

Transforming Bezier Vertex Handles

Vertices with Bezier or Bezier Corner properties display tangent handles when you select the vertex. You can drag the tangent handles of these vertices to change the direction and amount of curvature at a vertex.

- Drag either tangent handle of a Bezier vertex to move both handles, maintaining a continuous curve.
- Drag a tangent handle of a Bezier Corner vertex to move only the selected handle. The other handle is unaffected, resulting in a kink, or corner, in the spline.

•Press SHIFT while dragging a Bezier vertex tangent handle to break the continuity and convert the vertex to a Bezier Corner vertex.

•Move a handle to adjust both the direction and amount of curvature at a vertex.

- Rotate a handle about its local center to change only the direction at the vertex and leave the amount of curvature unchanged.
- Uniformly scale a handle about its local center to change only the amount of curvature at the vertex and leave direction unchanged.

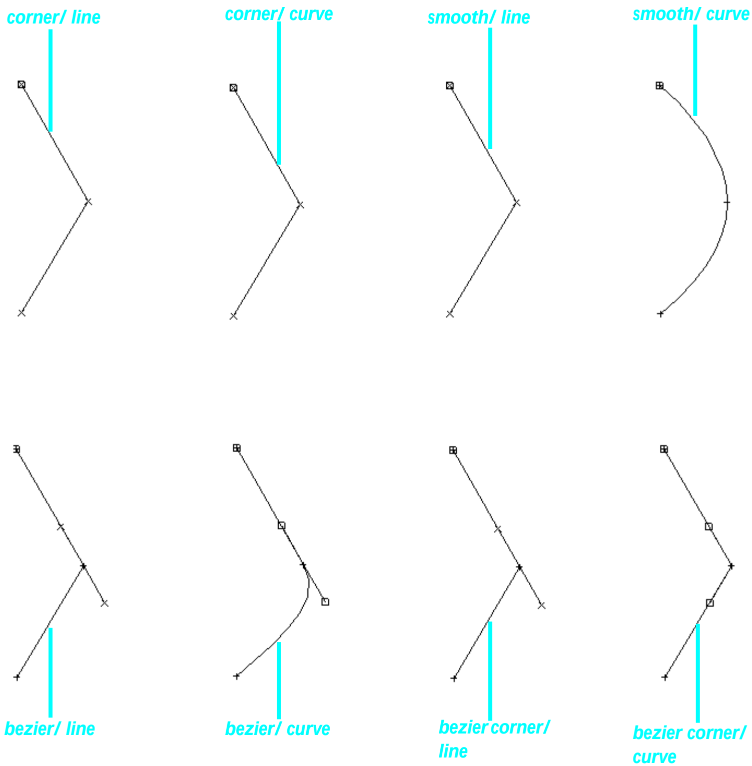
Locking Tangent Handles

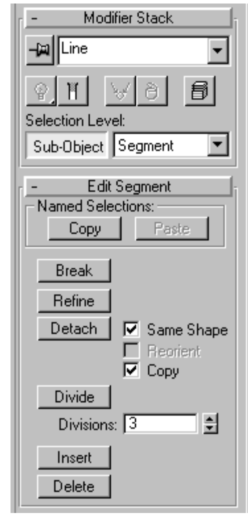
Normally you can transform the tangent handles of only one vertex at a time, even when multiple vertices are selected. Use the Lock Handles controls to transform multiple Bezier and Bezier Corner handles simultaneously.

All—Locks all handles of all selected vertices for simultaneous motion. This option is also useful when working with a single Bezier Corner vertex when you want to move both handles.

Alike—Locks all handles on all selected vertices so that moving the incoming tangent handle on one vertex moves the incoming tangent handle for all selected vertices. Moving the outgoing tangent handle on one vertex moves the outgoing tangent handle for all selected vertices. This option is really only noticeable with Bezier Corner vertices because moving either handle of a Bezier vertex always moves both handles.

Editing Spline Segments





Edit Segment rollout

Segment properties in the right click menu

Spline segments connect a pair of vertices, and one or more segments combine to form a spline. Choose Segment at the Editable Spline sub-object level, to expand the Edit Segment rollout.

Refining, Breaking, and Inserting Segments

Insert, Refine, and Break work like their counterparts on the Edit Vertex rollout. They add vertices to a segment and convert a single segment into two segments.

Dividing Segments

Select segments and click Divide to divide them into an equal number of divisions as specified in the Divisions parameter.

Detaching Segments

Click Detach, with Same Shape selected, to break selected segments at their connected vertices. If you select the Copy option, Detach duplicates the selected segments.

Detaching Segments to a Separate Shape

Click Detach, with Same Shape clear, to convert selected segments into an independent shape.If

you select the Copy option, Detach duplicates the selected segments. If Copy is clear, the selected segments are removed from the shape.

You can also choose the Reorient option to rotate the selected segments so their centers and local coordinate systems match the origin and orientation of the current construction plane. Otherwise, Detach leaves the selected segments in their original orientation.

Deleting Segments

Click Delete to delete one or more selected segments from a spline. Deleting a segment also deletes its vertices if no other segments share the vertices.

Changing Segment Properties

You can change the properties of a segment from Line to Curve using the shortcut menu.

The effect of changing segment properties varies according to the type of vertices at the segment end:

- Corner vertices always result in line segments regardless of the segment property.
- Smooth vertices can support both line or curve segment properties.
- Bezier and Bezier Corner vertices apply their tangent handles only to curve segments. Tangent handles are ignored by line segments.

A tangent handle associated with a line segment displays an X at the end of the handle. You can still transform the handle, but it has no effect until the segment is converted to a curve segment.

Editing Splines

Every shape object contains one or more splines. Choose Spline at the Editable Spline sub-object level to display the Edit Spline rollout.

Closing Splines

Click Close to create a segment between the end vertices of selected open splines.

Reversing Splines

Click Reverse to swap the location of the first vertex of selected open splines from one end vertex to the other.

Inserting Vertices in a Spline

Click Insert to add vertices to a spline just as you use Insert on the Edit Vertex rollout.

Mirroring Splines

Click Mirror to flip selected splines about their local X and Y axes. You can choose to mirror the original spline or mirror a copy of the spline.

Boolean Operations with Splines

Click Boolean to perform a Boolean operation on a selected spline and a second spline that you pick. The following conditions must be met for Boolean to work:

- Only one spline can be selected before clicking Boolean.
- The second spline you pick must be in the same shape as the selected spline.
- Both splines must overlap and be closed.
- Neither spline can be self-intersecting.

Union—Combines the two splines.

Subtraction—Carves the second spline out of the first spline.

Intersection—Creates a spline from the area where the two splines overlap.

Outlining Splines

Click Outline to create splines concentric to selected splines. The result depends on whether the selected spline is open or closed:

- Outlining a closed spline produces a second, concentric closed spline.
- Outlining an open spline connects the end vertices of the original spline with its outline.

You set the distance between the outlined splines by dragging or entering a value in the Width parameter.

Deleting Splines

Click Delete to remove a spline from a shape. Even if you delete all of the splines in a shape, the shape still exists. It is just an empty shape.

Changing Spline Properties

You change the properties of a spline from Line to Curve using the right-click menu. Changing the spline property also changes the property of all vertices in the spline:

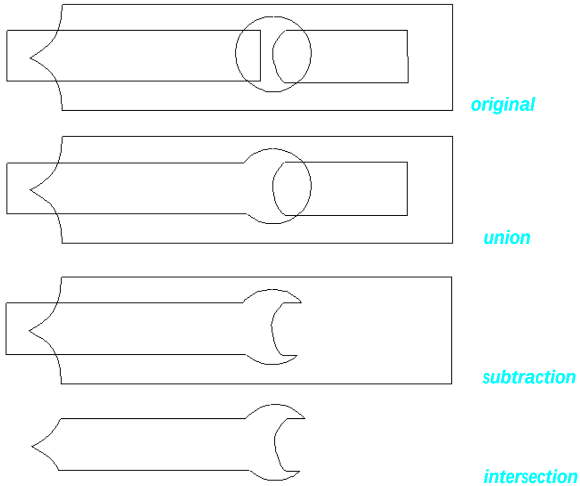
- Choosing Line converts vertices to Corners.
- Choosing Curve converts vertices Beziers.

Detaching Splines

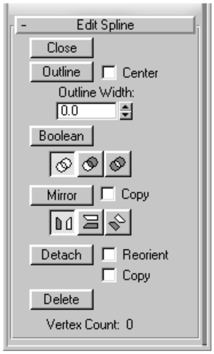
Click Detach to convert selected splines into an independent shape.

If you select the Copy option, Detach duplicates the selected splines. If Copy is clear, the selected splines are removed from the shape.

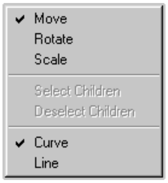
You can also choose the Reorient option to rotate the selected splines so their centers and local coordinate systems match the origin and orientation of the current construction plane. Otherwise, Detach leaves the selected splines in their original orientation.



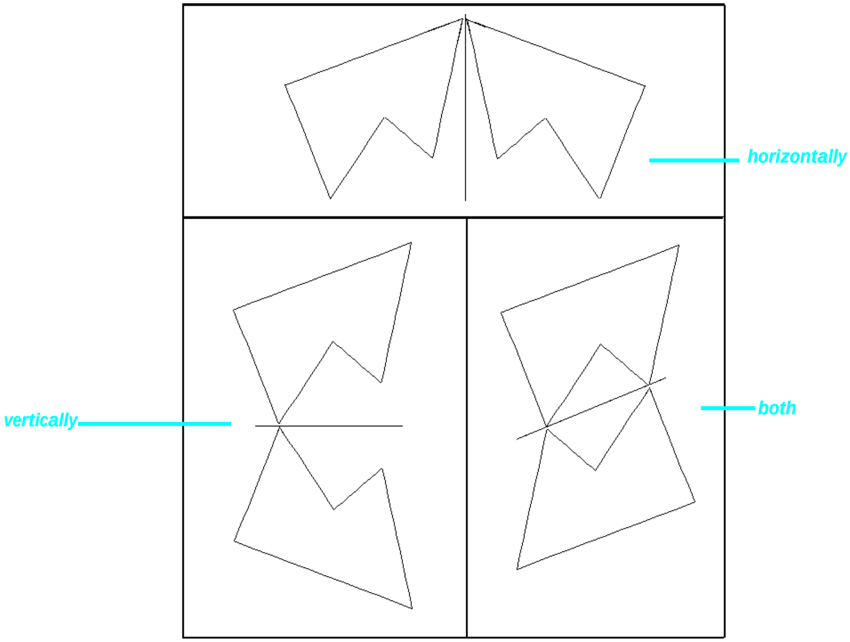
Results of Boolean operations



Edit Spline rollout



Spline properties
in the right-click menu



Mirroring splines

Chapter

9

Creating Lofts

3D Studio MAX contains a method for 3D object creation called *lofting*. You create shape objects to serve as a *path* for any number of cross-section shapes. The path becomes the framework that holds the cross sections forming your object.

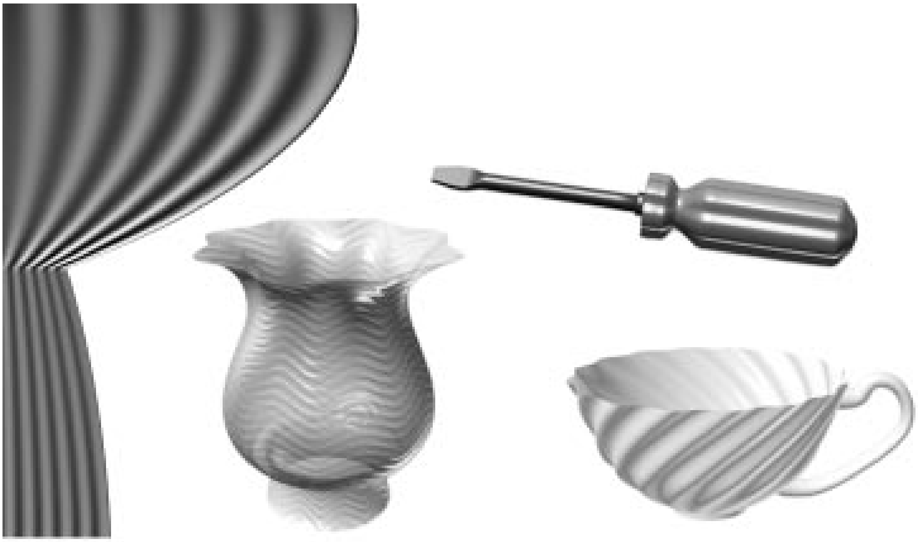
Once you create a loft object you can change and animate its parameters and sub-objects:

- Add and replace cross-section shapes or replace the path.
- Change or animate the parameters of the path and shapes.
- Change or animate the surface parameters of the loft object.

The lofting process first requires that you create shape objects to serve as the path and cross sections of your loft object. See chapter 8, “Creating Spline Shapes,” and chapter 10, “Creating NURBS Models.”

See the Online Reference for details about all of the Loft creation options.

What are Loft Objects?



The term *lofting* comes from early shipbuilding. A large framework called a loft was built to hold the hull of a ship while it was assembled. The process of hoisting the ribs (cross sections) of the hull into the loft became known as lofting. You may find an example from modern automobile design easier to understand.

A traditional method for building three-dimensional models of a vehicle design is to draw cross sections at a number of key points. These cross sections are cut out to form two-dimensional templates that are then placed on a rail. The model builders fill in the space between the templates to generate the surface of the model.

You create loft objects in 3D Studio MAX using a similar process. You first create two or more shape objects. The shapes can be either spline or NURBS curves. One of these shapes will be the rail, which 3DS MAX calls the path. The rest of the shapes are the cross sections of your object which are called shapes. As you arrange your shapes along the path, 3DS MAX generates a surface between the shapes.

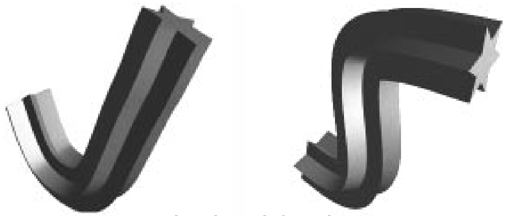
If there is only one shape on the path, 3DS MAX assumes an identical shape is located at each end of the path. The surface is then generated between the shapes.

3D Studio MAX places few restrictions on how you create your loft object, unlike the traditional shipbuilding and model-building examples above. Using 3DS MAX you can create curved, three-dimensional paths and even three-dimensional cross sections.

Uses for Loft Objects

Loft objects provide an extremely flexible modeling tool for creating complex objects. Common uses for loft objects include these complex objects:

- Objects with complex or variable cross sections.
- Objects with animated cross sections.
- Morph targets.



Animating a loft path

Animating loft shapes

Animating Loft Objects

You have many options for animating loft objects. Aside from the normal animation methods, such as animating loft transforms and modifiers, such as Taper and Bend, you can also do the following:

- Animate the *loft path* to make the loft wriggle and move while maintaining a constant cross section.
- Animate the *loft shapes* to create bulging and rippling effects under the loft skin.
- Animate *loft deformation curves*.

Basic Lofting Process

Creating loft objects is detailed and offers many choices, but the basic process is quite simple.

To create a loft object

1. Create a shape to be the loft path.

2. Create one or more shapes to be loft cross sections.

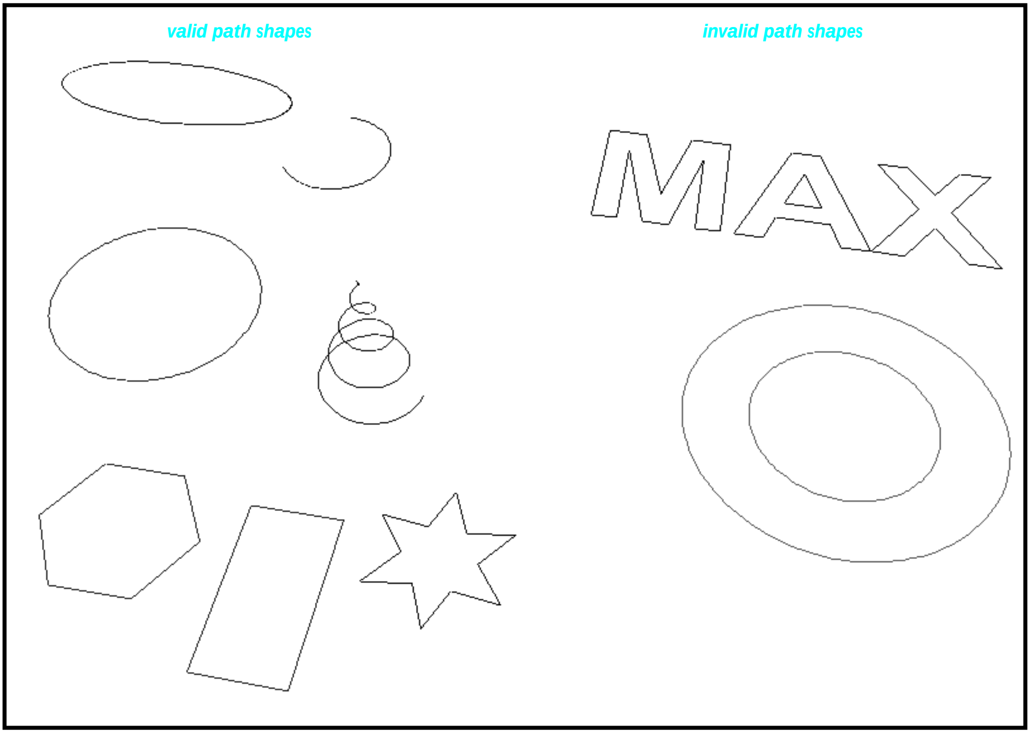
3. Do one of the following:

   - Select the path shape and use Get Shape to add the cross sections to the loft.
   - Select a shape and use Get Path to assign a path to the loft. Use Get Shape to add additional shapes.

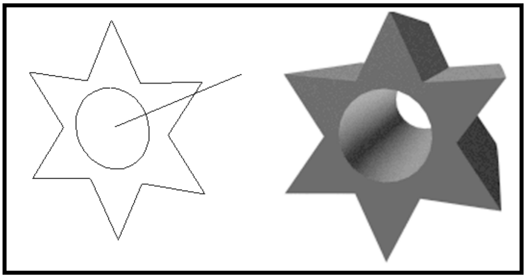
You can use the loft display settings to view the skin generated by your loft in both wireframe and shaded views.

ABCD

Creating Shapes for Loft Objects



Valid and invalid path shapes



Lofting a nested shape

You use the shape commands in the Create panel to create path and cross-section shapes for loft objects. Almost any open, closed, 2D, or 3D shape can be used as a path or a cross section. There are very few restrictions on shapes that you use to create loft objects.

Creating Path Shapes

The only restriction for path shapes is that the shape must be made of a single spline or NURBS curve. For example, compound shapes like a Donut or Text string are not valid path shapes.

Open spline paths with large Bezier tangents at their end vertices are valid paths but they might produce unpredictable results. Loft path shapes work best if their end vertices use the Corner type or have short Bezier tangents.

Creating Cross-Section Shapes

There are two restrictions for cross section shapes on a loft path:

- All shapes on the path must contain the same number of spline or NURBS curves. For example, you cannot use a Donut at one end of the path and a Rectangle at the other end.
- Compound shapes must have the same nesting order. You cannot have a nested shape, such as a circle inside a square, at one end of the path and a non-nested shape, such as a circle beside a square, at the other end.

3D Studio MAX checks shapes as you add them to the path and prevents you from adding incompatible shapes on the path.

Lofting Nested Shapes

Nested shapes contain multiple closed spline or NURBS curves where some curves are completely surrounded by another curve. When you loft a nested shape, the inner curve becomes a hole running up through the outer curve. If you have multiple nested curves within the same shape, they alternate from solid to hole as you move from the outer curve to the inner curve.

Lofting Self-Intersecting Shapes

Self-intersecting shapes contain overlapping curves, or curves that cross over themselves. The results of lofting overlapping shapes can be unpredictable:

- Shapes with overlapping closed curves, or an open curve that crosses over itself, loft best.
- Shapes with a closed spline that crosses over itself often do not cap properly.

Lofting Open Shapes

You can loft shapes with open curves to create thin, ribbon-like surfaces. You should apply two-sided materials to lofts using open curves for proper rendering.

Controlling Loft Density

The face density of a loft surface affects a number of factors regarding the appearance and use of the loft object:

Accuracy—Dense surfaces accurately represent curves and bends in the loft surface.

Deformability—Dense surfaces deform more accurately than sparse surfaces with fewer faces.

Rendering Quality—Sparse surfaces can contain long, thin faces that cause rendering and smoothing problems. Dense surfaces are made of smaller, evenly-sized faces that render better.

Rendering and Redraw Speed—Dense surfaces take longer to render and redraw. Sparse surfaces with fewer faces render and redraw quickly.

You control loft density by the choices you make when building the shapes that make up a loft object, and by your choices for the shape and loft parameters.

Lofting Morph Targets

A common use for loft objects is the generation of *morph targets*. Morphing is the process where the vertices of one object are animated to match the vertex positions of other objects. The objects used to create the morph are referred to as morph targets. This process can produce very complex and organic looking animation. See chapter 13, "Creating Compound Objects."

Loft objects make excellent morph targets because, given a few shapes on a path, you can transform and modify the shapes to create many morph targets. If you plan to use loft objects as morph targets, you must ensure that each loft object has the exact *same number of vertices*. This topic discusses the techniques needed to create morph targets successfully.

Using Morph Capping

When capping is set to Morph, the number of faces used to build the cap will always be the same as long as the shape has the same number of vertices.

Compare two lofts, on the next page, capped with Morph and Grid capping. If the shape is modified and capped again, Morph capping produces the same number of faces. Grid capping adapts the number of faces to best fit the shape and therefore cannot be used for creating morph targets.

Using Fixed Step Settings

When Adaptive Path Steps is selected, shapes can exist anywhere on the path. As the path and shapes are modified to produce morph targets, the number of faces used to produce the skin can change.

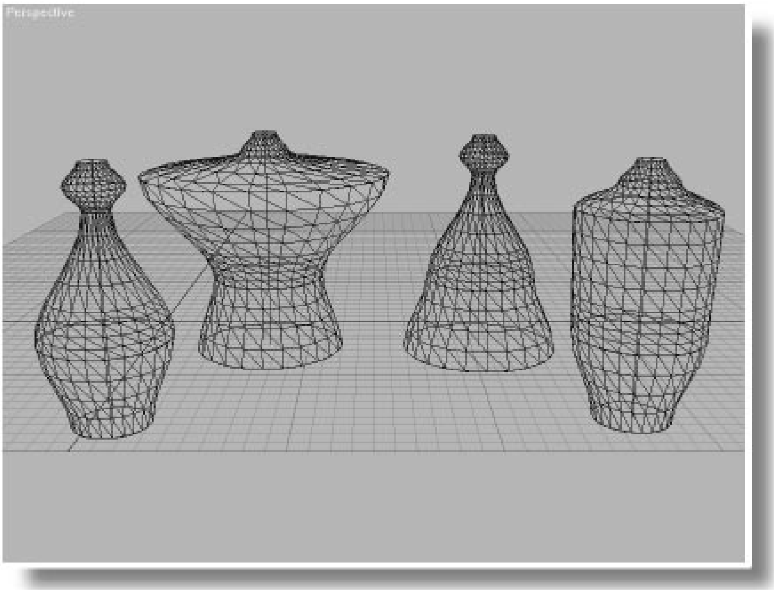
Clear the Adaptive Path Steps option to ensure predictable face counts when lofting morph targets.

Guidelines for Lofting Morph Targets

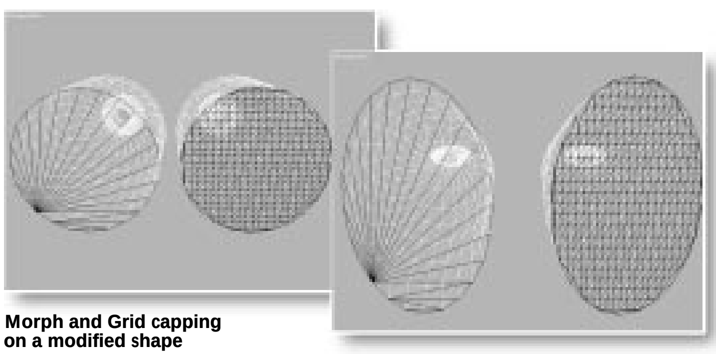
The following guidelines will help you successfully loft morph targets. None of these guidelines are absolutely required, but contradicting one or more of them might lead to unpredictable results. The only rule for morph targets is that they all have the same number of vertices.

These guidelines control the number of vertices and step settings for the path and shapes:

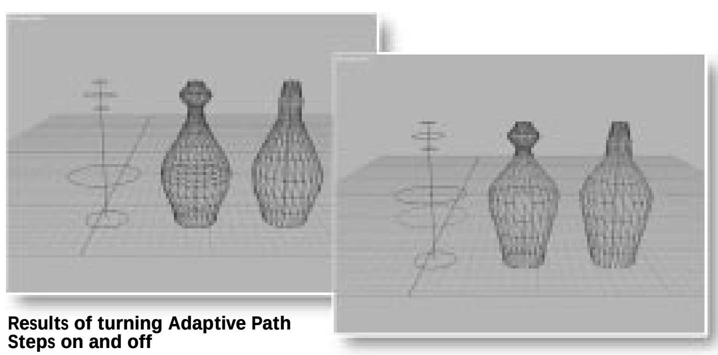
- The paths of all lofts used as morph targets should have the same number of vertices.
- The Path Steps setting for all loft objects should be the same.
- Adaptive Path Steps should be cleared.
- The shape levels and shape vertex count between different lofts should match. If the original loft used a first shape with three vertices and a second shape with four vertices, then all lofts created as morph targets should also have a first shape with three vertices and a second shape with four vertices.
- The sequence of open and closed shapes on the path should be the same. If the original loft used three shapes arranged on the path as closed, open, closed, then all lofts created as morph targets should use the sequence of closed, open, closed.
- The Shape Steps setting for all loft objects should be the same.
- Shape Optimize should be cleared.
- Capping should be set to Morph.



Lofted morph targets

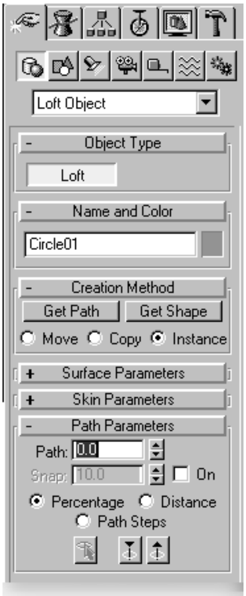


Morph and Grid capping on a modified shape



Results of turning Adaptive Path Steps on and off

Lofting with Get Path



You can create a loft object by selecting a shape and then clicking Get Path to assign a path to that shape. Use Get Path to do the following:

•Create a loft object where the loft base is aligned with the first cross-section shape.

•Replace the path of an existing loft with a new path.

When you click Get Path, the selected shape becomes the first cross-section shape of the loft and you pick a shape to be the path. The cursor changes to the Get Path cursor as you drag it over potential valid path shapes. If the cursor does not change over a shape, that shape is not a valid path shape and cannot be selected.

You choose from three options that control what happens to the picked path shape:

- Move the original shape into the loft.
- Place a copy of the shape in the loft.
- Place an instance of the shape in the loft.

Tip: You might find it easier to use the Instance option if you expect to edit or modify the path after the loft is created.

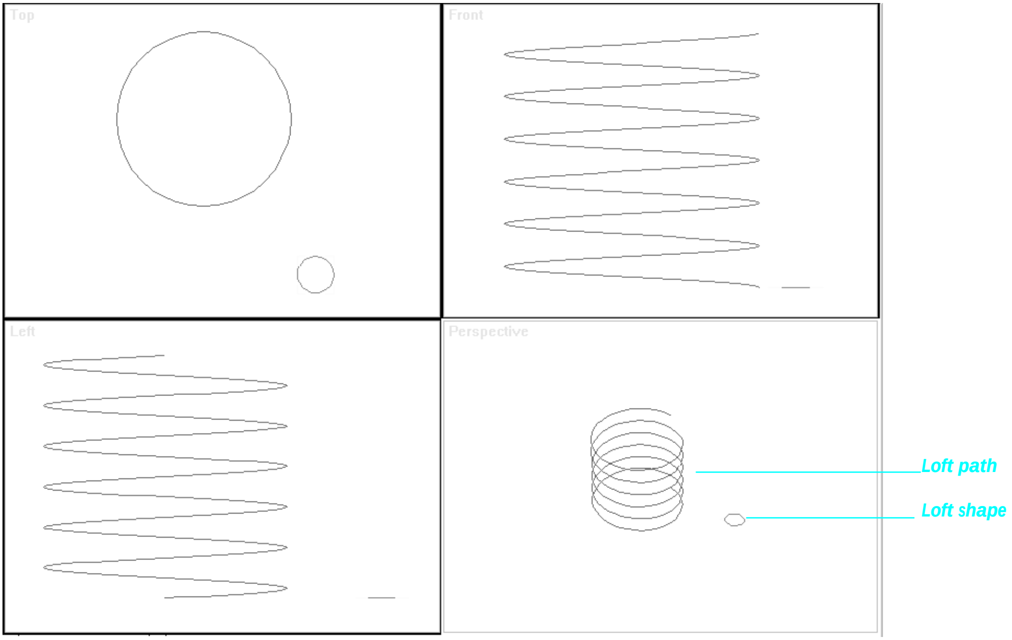
Orientation of the Path

You use Get Path when you want the path to move to the location of the selected shape. You also use Get Path when you have created a shape at the location where you want the base of your loft object to be.

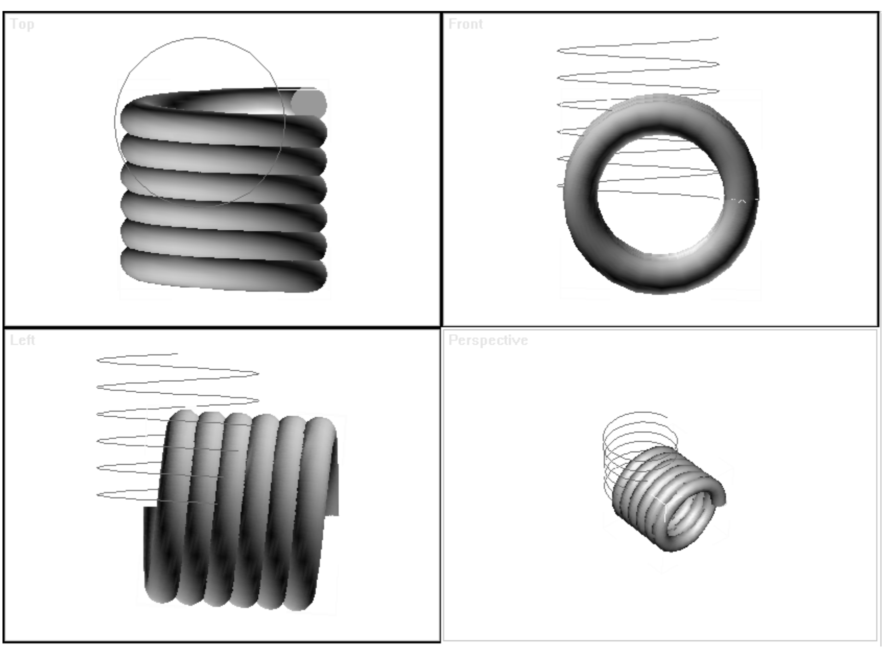
Get Path moves and rotates the path shape to align it with the first cross-section shape.

- The first vertex on the path is located at the first shape's pivot point.
- The tangent to the first vertex on the path is aligned with the positive Z axis of the first shape.
- The local Z axis of the path is aligned with the local Y axis of the first shape.
- The local coordinate system of the resulting loft object equals the local coordinate system of the path after it has been aligned with the first shape.

Sometimes, aligning the tangent of the path with the positive *Z* axis of the first shape does not produce the result you want. Press CTRL while getting a path to align the first vertex of the path tangent to the negative Z axis of the shape.



Loft path and shape in default views



Default views after lofting object with Get Path

Lofting with Get Shape

You can create a loft by selecting a path and then clicking Get Shape to place a cross section on the path. Use Get Shape to do the following:

- Create a loft object aligned with a path shape.
- Add shapes anywhere on the path.
- Replace shapes on the path with new shapes.

When you click Get Shape, the selected shape becomes the path of the loft and you pick a shape to be the first cross section. The cursor changes to the Get Shape cursor as you drag it over potential shapes. The shape you pick is placed at the first vertex of the path.

You choose from three options that control what happens to the picked cross-section shape.

- Move the original shape into the loft.
- Place a copy the shape in the loft.
- Place an instance of the shape in the loft.

Tip: You might find it easier to use the Instance option if you expect to edit or modify the shapes after the loft is created.

Orientation of the Shape

You use Get Shape when you want the shape to move to the location of the selected path. You also use Get Shape when you have created a path at the location where you want your loft object to be.

Get Shape moves and rotates the shape to align it with the current level of the path.

- The pivot point of the shape is located on the path at the current path level.
- The positive Z axis of the shape is aligned tangent to the path at the current path level.
- The local Y axis of the shape is aligned with the local Z axis of the path.

Sometimes, aligning a shape's positive Z axis tangent to the path does not produce the result you want. Press CTRL while getting a shape to align the shape's negative Z axis tangent to the path.

Setting the Path Level

When you add shapes to a loft, you specify the location, or *level*, on the path where shapes are placed. Enter a value in the Path parameter on the Path Parameters rollout to set the path level. You can identify the current level by looking for the path-level indicator, a small yellow "X" on the path.

You choose a method used to specify the path level from the following options:

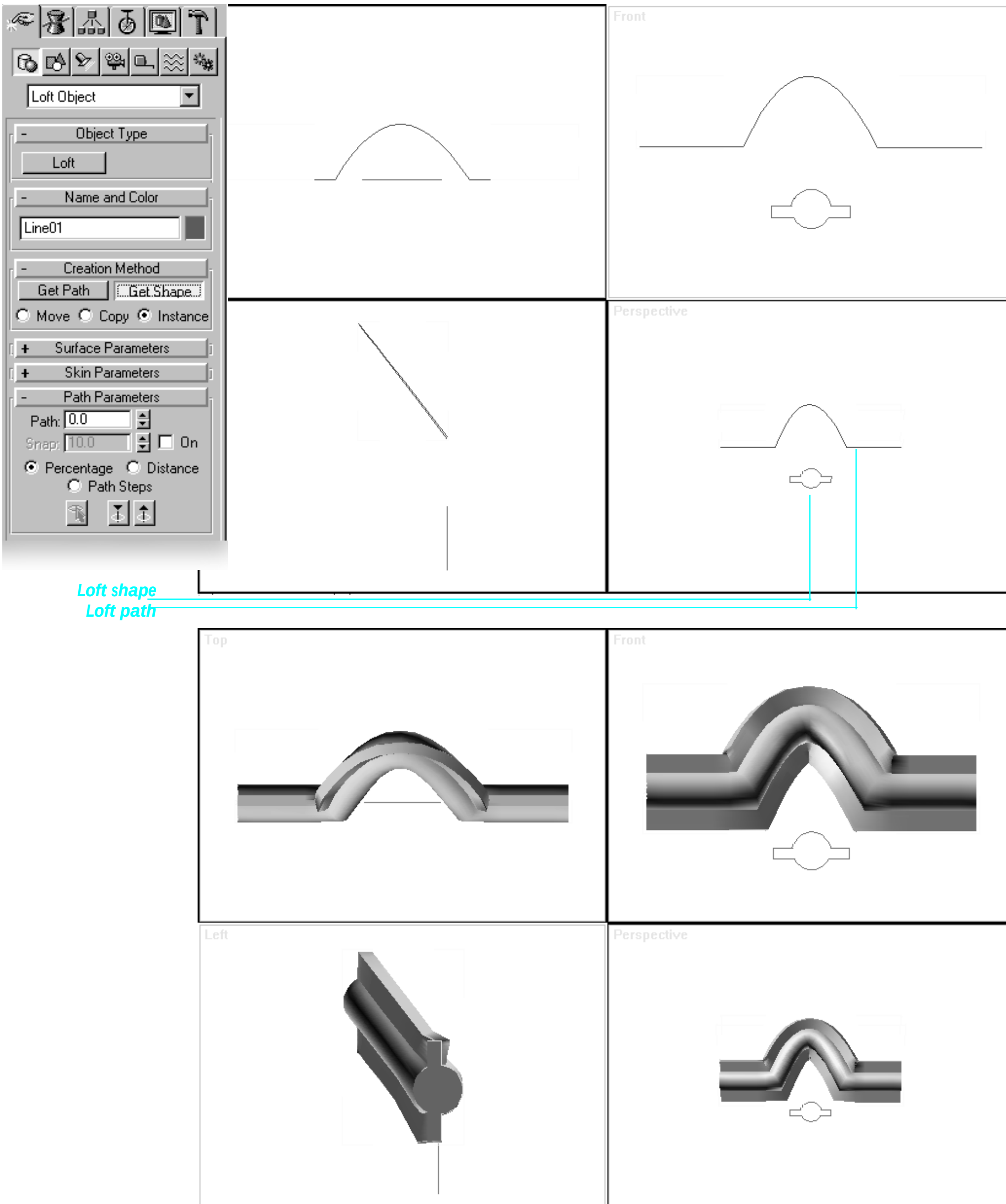
Percentage—Expresses path level as a percentage of the total path length.

Distance—Expresses the path level as an absolute distance from the first vertex of the path.

Path Steps—Expresses the path level as a number of steps from the start of the path. Path Steps are the vertices and subdivisions between vertices along the path.

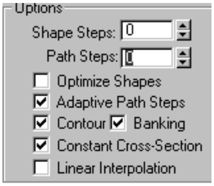
You can switch from one method to another at any time. Consider the following when switching the Path Level method:

- Shape path levels are stored using the percentage method. If you change the length or shape of the path, the path level distance or steps may also change. The path-level percentage remains constant.
- Selecting Path Steps causes all shapes on the path to move to the nearest step location. This may place some shapes on the same level and cause unpredictable results.



Default views after lofting object with Get shape

Controlling Skin Geometry



You control the face density, shape orientation, and viewport display of a loft by setting parameters on the Skin Parameters rollout. See the Online Reference for details.

Setting Shape Density

The parameters Shape Steps and Optimize Shapes, control face density around the loft perimeter.

Shape Steps—Affects the number of loft sides by setting the steps between each vertex of cross-section shapes.

Optimize Shapes—When selected, straight segments of cross-section shapes have zero steps. Only straight segments that have a match on all cross section shapes get optimized.

Setting Path Density

Parameters Path Steps and Adaptive Path Steps control the face density along the path of the loft.

Path Steps—Affects the number of loft segments by setting the steps between each main division of the path.

Adaptive Path Steps—When selected, path divisions are placed to generate the best skin. Divisions along the path occur at path vertices, shape locations, and deformation curve vertices When Adaptive Path Steps is cleared, divisions along the path only occur at path vertices.

Imagine two loft objects with opposite Adaptive Path Steps settings. If the cross-section shapes are moved to new positions, the loft with Adaptive Path Steps cleared produces the same number of divisions as before. The loft with Adaptive Path Steps selected changes the number of path divisions to produce a better-looking skin.

Using Contour and Banking

The Contour and Banking options affect the orientation of shapes on the path. These parameters affect all shapes on the path. If you want to change the orientation of individual shapes you can rotate shapes at the sub-object level. See "Transforming Loft Shapes."

Contour—When selected, each shape remains aligned to the path tangent. When cleared, all shapes remain parallel to the shape at level 0.

Banking—When selected, shapes rotate about the path as the path bends and changes height. Banking has no effect if the path is 2D.

Holding a Constant Cross Section

Select Constant Cross Section to scale cross-section shapes at sharp path corners to maintain a uniform width. When cleared, cross sections maintain their original size, causing pinching at path angles.

Choosing Linear or Curved Skinning

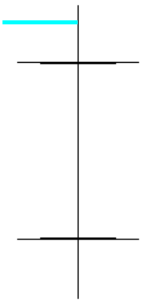
Select Linear Interpolation to generate a loft skin with straight edges between each shape. When cleared, loft skin with smooth curves between each shape is generated.

The generation of a curved skin requires a sufficient number of path steps to represent the curve. If path steps are set too low, or at 0, the skin will always appear linear.

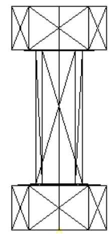
Capping the Loft Ends

The skin generated between closed shapes on an open path forms a hollow tube, open at both ends. Select capping options to close one or both ends of the loft and to specify the capping type.

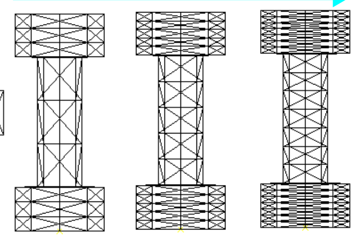
wireframe in front view

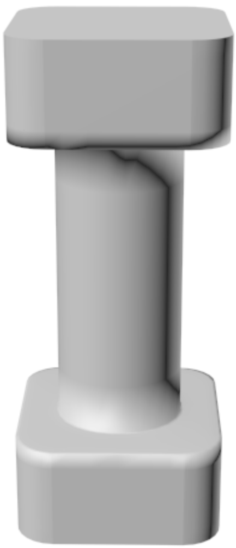


0 shape and path steps

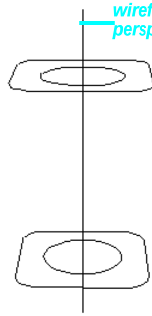


results of increasing shape steps

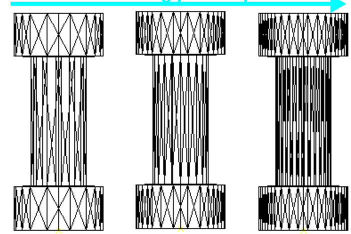




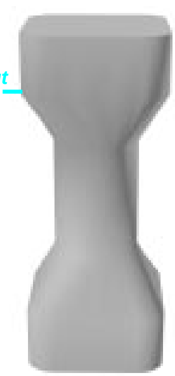
wireframe in perspective view



results of increasing path steps



skinned and lofted object with default settings

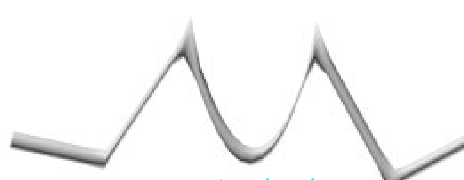


above object without adaptive path steps

banking selected

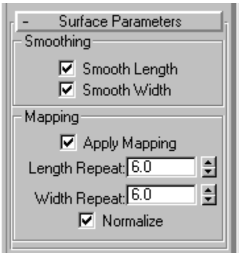
banking cleared

contour selected



contour cleared

Controlling Surface Appearance



You control smoothing and mapping properties of a loft object by setting parameters on the Surface Parameters rollout.

Smoothing the Loft Surface

Smoothing is a rendering technique that enables a surface made of many faces to appear as a single smooth skin.

- Select Smooth Length when you want a smooth surface along the length of the path. Use Smooth Length when your path curves or when shapes on the path change size.
- Select Smooth Width when you want a smooth surface around perimeter cross-section shapes. Use Smooth Width when your shapes change the number of vertices or change form.

Smoothing and Step Settings

Smoothing makes face edges less noticeable, but it cannot make up for using too few faces. Use Path Steps and Shape Steps parameters with smoothing to achieve the best results.

- A model with no smoothing but very high step settings will appear smooth because of the many small faces making up its surface.
- A model with smoothing but very low step settings will appear somewhat faceted because the smoothing functions do not have enough faces to work with.

Smoothing and step settings combine to create a smooth and efficient model.

Smoothing with Modifiers

You can also change the smoothing of a loft object by applying modifiers such as Optimize and Smooth.

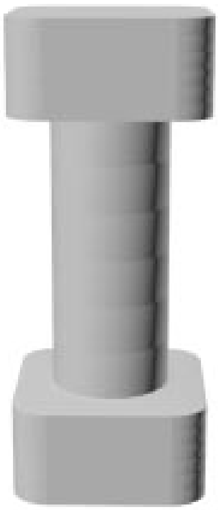
Lofting Map Coordinates

Mapping coordinates tell 3D Studio MAX how to apply mapped materials to an object. Mapped materials use image files to simulate surface effects such as patterns, bumps, and opacity. See chapter 21, “Designing Mapped Materials.”

Along with standard mapping functions available for any object, you can also use lofted mapping. Lofted mapping makes maps follow the loft path and repeat along the loft skin.

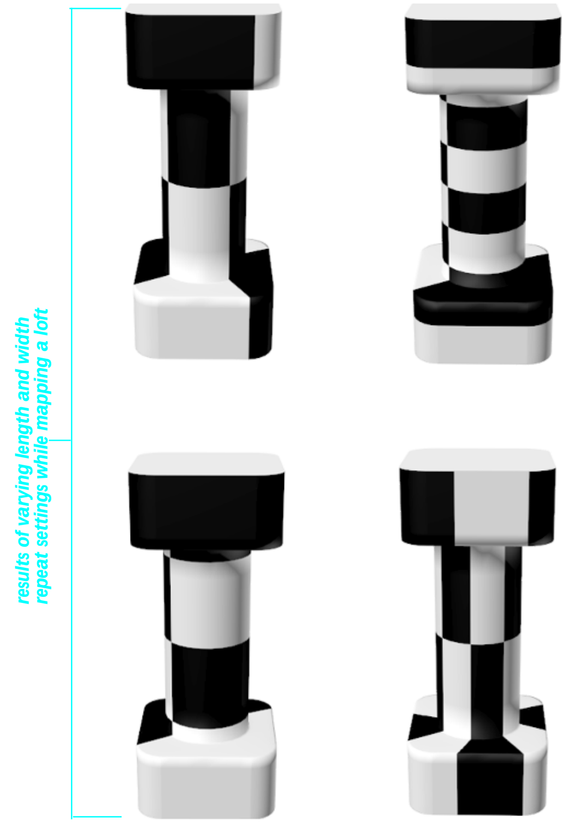
- Select Apply Mapping to enable lofted mapping parameters.
- Enter values in the Length and Width repeat parameters to set how many times maps repeat along the path length and perimeter.
- Select Normalize to assign map coordinates and repeat values evenly along the length of the path and around the shapes.

  When you clear Normalize, map coordinates and repeat values are applied proportionally according to the path division spacing or shape vertex spacing.



Smooth length unchecked

Smooth width unchecked



Normalized loft map

Normalized unchecked during loft mapping

Editing Lofts with Deformation Curves

You can create complex loft objects by placing shapes along a path, but this method has some limitations.

- You want your shape to rotate as it travels along the path to produce screw threads.
- You want your shape to change scale along the path to produce an ornate column.

Manually creating and placing shapes along the path to produce these models would be a difficult task. 3D Studio MAX solves this problem through the use of loft *deformation curves*. The deformation curves define changes in scale, twisting, teetering, and beveling along the path.

You activate and edit loft deformation curves using parameters on the Deformations rollout on the Modify panel.

Deformation windows use a similar layout. You can perform the following functions:

- Change deformation curve display.
- Edit deformation control points.
- Navigate the Deformation window.

Reading the Deformation Grid

The view where the deformation curves are displayed is called the *deformation grid*. This grid charts the value of the deformation along the length of the path.

Deformation values are measured on the vertical scale for each deformation type. The following table lists deformation types and deformation values.

Deformation Type	Deformation Value
Scale	Percentage
Twist	Rotation Angle
Teeter	Rotation Angle
Bevel	Current Units

Levels on the path are measured on the horizontal scale and appear as vertical lines.

- If Adaptive Path Steps is selected, levels display at path vertices, and shape locations.
- If Adaptive Path Steps is cleared, levels display only at path vertices.

Reading Deformation Curves

You can see one or two curves in the Deformation dialog based on deformation type and curve display setting. Deformation curves are color-coded by axis:

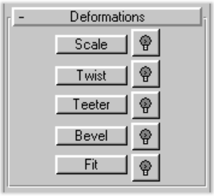
- A red curve displays deformation along the shape's local X axis.
- A green curve displays deformation along the shape's local Y axis.

At the bottom of the Deformation window are two edit fields. When a single control point is selected these fields display the path location and deformation amount of the control point.

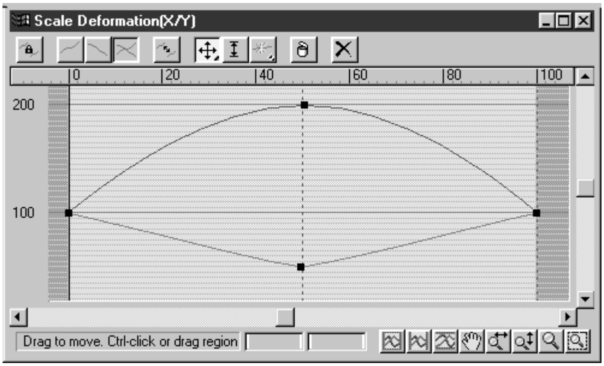
Displaying Deformation Curves

You can display one or both of the deformation curves by using the curve display buttons near the top left corner of the Deformation dialog.

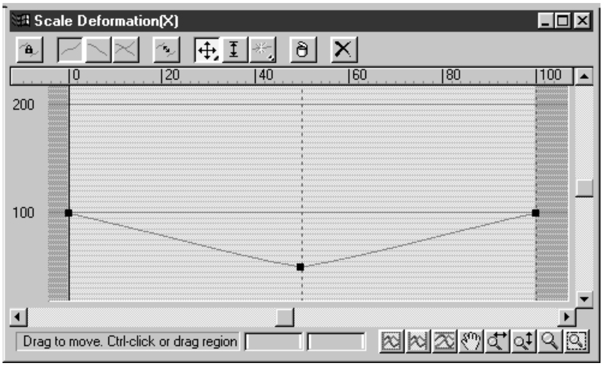
If a deformation uses only one curve, such as Twist, buttons for working with a second curve are disabled. The disabled buttons are: Make Symmetrical, Display X Axis, Display Y Axis, Display XY Axes, and Swap Deform Curves.



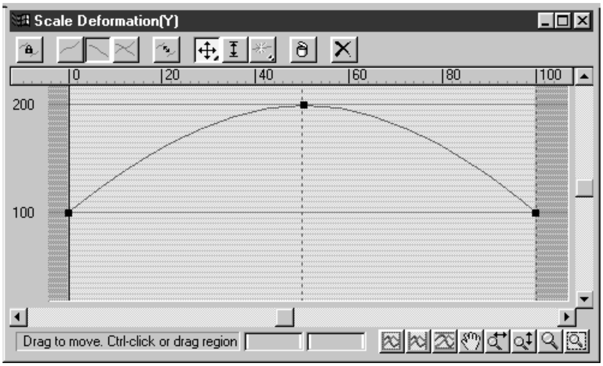
Deformation rollout on the Modify Panel



Displaying X and Y axis curves



Displaying the X axis curve



Displaying the Y axis curve

Editing Deformation Curves

When you first apply a deformation, you start with a straight line using a constant value. To produce more elaborate curves, you insert control points and change their properties.

Setting Curve Symmetry

You can apply the same deformation to both axes of a shape. Enable Make Symmetrical mode to copy the current deformation curve to the opposite axis. While Make Symmetrical mode is active, changes you make to one curve are also applied to the opposite curve.

Swapping Deformation Curves

Click Swap Curves to copy the X axis curve to the Y axis, and the Y axis curve to the X axis. It doesn't matter which curve is currently displayed or selected. Swap Curves has no effect when Make Symmetrical mode is active.

Setting Control Point Types

Control points on a deformation curve can produce curves or sharp corners based on their type.

Corner—Nonadjustable linear control point producing a sharp corner.

Bezier Corner—Adjustable Bezier control point with discontinuous tangent handles set to produce a sharp corner.

Bezier Smooth—Adjustable Bezier control point with locked continuous tangent handles set to produce a smooth curve.

You can change control point types at any time by right-clicking on a selection of one or more control points.

Inserting Control Points

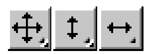
Click Insert Control Point to insert Corner or Bezier Smooth control point types.

Deleting Control Points

You can delete selected control points or all of the control points on the curve.

Click Delete Control Point to delete selected points from the deformation curve.

Click Reset Curve to delete all control points except the first and last, and reset the curve to its default value.



Moving Control Points

Click Move Control Point to move control points and drag Bezier handles. You can also move single control points by entering values in the control point Position and Amount fields at the bottom of the Deformation dialog.

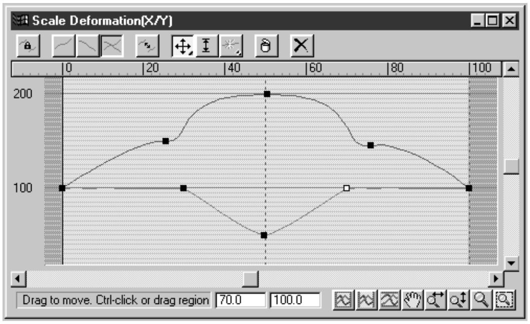
Control points are constrained horizontally to stay between the points to either side. Horizontal constraint is set by control point type:

- Corner control points can move very close together, until one is directly above the other.
- Bezier control points move no closer than the horizontal length of their tangent handles.

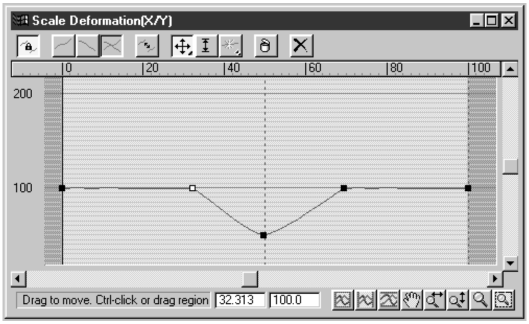
Dragging Bezier Tangent Handles

You click Move Control Point to drag a tangent handle's angle and magnitude on Bezier Smooth and Bezier Corner vertices.

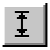
- Tangent angles cannot move beyond vertical. This prevents deformation curves from doubling back on themselves.
- Tangent magnitudes cannot extend past the preceding or next control point on the path.
- Press SHIFT while moving a Bezier Smooth tangent handle to convert the control point to a Bezier Corner type.



X and Y curves before symmetry is applied



X and Y curves after applying Make Symmetrical to X



Scaling Control Points

Click Scale Control Point to scale the value of selected control points. Use this button when you want to change only the deformation amount of selected control points while maintaining their relative ratio of values.

Animating Deformation Curves

Turn on the Animate button and transform control points to create animated deformations. Each animated control point is numbered and displays in Track View.

You can animate only the position and deformation amount of a control point. Changes to the angle and magnitude of Bezier control point tangent handles are not animated.

Applying Basic Deformation Types

Scale deformation applied

Twist deformation applied to scaled object

Teeter deformation applied to scaled object

Bevel deformation applied

This topic describes four deformation types. The fifth deformation type, Fit Deformation, is discussed in the following topics. You can apply any combination of deformations to a loft.

Using Scale Deformation

Use Scale deformation to create objects such as columns and bugles from a single shape that changes scale as it travels along a path.

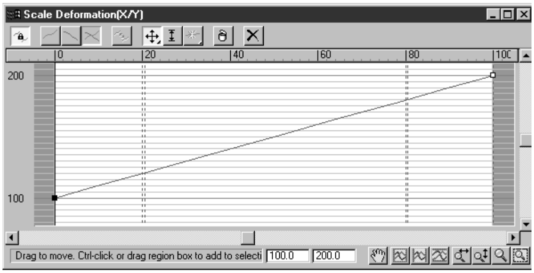
Using Twist Deformation

Use Twist deformation to create objects that spiral or twist along their length. Twist specifies the amount of rotation about the path.

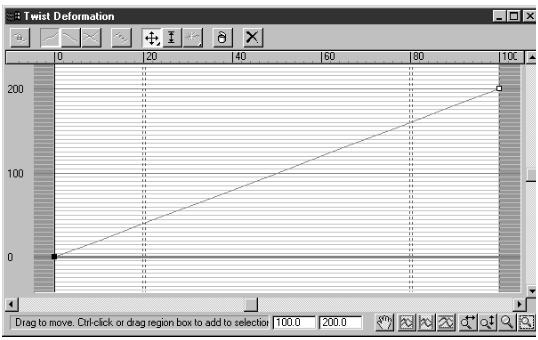
You can use Twist deformation as a way of exaggerating or reducing the amount of banking. Information about banking is presented in the topic “Controlling Skin Geometry.”

Using Teeter Deformation

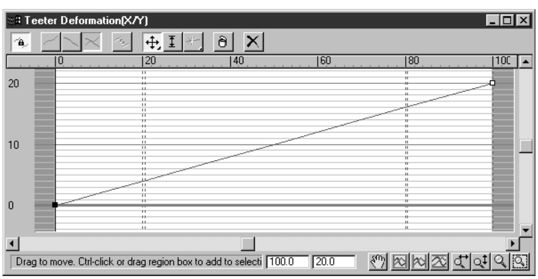
Use Teeter deformation to create objects that rotate about their local *X* axis and *Y* axis. Teetering is what 3D Studio MAX does automatically when you select Contour on the Skin Parameters rollout. Use Teeter deformation when you want to manually control contour effects.



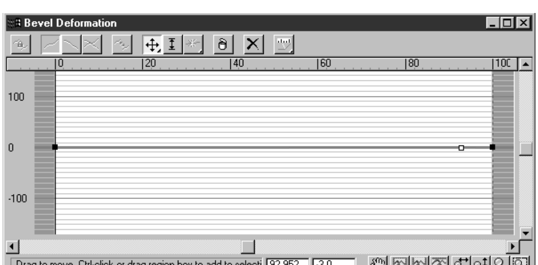
Scale deformation curve used previous illustration



Twist deformation curve used in previous illustration



Teeter deformation curve applied in previous illustration



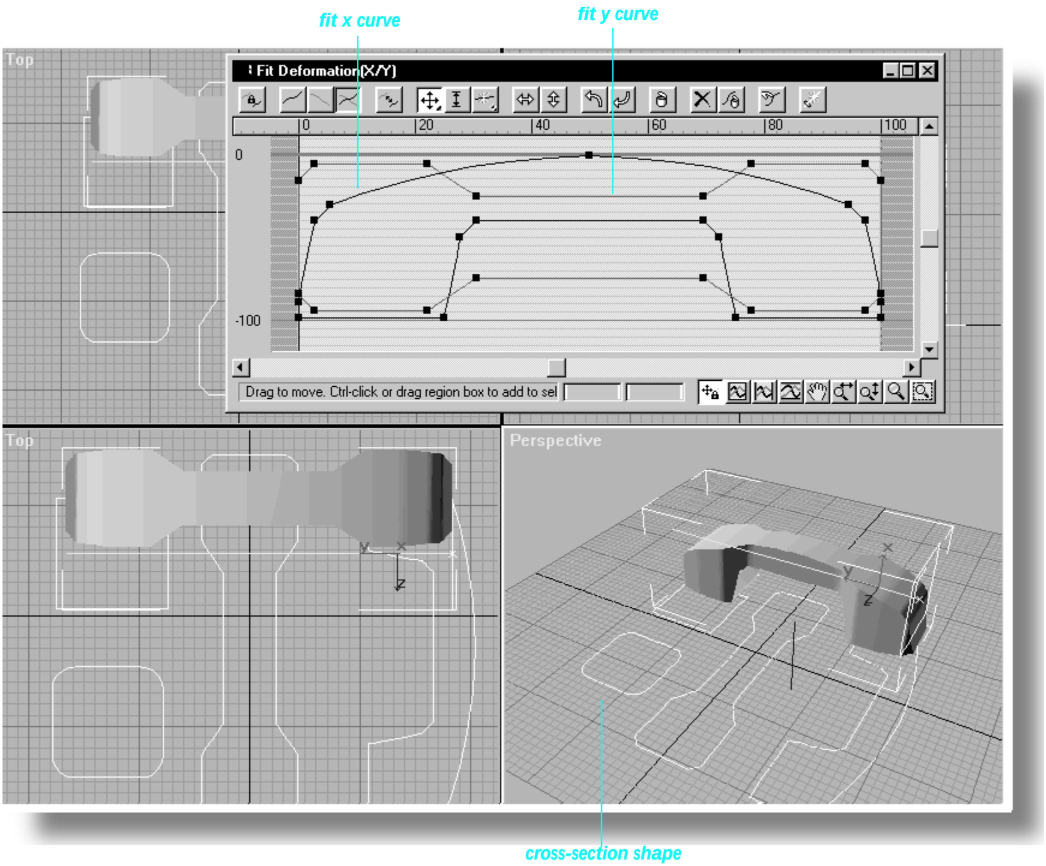
Bevel deformation curve applied in previous illustration

Using Bevel Deformation

Use Bevel deformation to create objects with eased edges. Almost every object that you encounter in the real world is beveled. Because it is difficult and expensive to manufacture a perfectly sharp edge, most objects are created with chamfered, filleted, or eased edges. Use Bevel deformation to simulate these effects.

Use the Adaptive Linear and Adaptive Cubic bevel options to correct for intersecting shapes when producing extreme bevels.

Applying Fit Deformation



Fit deformation uses two Fit curves that define the top and side profiles of your object. Use Fit deformation when you want to generate loft objects by drawing their profiles.

Understanding Fit Curves

Fit curves are really scale boundaries. As your cross-section shape travels along the path its X axis is scaled to fit the boundary of the X axis Fit curve and its Y axis is scaled to fit the boundary of the Y axis Fit curve.

One way of thinking about Fit Deformation and Fit curves is to imagine the six standard orthogonal views of a three-dimensional object. The six views can be grouped into three pairs of related parallel views:

- Left and Right
- Top and Bottom
- Front and Back

The Fit X curve, Fit Y curve, and shapes on the path each control one pair of views. You can set up your loft such that Fit curves and shapes on the path apply to any of the three pairs of orthogonal views.

The following is one possible arrangement:

- The Fit X curve defines the outline of an object as seen from the top, controlling the Top/Bottom views.
- The Fit Y curve defines the outline of an object as seen from the side, controlling the Left/Right views.
- The cross section shapes define the profile of an object's cross section, controlling the Front/Back view. The cross section shape is then scaled to the boundaries of the Fit X and Fit Y curves.

Creating Fit Curves

You can create Fit curves right in the Fit Deformation dialog or create shapes that you get from your scene. Either way, your Fit curves must follow certain rules:

- Curves must be single splines.
- Curves must be closed.
- Curves should not contain undercuts.
- Curves cannot extend past their first or last control point in the direction of the path.

Fit Curves and the Path

You choose whether the Fit curves are sized to the path or the path is sized to the Fit curves. Your choice depends on the following:

- Order in which you get fit shapes.
- Use of Generate Path.
- Use of Get Path.

Animating Fit Curves

Just as with other deformation curves, Fit curves can be animated. Each control point on the Fit curves is numbered and appears in Track View if you animate it.

Getting Shapes for Fit Curves

The primary method for creating Fit curves is to create a shape and then get the shape as a Fit curve.

 Click Get Shape on the Fit Deformation toolbar to get shapes from your scene as Fit curves.

Fit Curve Requirements

There are four requirements for creating shapes to use as Fit curves.

- The shapes must be single splines.

  Multiple spline shapes, such as a Donut or a shape made by attaching multiple splines, cannot be used as Fit curves.

- The splines must be closed.

  Shapes such as an open Line, Helix, or open Arc cannot be Fit curves. Use Edit Spline to close the spline.

- The splines should not contain undercuts.

  Undercuts occur when a spline reverses direction along the path axis. This happens because of vertex placement or Bezier handle position. You can bring a shape with an undercut into the Fit Deformation dialog but the effect on the Loft is unpredictable.

  You can correct undercuts by rotating the Fit curve 90 degrees so the undercut is not aligned with the path axis.

- Splines cannot extend past their first or last vertex in the direction of the path.

  This is always the result of Bezier tangent handles that push the curve beyond the vertex position. Edit the original shape at the Vertex sub-object level to change the Bezier tangents.

Fit Curve Orientation

When you get a shape for use as a Fit curve, the shape's original local X axis is aligned with the horizontal line representing the loft path in the Fit Deformation window. Note that transforms such as Rotate and Mirror, and changing the shape's pivot point do not affect shape orientation in the Fit Deformation dialog.

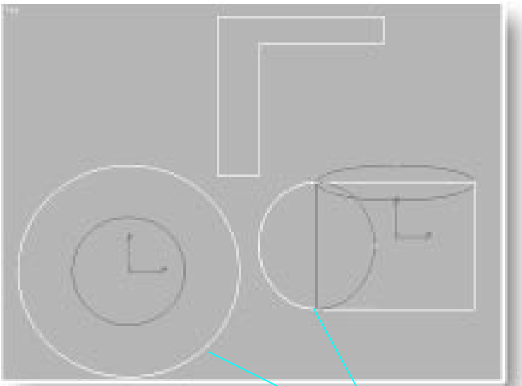
To change the orientation of a Fit curve:

- Use Rotate or Mirror on the toolbar of the Fit Deformation window.
- Rotate or Mirror the shape at the Spline sub-object level before getting the Fit curve.

Getting a Fit Curve

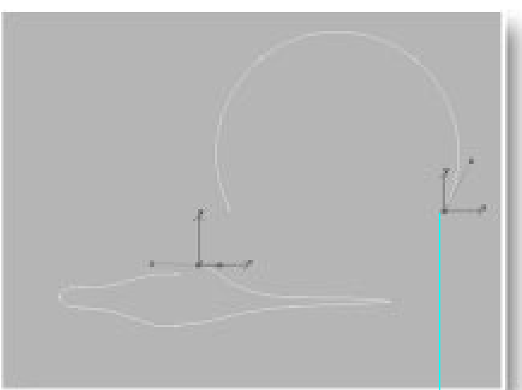
To set how a fit curve will be applied, you click one of the curve display buttons before you get a shape.

- Click Display X Axis or Display Y Axis before getting a shape to set whether you get the X Fit Curve or Y Fit Curve. The first shape you get sets the length of the path that is created by the Generate Path button.
- Click Make Symmetrical or Display XY Axes before getting a shape to get the same shape for both the Fit X Curve and Fit Y Curve.

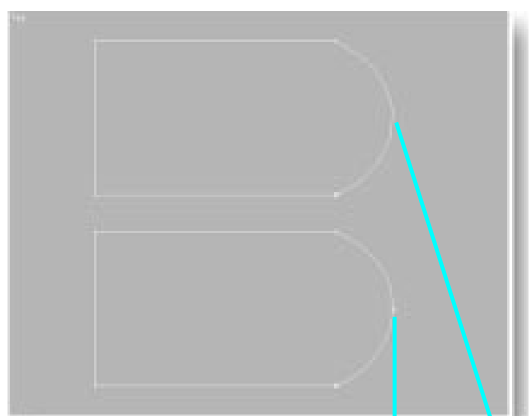


Fit curves must be
single spline shapes

Splines should not have undercuts



Splines must be closed



Splines should have vertices at their extents

Editing Fit Curves

Editing and transforming control points on a Fit Curve follow the same methods as editing deformation curves. The the topic “Editing Deformation Curves,” earlier in this chapter.

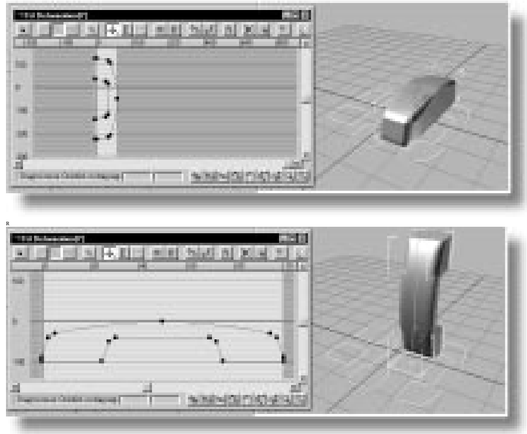
Resetting Fit Curves

The result of clicking Reset Curve is a little different for Fit Curves than it is for other deformation curves. Click Reset Curve to create a default rectangular Fit Curve.

A reset Fit Curve is a rectangle 100 units wide and centered on the path. If Make Symmetrical is on, then both Fit curves are reset even though only one might be displayed.

Rotating Fit Curves

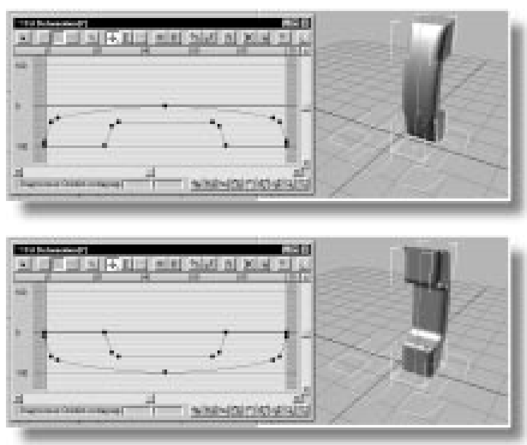
You can rotate a curve in the Fit Deformation window by clicking Rotate 90 CCW or Rotate 90 CW. Each time you click, the active curve is rotated 90 degrees.



Result of rotating a Fit curve

Mirroring Fit Curves

You mirror a curve in the Fit Deformation window by clicking Mirror Horizontally or Mirror Vertically. Each time you click, the active shape is flipped horizontally or vertically.



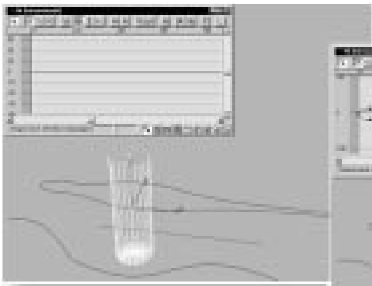
Result of mirroring a Fit curve

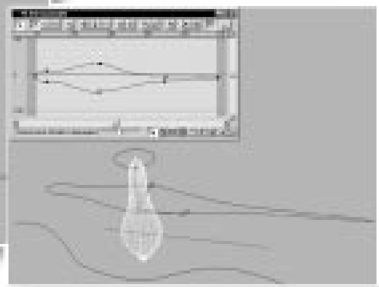
Deleting Fit Curves

Unlike the other deformation curves, you can delete one or both of the Fit X and Fit Y curves. When a Fit curve is deleted, Fit Deformation is ignored for that axis.

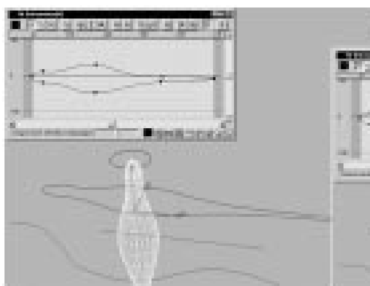
If Make Symmetrical is active then both Fit curves are deleted even though only one might be displayed.

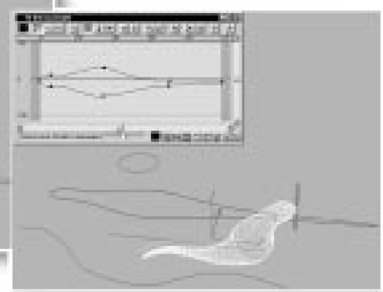
Generating a Path





Left: original loft
Right: skin after Get Shape





Left: skin after Generate Path
Right: skin after Get Path

You can generate a new loft path based on the size of the Fit curves. Clicking Generate Path replaces the current path with a straight path.

Until you click Generate Path, all shapes that you get as Fit curves are scaled along their local X axis to match the length of the current path. There is no requirement for you to replace the current path.

Generating a Path from a Fit Curve

Click Generate Path when you want to ensure that the loft object exactly matches the size of a Fit curve.

The length of the path created by Generate Path is set by either of the following:

- The length of the first shape placed in the Fit Deformation dialog using Get Shape.
- The length of the last shape placed in the Fit Deformation dialog while Make Symmetrical or Display XY Axes was on.

You can restore the original path after clicking Generate Path using one of two techniques:

- Click Undo to immediately undo the Generate Path command.
- Use Get Path to get a copy or instance of the original path from the scene.

Getting a Path After Getting Fit Shapes

You can use any path with Fit Deformation by clicking the Get Path button in the creation methods rollout. The Fit curves are rescaled to match the length of the new path.

Editing Loft Sub-Objects

Click Sub-Object at the base level on the Modifier Stack to edit the Loft Shapes and Loft Path. When Sub-Object mode is active you can do the following:

- Compare shapes on the path.
- Change shape orientation and position.
- Scale shapes on the path.
- Put a copy of any shape or the path to the scene as a Shape object.
- Access the Modifier Stack of any shape or the path and apply modifiers to them.
- Animate modifiers applied to the shapes and the path.

Putting Shapes and Paths to the Scene

You can clone a shape or path to the scene as an instance or independent copy. The result of using Put is a new shape object in your scene, lying on the ground plane and aligned with the world coordinate system.

Tip: Consider putting a copy of a shape or a path to the scene before you begin Sub-Object editing. Then you can retrieve your original shape or path by clicking Get Shape or Get Path.

Changing a Shape's Level on the Path

When you have a single shape selected you can use the Path Level field in the Shape Commands rollout to change the shape's level, moving it up and down the path.

You can clone a shape by pressing SHIFT while dragging the Path Level spinner.

Comparing Shapes

Use Compare to view two or more shapes in a 2D window and compare their positions with respect to each other and the path.

Click Pick Shape on the Compare window to choose which shapes to display.

The Compare window is useful for checking the relationship between shapes and the path.

- Shapes are displayed on a plane perpendicular to the path at each shape's level. If you have rotated a shape about its X axis or Y axis the shape might appear distorted.
- The first vertex of each shape is displayed as a small box on the shape.
- The path is displayed as a cross hairs marker. The path marker helps you compare shape relationships without being confused by bends or deformations in the path.

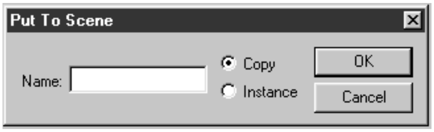
Aligning Shapes

Click one of the Align buttons to position selected shapes with their boundary, center, or pivot (Default button), located on the path.

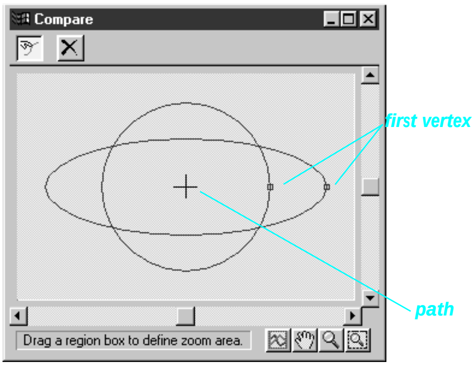
Note that you can use a combination of these buttons for corner alignment. If you click Left and then click Top, the upper-left corner of the shape's bounding box will be located on the path.

Deleting Shapes

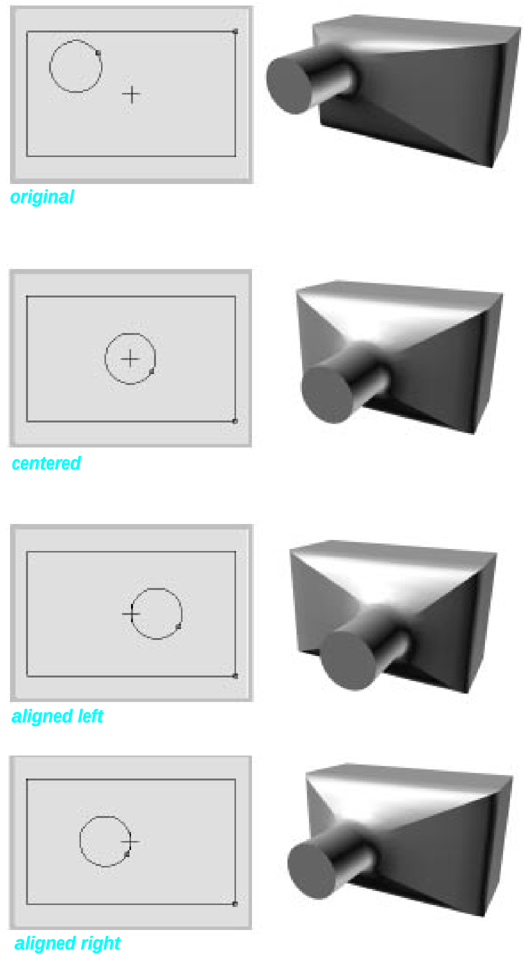
You can delete selected shapes from the path using the Delete button.



Put to Scene dialog

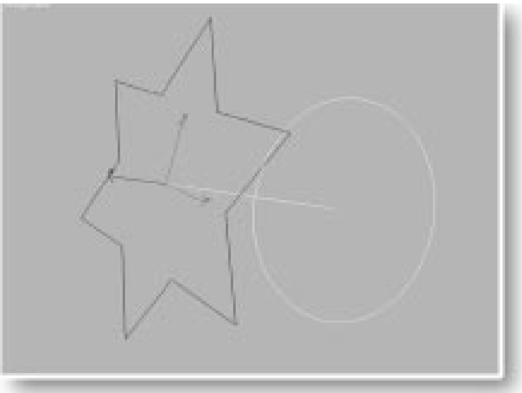


Displaying shapes in the Compare window

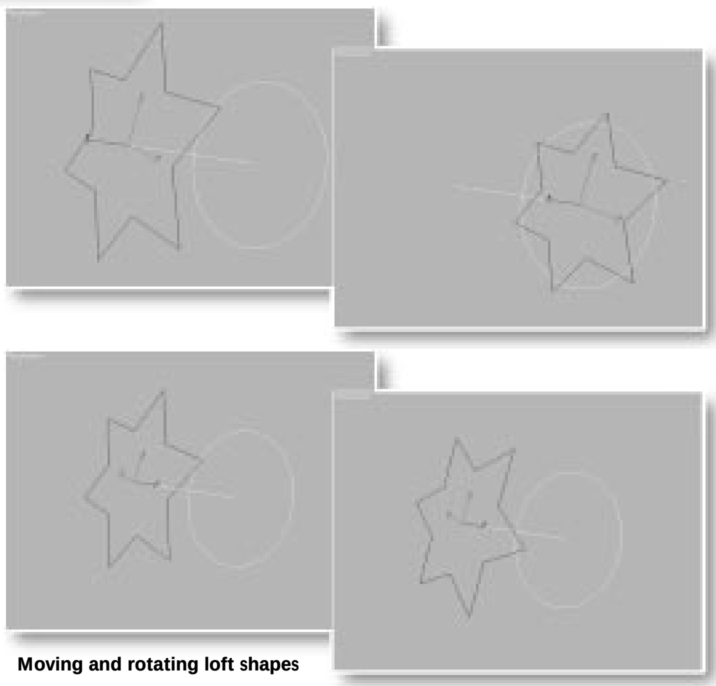


Aligning a shape

Transforming
Loft Shapes and Paths



Example of loft shape
local coordinate system



Moving and rotating loft shapes

Transforming shape sub-objects is much the same as transforming other objects in your scene. However, there are some special consider-ations to keep in mind.

Transform Coordinate System and Transform Center for Loft Shapes

When you transform shapes on a loft path, the transform coordinate system is locked to Local, and the transform center is locked to Pivot Points.

- The local coordinate system is defined as the local coordinate system of each shape after it has been placed on the path. This produces a coordinate system with the local Z axis tangent to the path and the XY plane perpendicular to the path.
- The transform center is defined as the point where the path intersects the XY plane of the shape's local coordinate system.
- When more than one shape is selected, each shape transforms about its own local coordinate system and center.

Moving, Rotating, and Scaling Loft Shapes

Because the transform coordinate system and transform center are locked, Loft Shape transforms exhibit the following characteristics:

- The center of rotation is always on the path.
- Moving a shape on the *Z* axis moves the shape up and down the path, like changing the shape's path level. The shape cannot move beyond the shapes preceding and following it on the path.
- The scale center is always on the path.

Transforms performed on shape objects in the scene do not affect instanced shapes on the path. Scaling, rotating, or moving the original shape object has no effect on its instance on the loft path. If you want transforms to affect both the original shape and the instance on the path, use an XForm modifier.

Resetting Shape Transforms

Click Reset to force one or more shapes back to their original orientation, and scale. Reset cancels any rotation or scale transforms you have applied to the loft shape.

Transforming Loft Paths

Rotate is the only transform you can perform on a path, and the transform managers are locked to the Local Z axis.

The path rotates using the fist vertex of the path as the center of rotation. The tangent to the path at the first vertex defines the axis of rotation (Z axis).

Aligning a Shape's First Vertex

The location of a shape's first vertex has great influence on how the skin of a loft is generated. 3D Studio MAX connects the first vertex of every shape on the path with a single seam.

If the first vertices of all the shapes in the path are not aligned, the loft skin twists from shape to shape. This effect can be particularly noticeable for lofts using open shapes.

To align the first vertices of your loft shapes, rotate the shapes until their first vertices align. The Compare window is a useful tool for watching how rotations effect the first vertex.

As you rotate the shape, you can judge how well the first vertices as follows:

- Display shapes in the Compare window and watch shape rotate.
- Turn on Skin Display and watch the effect on the loft skin in a viewport.

You can also rotate the shape at the Spline sub-object level.

Modifying and Animating Loft Sub-Objects

You can modify and animate loft shapes and loft paths by applying modifiers to the loft sub-object's Modifier Stack.

- Collapse loft sub-object shapes to Editable Splines to edit and animate shape and path sub-objects such as vertices and segments.
- Apply and animate modifiers, such as Bend Taper, and XForm, to loft sub-objects.

There are two ways of applying modifiers to loft sub-objects:

- Access the Loft Sub-Object Modifier Stack and apply a modifier to the sub-object.
- Apply a modifier to the instance of the sub-object in the scene.

You might find it easier to work with instance shapes in the scene than to work directly on the sub-objects in the loft. Consider using the Instance option with Get Shape and Get Path.

Changing and Animating the Sub-Object Parameters

If you access the sub-object's creation parameters, you can change and animate a loft's cross-section shapes or the path.

- Change the number of Sides for an NGon cross section shape to convert a loft from a three-sided to a five-sided prism.
- Animate the Height of a Helix path to simulate a spring being compressed.

Applying Modifiers to Loft Sub-Objects

Use geometric modifiers for such purposes as twisting individual loft shapes or bending the loft path. You can also animate the parameters of the modifier.

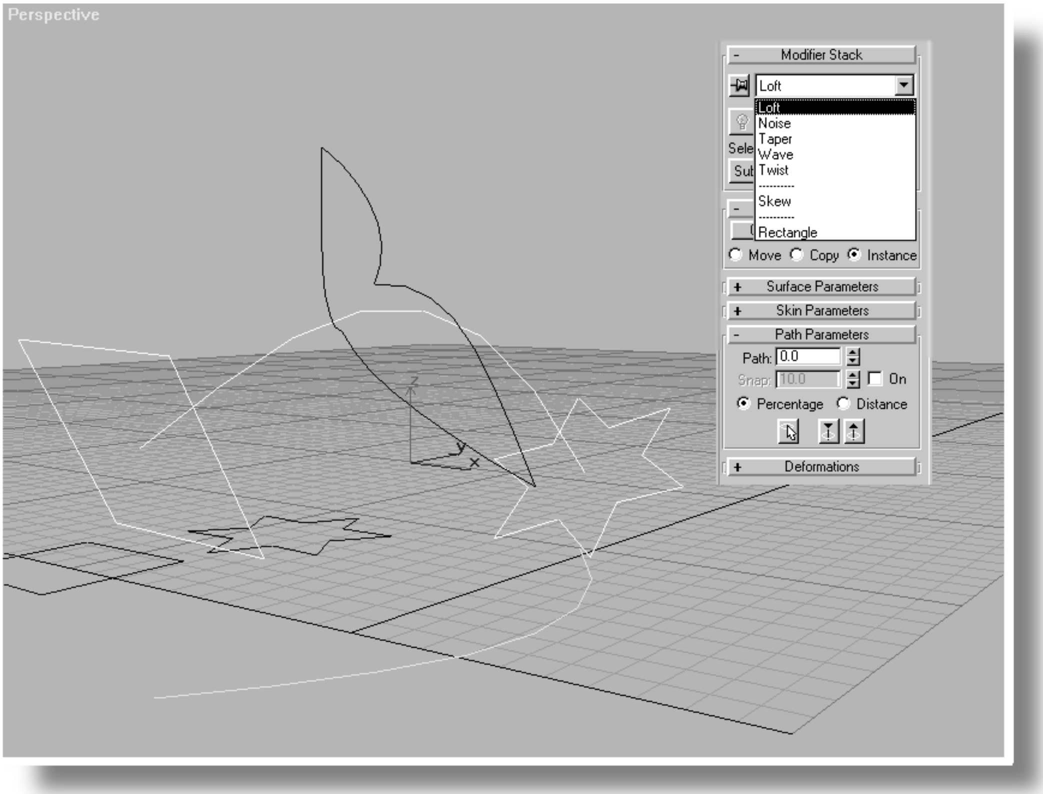
For example, apply and animate a Bend to a loft path to animate it bending back and forth. By applying the Bend to only the path, you can bend the loft without affecting the cross-section shapes. If you apply the Bend to the entire loft, the loft skin is affected, both along the path and around the cross section.

Accessing Editable Spline Properties of Loft Sub-Objects

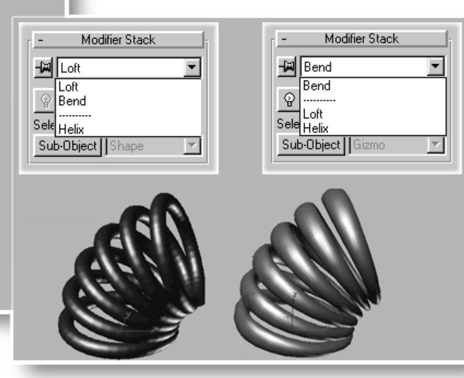
Probably the most common use for accessing the Loft Sub-Object Modifier Stack is to access the Editable Spline properties of Shapes and Paths. You can edit loft shapes and paths at their sub-object levels of spline, segment, and vertex.

Animating Transforms with the XForm Modifiers

You can also animate loft sub-object transforms by applying an XForm or Linked XForm modifier to a loft shape, and then animating transforms of the modifier's gizmo. See the Online Reference for details on using the XForm and Linked XForm modifiers.



Accessing the Loft Sub-Object Modifier Stack



Results of bending just the loft path versus bending the entire object

Chapter

10

Creating and Editing NURBS Models

3D Studio MAX Release 2 provides NURBS surfaces and curves. NURBS have become an industry standard for designing and modeling surfaces. They are especially suited for modeling surfaces with complicated curves. NURBS stands for Non-Uniform Rational B-Splines. The tools for modeling with NURBS do not require an understanding of the mathematics that produces these objects. NURBS are popular precisely because they are easy to manipulate interactively, and because the algorithms that create them are both efficient and numerically stable.

In 3DS MAX, you can also model surfaces using polygonal meshes or patches. Compared to NURBS surfaces, meshes and patches have these shortcomings:

- It is more difficult to create complicated curved surfaces using polygons.
- Because meshes are faceted, facets appear at the edge of rendered objects. You must have a large number of small faces to render a smoothly curved edge.

NURBS surfaces, on the other hand, are analytically generated. They are more efficient to calculate, and you can render a NURBS surface that appears to be seamless. (A rendered NURBS surface is actually approximated by polygons, but the NURBS approximation can be very fine grained.)

NURBS Models: Objects and Sub-Objects

Like 3DS MAX Shape objects, a NURBS model can be an assemblage of multiple NURBS sub-objects. For example, a NURBS object might contain two surfaces that are separate in space. Both NURBS curves and NURBS surfaces are controlled by either point or control vertex (CV) sub-objects. Points and CVs behave somewhat like the vertices of spline objects in 3DS MAX, but there are differences.

The parent object in a NURBS model is a NURBS surface. Sub-objects can be any of the objects listed here:

- *Surfaces*. There are two kinds of NURBS surfaces in 3DS MAX. A *Point Surface* is controlled by points, which always lie on the surface. A *CV Surface* is controlled by control vertices (CVs). Instead of lying on the surface, CVs form a control lattice that surrounds the surface.

- *Curves*. There are also two kinds of NURBS curves in 3DS MAX. These correspond exactly to the two kinds of surfaces. A *Point Curve* is controlled by points, which always lie on the curve. A *CV Curve* is controlled by control vertices (CVs), which don't necessarily lie on the curve.

- *Points*. Point Surfaces and Point Curves have point sub-objects. You can also create separate point sub-objects that are not part of a surface or a curve.

- *CVs*. CV Surfaces and CV Curves have CV sub-objects. Unlike points, CVs are always part of a surface or a curve.

- *Imports*. Imports are 3DS MAX objects—including other NURBS objects. Within the NURBS model, they render as NURBS; but they retain their original parameters and modifiers.

Creating NURBS Models

3DS MAX provides a variety of ways to create NURBS surfaces. This is a summary of how you create top-level, parent NURBS objects:

- Create NURBS curves in the Shape panel of the Create command panel.

  Note: NURBS curves can have curve and point sub-objects, but they can't have surface sub-objects.

- Create NURBS surfaces in the Geometry panel of the Create command panel. When you use this technique, the NURBS surface is initially a flat rectangle. You can alter it using the Modify command panel.

- Turn a 3DS MAX geometry primitive into a NURBS object by using the Edit Stack button in the Modify command panel.

  The primitive objects you can convert to NURBS objects are the Standard Primitives, not the Extended Primitives.

Working with NURBS Models

When you work with NURBS models, usually you follow these general steps:

- Create one NURBS object as the "starter" object. This can be either a surface object, or a converted geometry primitive.

  Often modelers like to identify a single, master surface as the main component of the model. Converted geometry primitives are good if you want the starter surface to become the master surface. Point and CV surfaces are good as starters for rectangular surfaces.

- Go to the Modify panel. Edit the original object, or create additional sub-objects as described in the following section.

  You might even choose to delete the original, starter object once you have built a model from newer sub-objects.

Going immediately into the Modify panel avoids the problem of creating additional top-level NURBS objects, which you can't use to build relational, dependent sub-objects. See the sections that follow.

Two general references for modeling with NURBS are *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Guide* by Gerald Farin (Academic Press, fourth edition 1996), and *Interactive Curves and Surfaces: A Multimedia Tutorial on Computer Aided Graphic Design* by Alyn Rockwood and Peter Chambers (Morgan Kaufman Publishers 1996).

Modifying NURBS Models and Creating Sub-Objects

NURBS are immediately editable on the Modify panel. You don't have to apply a modifier, as you do for most kinds of 3DS MAX objects.

While you are editing a NURBS object in the Modify panel, you can create sub-objects "on the fly," without having to go back to the Create panel. This is an exception to the way you usually use 3DS MAX. The Modify panel for NURBS curve and NURBS surface objects includes rollouts where you can create new NURBS sub-objects.

This is a summary of how to create sub-objects:

- Use the curves and surfaces have a rollout for creating points. A point you create individually is either an independent point or a dependent point tied to other NURBS geometry.

- Both curves and surfaces have a rollout for creating curves. Curve sub-objects are either independent Point Curves or CV Curves, or they are *dependent* curves whose geometry is based on other curves already present in the model. For example, a Blend curve is a dependent curve sub-object that connects the endpoints of two other curves.

- Surfaces have a rollout for creating surfaces. Surface sub-objects are either independent Point Surfaces or CV Surfaces, or they are dependent surfaces whose geometry is based on other surfaces already present in the model. For example, a Blend surface is a dependent surface sub-object that connects the edges of two other surfaces.

- You can attach 3DS MAX objects. If the attached object is not already a NURBS object, it is converted to NURBS geometry. A NURBS curve can attach another NURBS curve or a spline curve. A NURBS surface can attach a curve, another NURBS surface, or a convertible 3DS MAX object. The attached object becomes one or more curve or surface sub-objects.
- You can import 3DS MAX objects. The attached object retains its parameters. It renders as a NURBS sub-object, but you can still edit it parametrically at the Imports sub-object level. At this sub-object level, viewports display its usual geometry, not its NURBS form. A NURBS curve can import NURBS curves or spline curves. A NURBS surface can import curves, surfaces, or convertible 3DS MAX objects.

Note: You can detach a NURBS sub-object to make it a new NURBS object of its own, and you can extract an imported object to create an independent, top-level object once again.

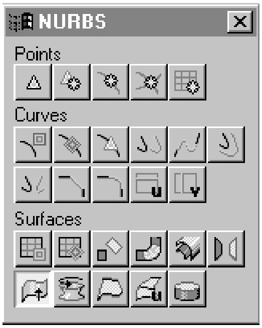
Using the NURBS Toolbox to Create Sub-Objects

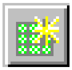
Besides using the rollouts at the NURBS object level, you can use the NURBS toolbox to create sub-objects.

To see the toolbox:

1.  With the NURBS object selected, go to the Modify panel's Display area.

2. Turn on NURBS Creation Toolbox.





NURBS creation toolbox

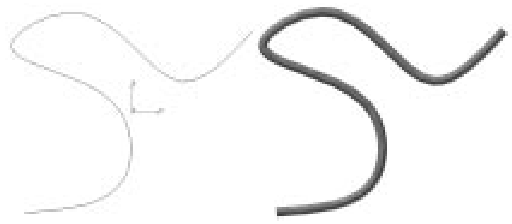
The toolbox contains buttons for creating NURBS sub-objects. From a curve object, you can create points or other curves. From a surface object, you can create points, curves, or other surfaces. In general, the toolbox behaves like this:

- While the toolbox button is turned on, the toolbox is visible whenever a NURBS object or sub-object is selected and you are in the Modify panel. It disappears whenever you deselect the NURBS object or make a different panel active. When you return to the Modify panel and select a NURBS object, it reappears.
- You can use the toolbox to create sub-objects either at the top, object level, or at any NURBS sub-object level.
- When you turn on a toolbox button, you go into creation mode, and the Modify panel changes to show the parameters (if there are any) for the kind of sub-object you are creating.
- If you are at the top, object level and use the toolbox to create an object, you must then turn on Sub-Object to edit the new sub-object. (This is no different from using the buttons in the rollouts.)

- If you are at a sub-object level and use the toolbox to create an object of the same sub-object type, you can edit it immediately after you turn off the create button (or right-click).
- If you are at a sub-object level and use the toolbox to create an object of a different sub-object type, you must use the list to change the sub-object level before you can edit the new sub-object.

The individual creation buttons are described by their tool tips, and in the online reference.

NURBS Curves

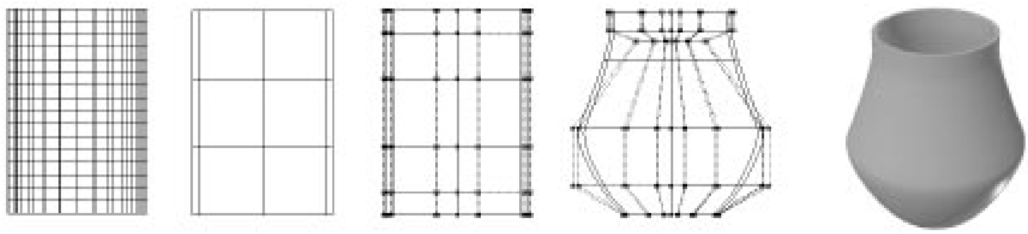


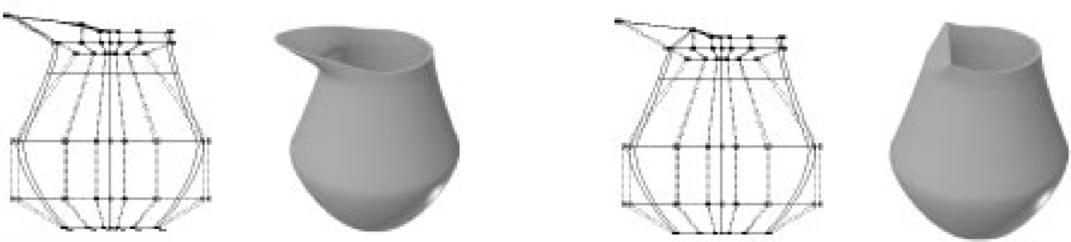
NURBS curves are Shape objects, and you can use them as you do splines. You can use the Extrude or Lathe modifiers to generate a 3D surface based on a NURBS curve. You can use NURBS curves as the path or the shape of a Loft. (Lofts created using NURBS curves are loft objects, not NURBS objects.)

You can also use NURBS curves as Path Controller paths or motion trajectories.

You can give a NURBS curve a thickness so it renders as a cylindrical object. (The thickened curve renders as a polygonal mesh, not as a NURBS surface.)

CV Curves and CV Surfaces





CV Curves and CV Surfaces have control vertices as splines do. The position of the control vertices (CVs) controls the shape of the curve or the surface. However, unlike spline vertices, CVs don't necessarily lie on the curve or surface they define. The CVs define a control lattice. The control lattice connects the CVs. 3DS MAX displays it in dashed yellow lines. The control lattice surrounds the NURBS curve or surface.

Tip: When you use Zoom Extents, 3DS MAX displays the entire extents of a NURBS object, including its control lattice. Because a CV can be at some distance from an object, the curve or surface itself (the object's renderable geometry) is sometimes hard to see. Use Zoom Region or Field of View to zoom in.

You can move a CV at the Curve CV or Surface CV sub-object level in the Modify command panel. Other transforms—rotate and scale—work as well. Rotate and scale are especially useful when you select multiple CVs.

Each CV also has a *weight*, which you can use to adjust the CV's effect on the curve or surface. Increasing the weight pulls the surface towards the CV. Decreasing the weight relaxes the surface away from the CV.

Changing a surface by increasing the weight of one CV

Weights can be a useful way to "tune" the appearance of a NURBS curve or surface.

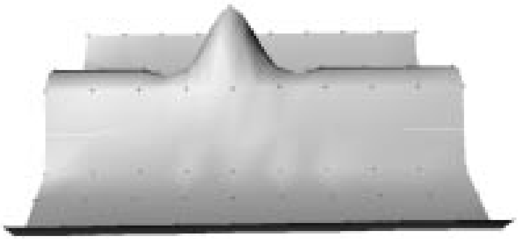
The weight value of a CV is rational. That is, it is relative to other CVs in the curve or surface. Changing the weight of all CVs at once has no effect, because this doesn't change the ratio between weights.

Points, Point Curves, and Point Surfaces

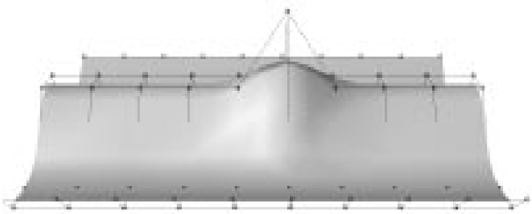
Point Curves and Point Surfaces are similar to CV Curves and Surfaces, but the points that control them are required to lie on the curve or surface. Unlike CVs, points do not have a weight.

Point Curves and Point Surfaces can be more intuitive to create and work with. However, they can give you unexpected results because more than one NURBS solution is possible for a given set of NURBS points. This is not true of CV Curves and Surfaces. You can think of a Point Curve or Point Surface as being dependent on the points to which it fits.

Points that you create individually are the same as the points on Point Curves and Surfaces, except that they aren't (initially) part of a curve or surface. You can create a Point Curve by fitting it to points that you select. You can use both points that are part of curves or surfaces, and individual point sub-objects, when you fit the new Point Curve.



Point Surface

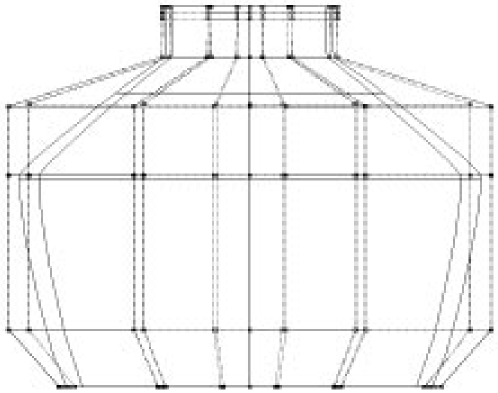


CV Surface

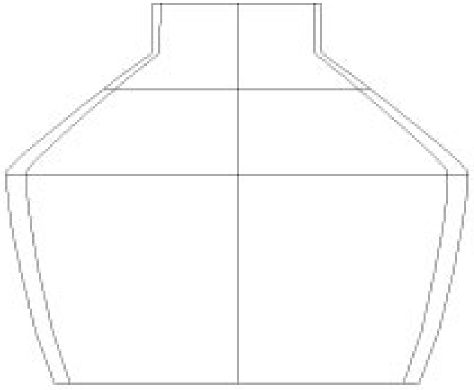
Sub-Object Selection

When you work with NURBS models, you often work with sub-objects. While you are at the sub-object level, you use the usual 3DS MAX selection techniques, such as clicking or dragging a region, or holding down CTRL, to choose one or more sub-objects. You can also select NURBS sub-objects by name. Turn on the Plug-In Keyboard Shortcut Toggle button in the status bar, and then press the “H” key. This displays a select-by-name dialog that lists only sub-objects at the current level. Choose one or more objects in the list, and then click Select. This dialog is especially useful at the Curve and Surface sub-object levels. The large number of CVs or points in a complicated curve or surface makes them harder to distinguish by name—but you can assign your own names to NURBS sub-objects you want to edit frequently.

Tip: When you work with NURBS, often you switch frequently between the object and sub-object levels, or from one sub-object level to another. Two keyboard shortcuts can help you do this. The Sub-Object Selection Toggle (default = CTRL + B) switches between object and sub-object levels. The Cycle Sub-Object Level shortcut (default = INSERT) switches from one sub-object level to another. Also, the command panel’s right-click popup menu (available whenever the pan hand is visible) helps you navigate the rollouts in the current command panel.

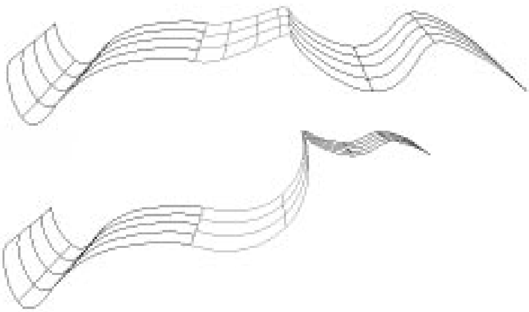


CV Subobjects



Surface Subobject

Dependent Sub-Objects



Moving a parent surface changes the blend surface

NURBS sub-objects are either independent or dependent. A dependent sub-object is one based on the geometry of other sub-objects. For example, a Blend surface smoothly connects two other surfaces. Transforming—or animating—either of the original, parent surfaces causes the shape of the Blend to change, as it maintains a connection between the parents

This immediate, interactive relation between the parent and dependent sub-objects is known as *relational modeling*. Relational modeling is one of the reasons NURBS models can be particularly easy to change or to animate.

Important: Dependent sub-objects must have parents that are sub-objects of the same NURBS model. Dependent relationships can't exist between object-level NURBS curves or surfaces. If you want to use a top-level NURBS object to create a dependent object, you must first attach or import the top-level object.

You have the option of making a dependent sub-object independent. After you do so, the sub-object is no longer related to its parents. Changes to the former parents don't affect it, but you can directly edit and transform it as an independent sub-object in its own right.

At the appropriate sub-object level, dependent NURBS are displayed in green in wireframe viewports. (You can change the display color using the Colors panel of the Edit/Preferences dialog.)

Sometimes changes you make to the parent objects make it no longer possible to correctly update the dependent object's geometry. For example, a Fillet between two curves requires the curves to be coplanar. If you move one curve (or its CVs or points) so that the curves are no longer coplanar, 3DS MAX cannot correctly update the Fillet. In this case, the dependent object's geometry reverts to a default position, and 3DS MAX displays it in orange to indicate that it has an error condition. (You can change the error color using the Colors panel of the Edit/Preferences dialog.)

Transforming Dependent Sub-Objects

In general, you can select and transform dependent sub-objects, but the effect of the transform depends on the sub-object type. Some dependent objects have a gizmo, similar to the gizmo used with modifiers. Sub-objects that don't have gizmos can't change relative to their parent objects. For these kinds of sub-objects, transforms apply equally to the sub-object and its parents. For example, moving a blend sub-object moves its parents as well. Sub-objects that have gizmos can change relative to their parent objects. In this case, as with modifiers that use gizmos, you are really transforming the gizmo. For example, rotating a mirror sub-object changes the mirror axis, and therefore the mirror's position relative to the parent curve or surface.

NURBS and Modifiers

In general, you can apply modifiers to NURBS models as you do to other 3DS MAX objects. The exceptions are the edit modifiers that apply to meshes and patch surfaces. These don't apply to NURBS objects, and are disabled when a NURBS object is selected.

Deforming NURBS Objects

Deform modifiers such as Bend and Twist operate on CV and point sub-objects. They don't change the NURBS model into an editable mesh object. This means that you can use a deform modifier, collapse the stack, and still have a NURBS object that you can edit further. However, because the deform modifiers directly affect CVs and points, not the mesh approximation of the NURBS model, they can produce unexpected results. For example, a Ripple modifier does not ripple the surface if the CVs are farther apart than the wavelength of the ripples. If you want the modifier to affect the mesh approximation instead of the CVs, you can apply a Mesh Select modifier first. Then when you collapse the stack, you get an editable mesh, not a NURBS object.

NURBS Objects and the UVW Map Modifier

When you apply a UVW Map modifier, it affects the NURBS object the same way it affects a mesh. If you then collapse the stack, the modifier is still in effect. However, you can override the mapper for individual surface sub-objects. To do so, click to select the surface's Generate Mapping Coords. check box. When the check box is selected, you get the natural mapping of the surface; when it is cleared, you get the mapping from the collapsed UVW modifier.

The NURBS Select Modifiers

Two modifiers, NCurve Sel and NSurf Sel, let you place a NURBS sub-object selection on the modifier stack. This lets you modify only the selected sub-objects. Also, selected curve sub-objects are Shape objects that you can use as paths and motion trajectories.

NSurf Sel can select any kind of NURBS sub-objects except imports. NCurve Sel selects only objects available to top-level curve objects. In other words, it can select any kind of NURBS sub-objects except surfaces and imports. Each sub-object selection is of one particular sub-object level only.

To use a NURBS select modifier:

1. With a NURBS object selected in the Modify command panel, apply either NCurve Sel or NSurf Sel.

   The select modifiers have no controls at the object level.

2. Click to turn on Sub-Object in the Modifier Stack rollout, and choose a sub-object level from the list.

   The select modifiers have the same selection controls you see for the corresponding sub-object type, except that selecting connected curves or surfaces is not available.

   While applying the modifier, you can also select NURBS sub-objects by name. Turn on the Plug-In Keyboard Shortcut Toggle button in the status bar, and then press the "H" key. This displays a select-by-name dialog that lists only sub-objects at the current level. Choose one or more objects in the list, and then click Select.

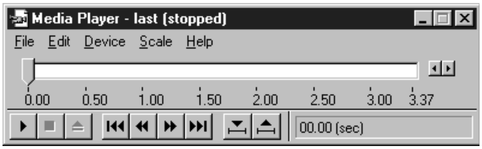
3. Use the selection controls to create a selection set of the chosen sub-object type.

   With the NSurf Sel modifier, the selection can be of any NURBS sub-object level except imports. With NCurve Sel, the selection can be of any sub-object level except surfaces and imports.

Once you have used the modifier to create the selection, you can apply other modifiers to it. If the selected sub-object is a curve, you can also use it as a path or trajectory.

Note: The NURBS select modifiers don't support copy and paste of selections as Mesh Select does. Copying and pasting mesh selections is based on vertex indexes. NURBS selections are based on object IDs, which are unique to each model.

NURBS and Animation



In general, you can animate NURBS Curves and NURBS Surface by turning on the Animate button and transforming sub-object attributes such as CV or point positions, by animating the parameters that control dependent NURBS objects, and so on. You can't animate NURBS object creation or creation parameters, or fundamental changes to NURBS geometry such as adding or deleting CVs or points, attaching objects, and so on.

Some NURBS editing operations remove animation controllers.

Operations that Remove Animation

In general, the following operations remove animation from a NURBS object or sub-object:

- Make Independent

  Removes the animation of anything directly dependent on the object.

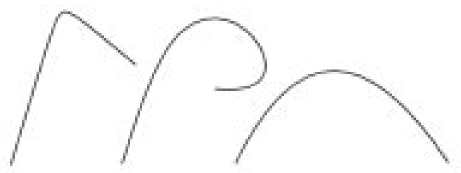
- Break, Extend, Join, Refine, and Delete

  Any operation that changes the number of points or CVs in a curve or surface loses the animation on all points or CVs to be lost.

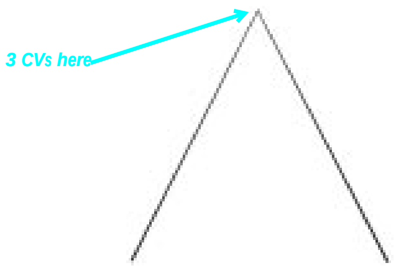
- Fuse

  The animation of the point or CV being fused to the other point or CV (the second one chosen) is lost. The first point or CV acquires the animation of the second.

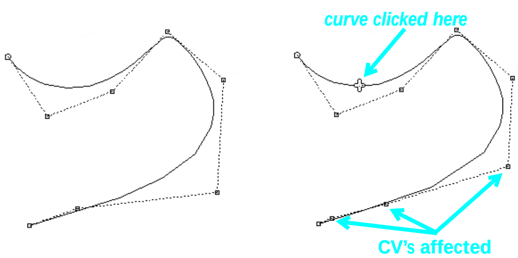
NURBS Concepts



Levels of curve continuity



Effects of multiplicity



Refining a curve moves its CVs

NURBS curves and surfaces did not exist in the conventional drafting world. They were created specifically for 3D modeling using computers. Curves and surfaces represent contours or shapes within a 3D modeling space. They are constructed mathematically. NURBS mathematics is complex, and this section is simply an introduction to some NURBS concepts that might help you understand what you are creating, and why NURBS objects behave as they do. For a comprehensive description of the mathematics and algorithms involved in NURBS modeling, see *The NURBS Book* by Les Piegl and Wayne Tiller (New York: Springer, second edition 1997).

The term NURBS stands for Non-Uniform Rational B-Splines. Specifically:

- A *B-spline* (for basis spline) is a way to construct a curve that is interpolated between three or more points.

  Shape curves you create in 3DS MAX using the Line tool and other Shape tools are Bézier curves, which are a special case of B-splines.

- *Rational* means that the equation used to represent the curve or surface is expressed as a ratio of two polynomials, rather than a single summed polynomial. The rational equation provides a better model of some important curves and surfaces, especially conic sections, cones, spheres, and so on.

- *Non-Uniform* means that the extent of a control vertex's influence can vary. This is useful when modeling irregular surfaces. Also, uniform curves and surfaces aren't invariant under perspective projection—a serious limitation for interactive 3D modeling.

The non-uniform property of NURBS brings up an important point. Because they are generated mathematically, NURBS objects have a *parameter space* in addition to the 3D geometric space in which they are displayed. Specifically, an array of values called *knots* specifies the extent of influence of each control vertex (CV) on the curve or surface. Knots are invisible in 3D space and you can't manipulate them directly, but occasionally their behavior affects the visible appearance of the NURBS object. This section mentions those situations. Parameter space is one-dimensional for curves, which have only a single *U* dimension topologically, even though they geometrically exist in 3D space. Surfaces have two dimensions in parameter space, called *U* and *V*.

NURBS curves and surfaces have the important properties of not changing under the standard geometric affine transformations (Transforms in 3DS MAX), or under perspective projections. The CVs have local control of the object—moving a CV or changing its weight does not affect any part of the object beyond the neighboring CVs. (In 3DS MAX, you can override this property by using the Affect Region controls.) Also, the control lattice that connects CVs surrounds the surface. This is known as the *convex hull* property.

All curves have a *degree*. The degree of a curve is the highest exponent in the equation used to represent it. A linear equation is degree 1, a quadratic equation degree 2. NURBS curves typically are represented by cubic equations and have a degree of 3. Higher degrees are possible but usually unnecessary.

Curves also have *continuity*. A continuous curve is unbroken. There are different levels of continuity. A curve with an angle or cusp in it is C^0 continuous—that is, the curve is continuous but has no derivative at the cusp. A curve with no such cusps but whose curvature changes is C^1 continuous. Its derivative is also continuous, but its second derivative is not. A curve with uninterrupted, unchanging curvature is C^2 continuous. Both its first and second derivatives are also continuous. A curve can have still higher levels of continuity, but for computer modeling these three are adequate. Usually the eye can't distinguish between a C^2 continuous curve and one with higher continuity.

Continuity and degree are related. A degree 3 equation can generate a C^2 continuous curve. This is why higher-degree curves aren't generally needed in NURBS modeling. Higher-degree curves are also less stable numerically, so using them isn't recommended.

Different segments of a NURBS curve can have different levels of continuity. In particular, by placing CVs at the same location or very close together, you reduce the continuity level. Two coincident CVs sharpen the curvature. Three coincident CVs create an angular cusp in the curve. This property of NURBS curves is known as *multiplicity*. In effect, the additional one or two CVs combine their influence in that vicinity of the curve. By moving one of the CVs away from the other, you increase the curve's continuity level again. In 3DS MAX, multiplicity also applies when you fuse CVs. Fused CVs create a sharper curvature or a cusp in the curve. Again, the effect goes away if you unfuse the CVs and move one away from the other.

Refining a NURBS curve means to add more CVs. Refining gives you finer control over the shape of the curve. When you refine a NURBS curve in 3DS MAX, the software preserves the original curvature. (Technically speaking, it maintains a uniform knot vector.) In other words, the shape of the curve doesn't change, but the neighboring CVs *move away from* the CV you add. This is because of multiplicity—if the neighboring CVs didn't move, the increased presence of CVs would sharpen the curve. With the 3DS MAX interface, you first refine the curve, then change it by transforming the newly added CVs or adjusting their weights. NURBS surfaces have essentially the same properties as NURBS curves, extended from a one-dimensional parameter space to two dimensions.

In 3DS MAX you can work with point curves and point surfaces as well as CV curves and surfaces. The points that control these objects are constrained to lie on the curve or surface. There is no control lattice, and no weight control. This is a simpler interface that some users find easier to work with. Also, point-based objects give you the ability to construct curves based on dependent (constrained) points, and then use these to construct dependent surfaces.

You think of point curves and surfaces as an interface to CV curves and surfaces, which are the fully defined NURBS objects. The underlying representation of the curve or surface is still constructed using CVs. There can be more than one CV solution for a point curve or surface—because of this, occasionally 3DS MAX might construct a point curve or surface in a way you didn't expect.

You can also think of a point curve or surface as dependent on its points. You can use the Make Independent button to convert a point curve or surface to the CV form. On the other hand, you can't convert from the CV to the point form, because there are multiple point solutions for a single CV curve or surface.

Chapter

11

Creating Patch Surfaces

Using the Edit Patch modifier in 3D Studio MAX, you have very fine control over surface modeling operations. The modifier converts the surface of an object into a collection of Bezier patches surrounded by a control lattice. By transforming components of the lattice, you reshape the three-dimensional surface. Patch modeling is particularly good for sculpting smoothly curved and irregular shapes.

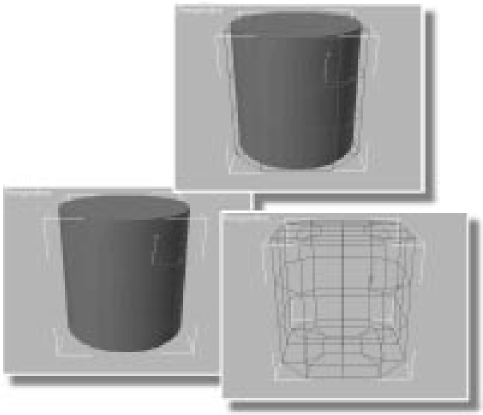
With Edit Patch, you can work at any of four levels:

- Object
- Patch
- Edge
- Vertex

Each level provides additional features to aid in modeling. Once you're satisfied with the general shape, you can convert an Edit Patch object into an editable mesh for further work.

Note: NURBS surfaces are an alternative to Edit Patch. They are recommended for more complex surface modeling. See chapter 10, "Creating NURBS Models."

Introducing Edit Patch



Lattice and surface of a patch (top)
Surface only (left) and lattice only (right)

You use the Edit Patch modifier to convert an object into a *patch*. The surface becomes a collection of Bezier patches surrounded by a control lattice. By transforming components of the lattice, you reshape the three-dimensional surface. Patch modeling is particularly good for sculpting gently curved and elongated shapes.

This chapter introduces the underlying concepts, controls, and use of the Edit Patch modifier. See online reference for specific procedures.

Comparing Edit Patch to NURBS

Edit Patch retains compatibility with 3D Studio MAX Release 1. NURBS surfaces, new to Release 2, far surpass the functionality offered by Edit Patch for modeling complex and delicately curved surfaces.

- If you want a quick solution to producing a gentle bulge in geometry, or if you're already familiar with the modifier, Edit Patch remains a viable tool.
- If you plan to spend hours or days on a curvilinear model, consider learning and using NURBS. See chapter 10, "Creating NURBS Models," for an introduction to NURBS features.

Bezier Patch Structure

When you apply an Edit Patch modifier to an object, 3DS MAX converts the object's geometry into a collection of separate Bezier patches. Each patch is made up of a *lattice* and a *surface*.

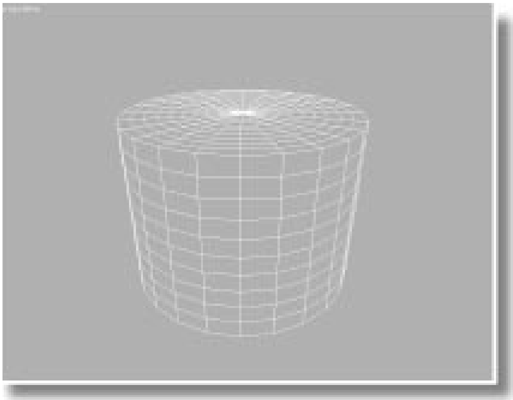
Lattice—The framework of control points and connecting tangents that surrounds the surface. The lattice does not appear in scanline renderings.

Transforming components of this lattice is the primary modeling technique.

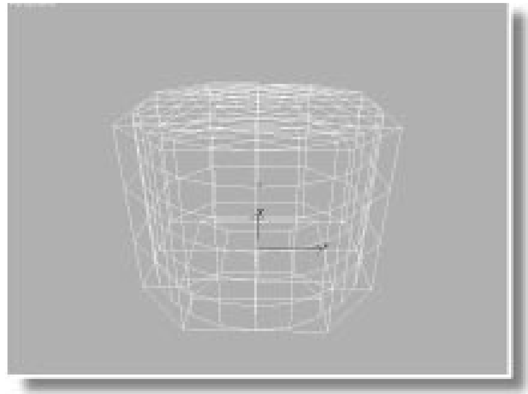
Surface—The Bezier patch surface whose shape is controlled by the lattice. The surface is the renderable geometry of the object.

- For flat-surfaced objects like boxes, the lattice and surface show no separation in the viewport.
- For round and irregular surfaces, the lattice is visibly separate from the surface.

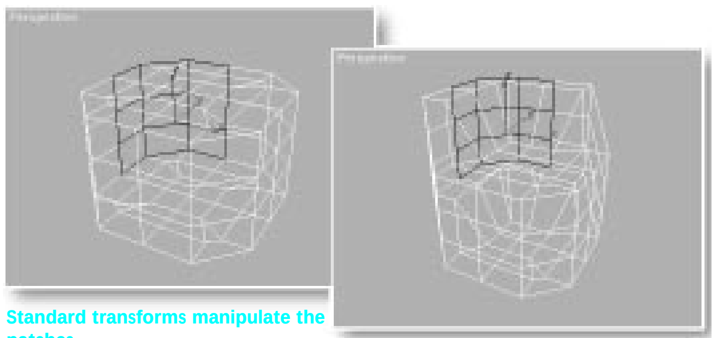
The figures on the opposite page illustrate the general use of this modifier.



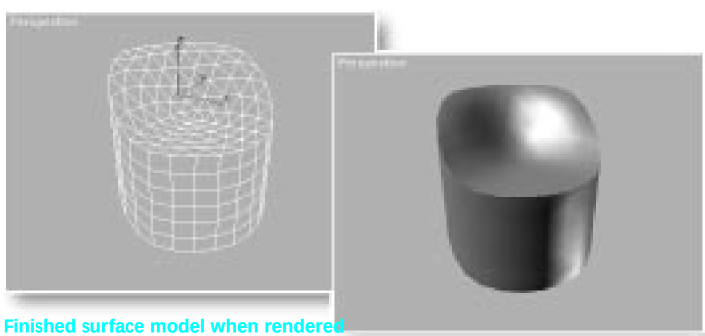
Original geometric primitive



Edit Patch applied and set to patch level

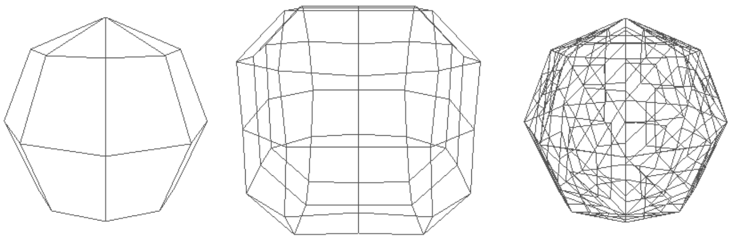


Standard transforms manipulate the patches

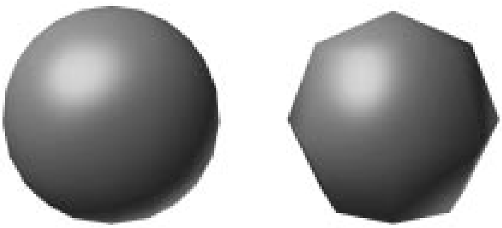


Finished surface model when rendered

Applying and Using Edit Patch



Left: Original primitive
Middle: Edit Patch applied to primitive (lattice only shown)
Right: Primitive converted to an editable mesh before applying Edit Patch (lattice only shown)



Resulting surfaces when rendered

Like other modifiers, Edit Patch is added to an object's modifier stack. You begin by selecting an object.

- Click Edit Patch on the Modifiers rollout of the Modify panel. The Sub-Object button turns on, and the Edit Vertex rollout appears.

To work at sub-object level:

- Choose a selection level: Patch, Edge, or Vertex. The corresponding rollout appears. Select the geometry you want to edit.

To work at whole-object level:

- Click the Sub-Object button to turn it *off*. The Edit Object rollout appears.

Using Edit Patch

After applying Edit Patch, you have these options to reshape and model the surface of the object:

- Move, rotate, or scale the sub-object selection.
- Use options on the current rollout to modify the selection, such as Weld on the Edit Vertex rollout.
- Apply available modifiers to the selection, such as a Bend on a number of edges.

After you complete patch editing, these are some of the possible options:

- Use available modifiers to continue refining the geometry.
- Use the patch as a "pick object" for the object-space or world-space Patch Deform modifier. These modifiers use the contours of the patch as a template to deform another object. See online reference for procedures.
- Convert to an editable mesh by first collapsing the modifier stack to produce a patch object. Then select the patch object and convert it to an editable mesh. See chapter 12, "Editing Meshes."

Using Edit Patch with Other Modifiers

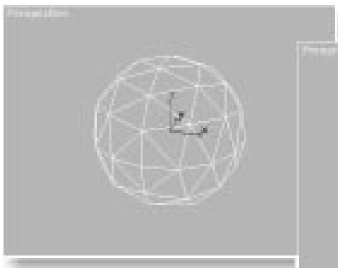
Edit Patch acts as a self-contained modeling environment. The results of the modeling can then be passed on to geometric modifiers, or to an editable mesh. These are some general guidelines:

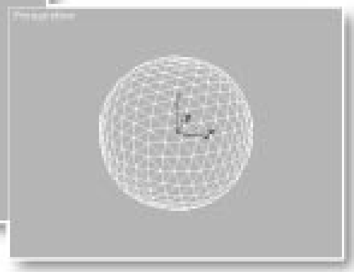
- Apply Edit Patch to a primitive object as the *first* modifier on the object's stack.
- Convert an Edit Patch object to an editable mesh only after you have completed patch modeling.

Any geometric modifier or an editable mesh creates a *mesh* representation of an object. This is not usually what you want with patch modeling. Here's what happens:

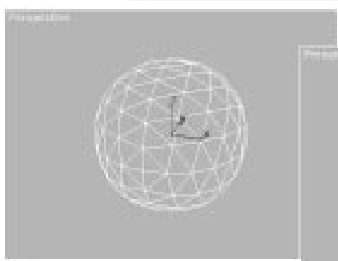
- When you apply Edit Patch to a mesh, each triangular face of the mesh object becomes a *separate* patch.
- With the resulting large number of patches, modeling becomes harder and less productive. The advantage of a few large patches to control the surface is lost.
- The underlying mesh restricts the smooth reshaping possible with patch modeling.

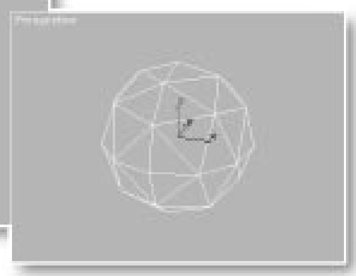
Editing at Object Level



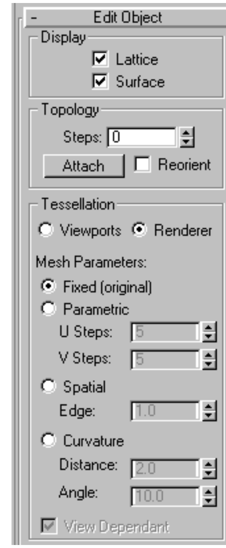


Effect of increasing the number of steps





Effect of decreasing the number of steps



At the Object level of Edit Patch, you are working with the whole object. From this level, you can adjust patch density and attach other objects to the patch objects.

When you apply Edit Patch, sub-object selection is turned on by default.

- To reach the Object level of the modifier, turn off the Sub-Object button. The Edit Object rollout appears.

Adjusting Patch Density

As you edit an object with Edit Patch, you might need a more complex surface geometry to work with. This adjustment affects only the surface, not the lattice.

You make the adjustment by increasing the number of divisions in the geometry. Tessellation increases over the entire surface. You can also decrease the tessellation if the geometry is more complex than you need.

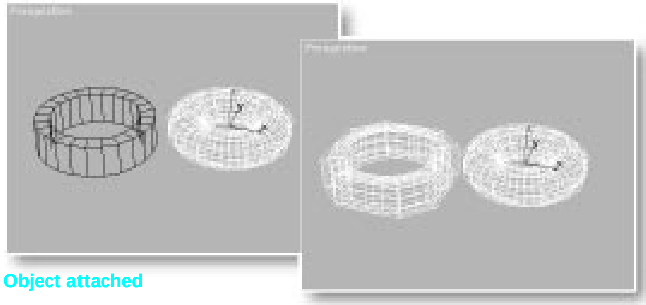
- To adjust patch density, display only the surface of the patch by clearing Lattice.
- Increase or decrease the number of steps. Default=5.

The illustrations use a sphere to illustrate the effects of increasing and decreasing surface density. In these figures, the lattice is turned off.

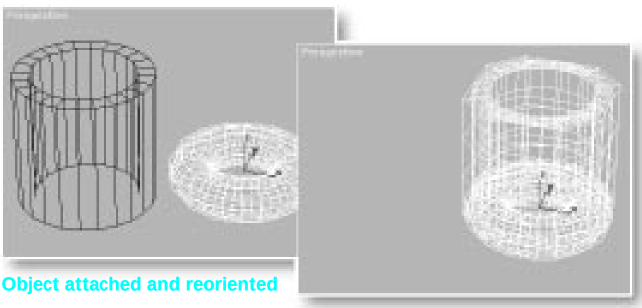
Attaching Objects with Edit Patch

At the Object level, you can attach any geometric object to an Edit Patch object, and optionally align (or *reorient*) the attached object as well.

The process converts the attached object to a patch surface, removing any stack and animation data, but keeping its topology.



Object attached



Object attached and reoriented

- If you attach a parametric object or another patch object, the resulting patch surface has the same number of divisions as the Edit Patch object.
- If you attach a mesh object, each face becomes a separate patch.

To attach an object:

1. On the Edit Object rollout, click Attach.

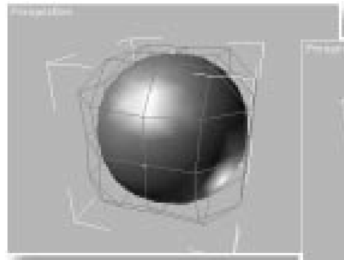
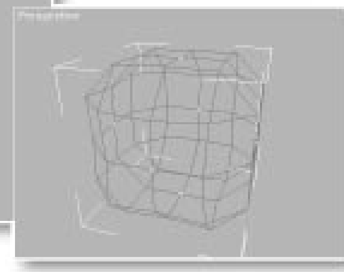
2. Either clear Reorient to leave the attached object in place, or set Reorient to move the attached object into alignment with the patch object.

3. Select an object to attach.

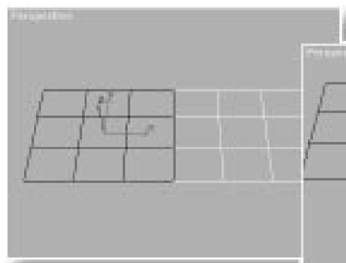
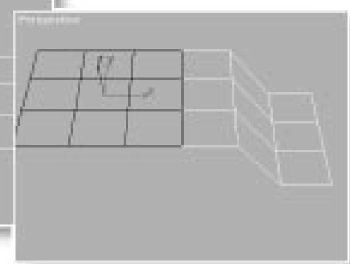
The attached object takes on a patch structure, and is now part of the Edit Patch object.

Working with Patches at Sub-Object Level

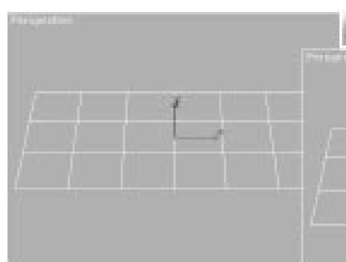
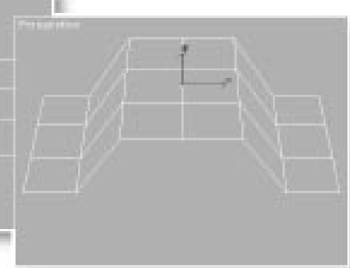
Display area

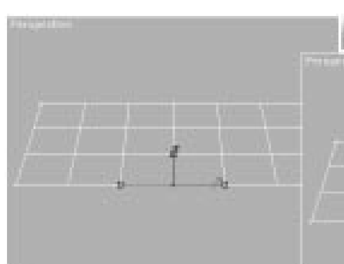
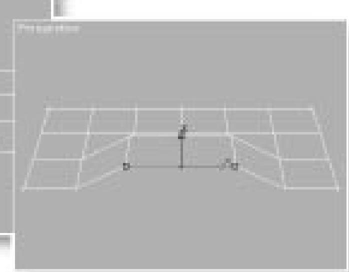
Effect of Display settings: Lattice and surface (left) and lattice only

Single patch selected and transformed

Single edge selected and transformed

Single vertex selected and transformed

During patch modeling, you work at different sub-object levels to transform or otherwise modify the lattice and surface that make up the individual patch.

Levels of Edit Patch

The Edit Patch modifier has three levels of sub-object editing: Patch, Edge, and Vertex. The selection you make at each level appears in the viewport as a component of the lattice. Each level maintains its own sub-object selection. When you return to a level, the selection reappears.

Patch—Selects the entire lattice of a patch. At this level, a patch can be detached, deleted, or its surface subdivided.

When a patch is subdivided, the surface is broken into smaller patches, each with its own lattice.

Edge—Selects a bounding edge of the patch lattice. At this level, edges can be subdivided, and new patches added to open edges.

- The axis coordinate icon is in the middle of a single selected edge.
- For multiple selected edges, the icon is at the selection center.

Vertex—Selects a vertex control point and its vector handles on a patch lattice. At this level, vertices can be welded and deleted.

- The axis coordinate icon is at a single selected vertex.
- For multiple selected vertices, the icon is at the selection center.
- Vector handles appear as small green squares around the vertex.

Transforming a Selection

When you transform a selected lattice component, the surface is smoothly drawn along the current transform axis. In effect, the lattice acts like a gizmo for the patch surface. Changes to the lattice are translated to the surface geometry of the patch. If a component is shared by another patch, as is often the case, both patches are transformed.

To illustrate selection and transformation for each level, figures on the opposite page use a simple case of two quad patches forming a single flat surface.

Selections are shown on the lattice, with the surface display turned off. Transformation in each case is a move along the Z axis. The effect is shown in a rendered viewport with the surface display turned on.

Displaying Lattice and Surface

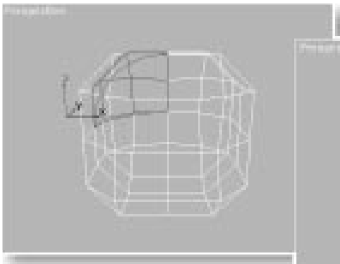
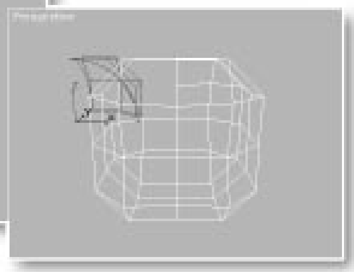
When using Edit Patch, you often need to switch between lattice and surface. For example, you might want to turn off the surface while you select control points on the lattice. Then, when your selection is complete, you could return your view of the surface to see the effects of pulling on the control points.

Every rollout in Edit Patch—Object, Patch, Edge, and Vertex—has the same set of controls to set the display of lattice and surface.

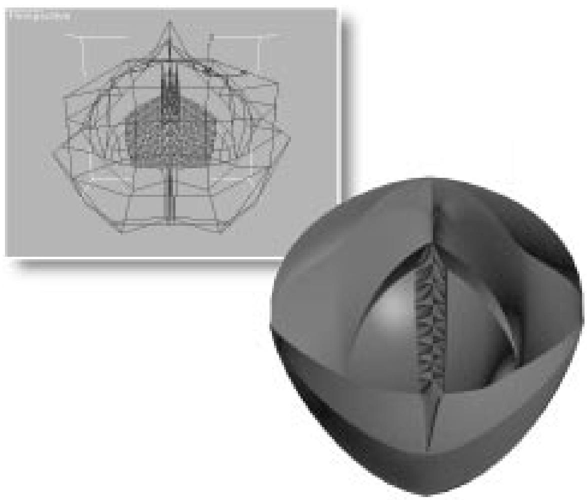
By default, both boxes are set, making lattice and surface visible.

- When you change settings in the Display area at one level, the same settings appear when you switch to another level.
- If both settings are cleared, the display disappears.

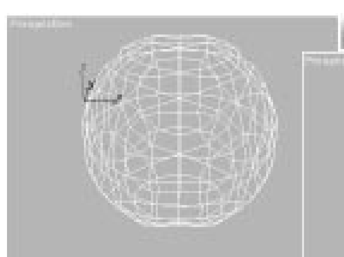
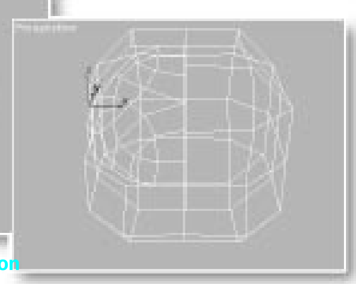
Editing at Patch Level

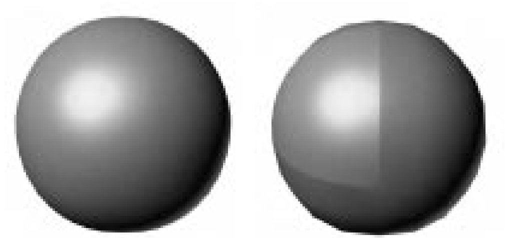
Detach only and Detach with reorient



Effect of modeling with multiple subdivisions

Patch subdivided with propagation



Patch subdivided without propagation

Patches are composed of a lattice and a surface.

- Lattice and surface are the primary control structures used in patch modeling.
- Edges and vertices are components of the patch lattice.

At Patch level, selected patches can be subdivided, detached, and deleted.

When you apply Edit Patch, sub-object selection is turned on by default. Select the sub-object Patch level to display the Edit Patch rollout.

Detaching and Copying Patch Surfaces

Patches, along with their surfaces, can be detached as separate objects or copies. With either operation, they can be reoriented to the center of the active grid. Patch surfaces can also be deleted. Each option assumes you begin with a selection of one or more patches.

Detach—Detaches selected patches as a separate object and leaves the object in its original position.

Copy—Detaches copies of selected patches as a separate object and leaves the object in its original position.

Reorient—Used with either Detach or Copy option. The object is moved to the origin of the active grid.

See online reference for specific procedures.

Subdividing Patches

Edit Patch provides an option for subdividing selected patches into smaller patches, and optionally adjacent patches as well.

Subdivision increases the number of vertices in the lattice and adds tessellation to the surface.

With multiple subdivides, you can create fine-grained geometry that models smoothly even at very small scale.

These are the options:

Subdivide—Divides selected patches. You can subdivide multiple times.

Each subdivision increases the number of patches, which become increasingly smaller.

Propagate—Extends subdivision to unselected patches bordering the selection. Default=set.

Propagate maintains surface continuity and smoothing. If Propagate is not set, only selected patches are subdivided, disconnecting those patches from the rest of the patch surface. You can see this discontinuity when the surface is rendered. In most cases you want to set Propagate.

Editing at Edge Level

Working at the Edge level gives you broad control when transforming a surface. You can lift and move the boundary edge of a patch, or scale and rotate it. The edges themselves can be subdivided. You can also add triangular and quadrilateral patches to open edges, extending the patch whenever needed. These features are covered in the following topics.

When you apply Edit Patch, sub-object selection is turned on by default. Select the sub-object Edge level to display the Edit Edge rollout.

Adding Patches to a Selected Edge

As you model, you might want to extend a surface. You can do this by adding a new patch to an *open* edge.

An open edge is one that bounds a single patch, and is therefore not shared with another patch.

You have these options:

Add Tri—Adds a triangular patch to a selected edge.

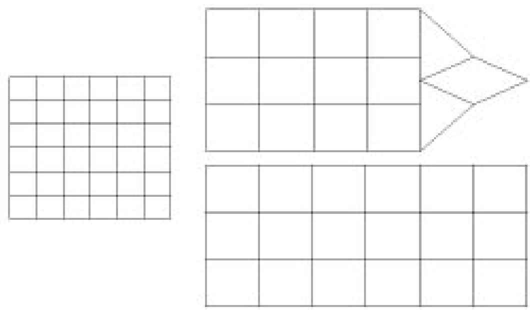
Add Quad—Adds a quadrangular patch to a selected edge.

Building on Existing Geometry

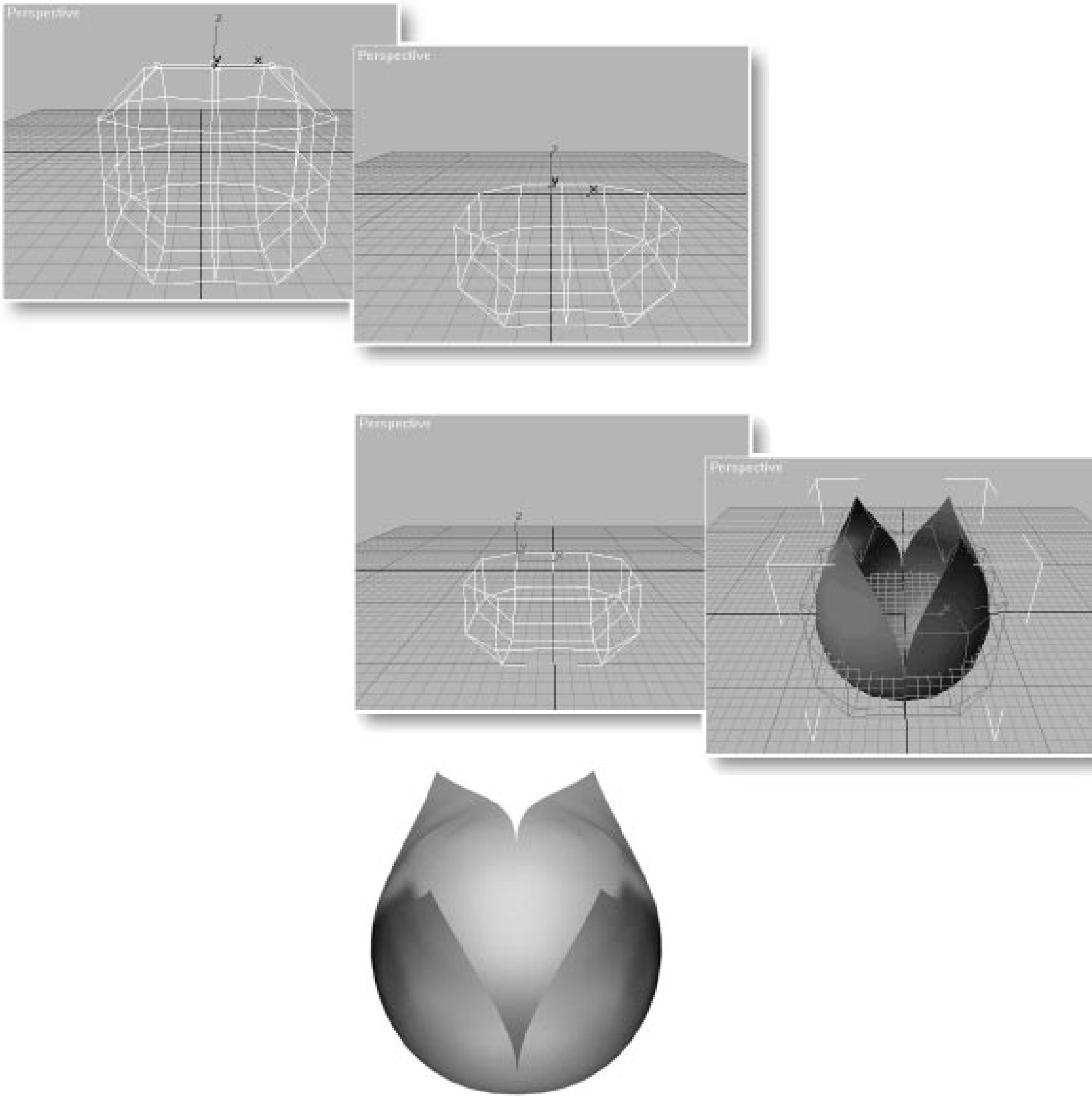
You can add Tri and Quad patches to any open edge of an object, The new patches adapt to the existing geometry. On closed objects like a sphere, you can delete one or more existing patches to create open edges, and then add new patches. When you add a patch to a curved edge, the new patch follows that curve and seamlessly extends it.

The figures on the opposite page show an example of building on existing geometry.

- The first set of figures begins with the lattice of a sphere, its top vertex selected. Deleting the vertex removes the top half of the lattice, creating open edges around the circumference.
- The second set shows the effect of adding a Tri Patch to each of the open edges.

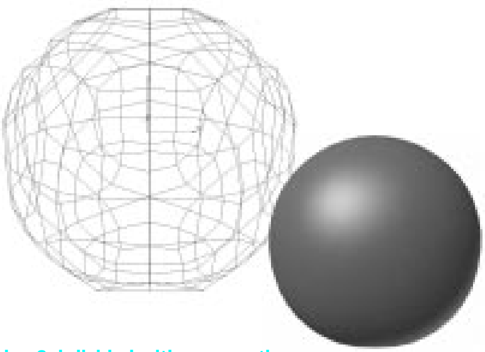


Tri and Quad patches added to an edge of the patch lattice (left)

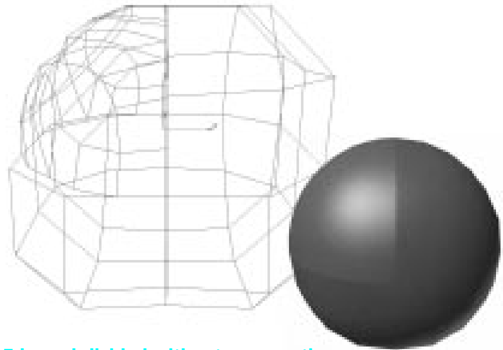


Building a new 3D surface with tri patches

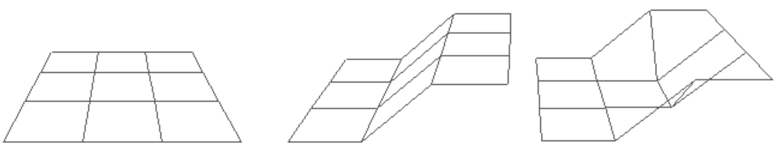
Subdividing Edges and Unlocking Interior Edges



Edge Subdivided with propagation



Edge subdivided without propagation



Effect of lifting edges with Auto and Manual Interior

Edit Edge provides an option for subdividing patch edges. This increases the number of patches as well as the number of vertices in the lattice and adds tessellation to the surface. When a selected edge is shared by two patches, both patches are subdivided.

Subdivide—Divides a selected edge into two edges. New vertices are added at right angles to the selected edge, evenly spaced between existing vertices to form a denser lattice.

Propagate—Extends subdivision to edges in contiguous patches. Default=set.

Propagate maintains surface continuity and smoothing. If Propagate is not set, only patches sharing the selected edge are subdivided, disconnecting those patches from the rest of the patch surface. You can see this discontinuity when the surface is rendered. In most cases you want to set Propagate.

Unlocking Interior Edges

When you move an outer or *boundary* edge of a patch, the adjacent interior edges are normally "locked" so that they move in parallel with the boundary edge. This is often useful, because it provides a uniform transition across the patch. This default is known as Auto Interior.

At *Patch* level, you can change the default on a patch-by-patch basis by choosing Manual Interior. When you move a boundary edge, only edges connected to the patch boundary are affected. The interior edges are no longer locked to the boundary edge.

Warning: If you return a patch to the default, changes caused by Manual Interior are lost.

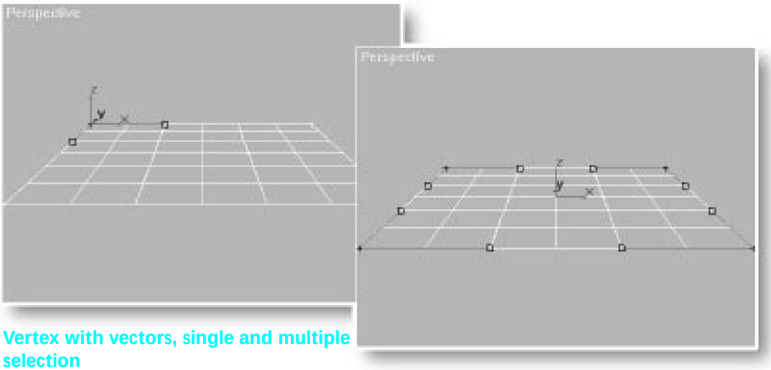
To unlock interior edges:

1. At Patch level, select one or more patches to unlock.
2. Right-click the selection and choose Manual Interior from the pop-up menu.

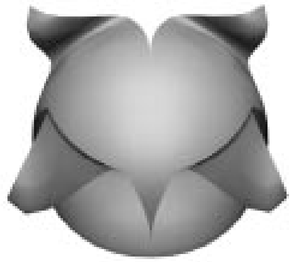
   The check mark moves from Auto Interior, the default, to Manual Interior. Interior edges are now unlocked.

Note: Manual Interior also allows selection of interior vertices. See "Transforming Interior Vertices."

Editing at Vertex Level



Vertex with vectors, single and multiple selection



Example of Vertex level editing

Working at the Vertex level gives you very detailed control when transforming a surface. You can move an individual vertex or any of its vector handles. Welding vertices lets you combine separate patches into a single patch.

Welding Vertices

This feature is intended to weld vertices on adjacent patches with open edges. The effect is to close the gap between patches. By default, both patches move when welded.

Weld Requirements

Welding patch vertices has a number of special requirements.

- Patches containing the vertices must be part of the same object. You can do this by attaching an object. See "Editing at the Object Level."
- Vertices must be on different patches. Nothing happens when you try to weld vertices within a patch.
- Vertices must be on "open" edges—edges used by only one patch. They cannot be shared edges.
- Welds cannot produce an edge used by more than two patches.

To weld vertices:

1. Select two valid vertices that meet the requirements listed above.

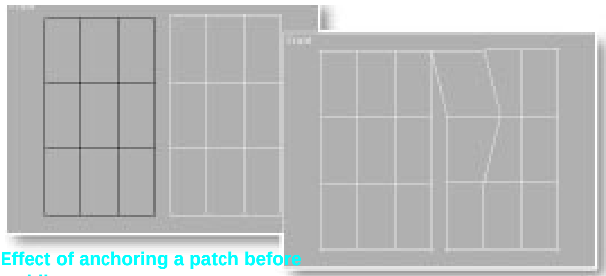
2. Set Weld Threshold to a value at least equal to the distance between the selected vertices.

3. Click Weld. The two patches containing the vertices move together and join at the welded vertex.

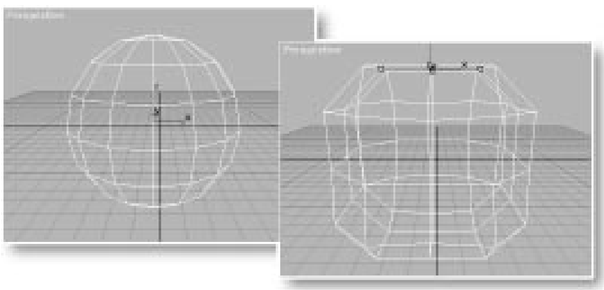
These are the welding controls:

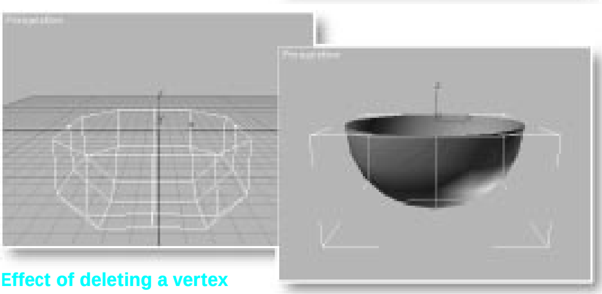
Weld—Creates a weld when all conditions are met.

Weld Threshold—Maximum distance in current units between vertices for weld to take effect. Continue to increase this setting as necessary until the selected vertices weld.



Effect of anchoring a patch before welding





Effect of deleting a vertex

Anchoring a Patch Before Welding

By default, the welding process shifts the geometry of both patches to a common center. You can anchor one patch so that the other patch moves to its location when the weld occurs.

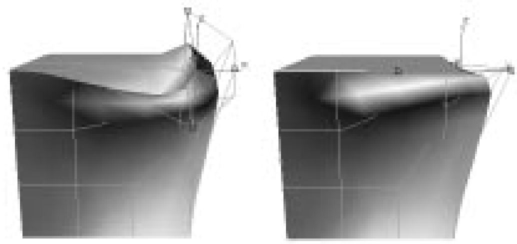
- Before you begin the weld, select the patch you want anchored *at Patch level*.
- Return to Vertex level and weld the vertices.

When the weld occurs, the anchor patch remains fixed while the other patch moves to make the weld.

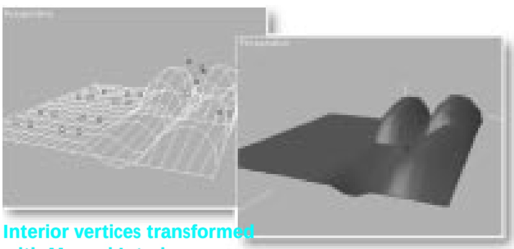
Deleting Vertices

A Delete option lets you remove vertices from the lattice. Use this option with caution. Deleting a single vertex also deletes any patches that share this control point. For example, if you delete the single vertex at the top of a sphere's lattice, the top four patches are also deleted, leaving a hemisphere.

Transforming Vertices and Vectors



Effect of Coplanar and Corner settings



Interior vertices transformed with Manual Interior

When you select a vertex in the lattice of a patch, its vector handles appear as green squares surrounding the vertex. By transforming either the vertex itself or any of the vectors, you gain additional control over modeling effects. Three settings control the options for transforming these elements: Coplanar, Corner, and Lock Handles.

Setting Coplanar and Corner

The Coplanar and Corner settings are a toggle on the right-click menu for one or more selected vertices. The setting is saved with the file. When you return to a particular vertex, the last setting is still in effect.

Coplanar—Keeps a selected vertex and its vectors in the same plane. Moving a vector handle moves the others as well to maintain a coplanar relationship.

Corner—Allows a selected vertex and its vectors to move out of plane. See “Setting Lock Handles” for interactions.

Vertices default to one of these types depending on their position in the geometry. In the following figure, a Corner vertex (at the corner of a box) is converted to Coplanar, distorting the geometry as the vectors realign. The original Corner vertex creates a smoother deformation when moved.

Setting Lock Handles

Lock Handles is a single option in the Topology area of the Edit Vertex rollout. Lock Handles affects only Corner vertices—it has no effect on Coplanar vertices.

Lock Handles—When cleared, vector handles of Corner vertices can be moved independently to any angle. Default=cleared.

When set, the relative angles between vectors are maintained. The vectors then move together around a given vertex. The vertex itself does not have to be selected.

- *Clear* Lock Handles when you want to alter the vector angles of a Corner vertex.
- *Set* Lock Handles to preserve the new angles.

Switching Between Settings

A keyboard shortcut lets you quickly switch from Coplanar to Corner, with the effect determined by the Lock Handles setting. To switch from Coplanar to Corner, use this technique:

- As you move a handle of a Coplanar vertex, hold down SHIFT. This switches the vertex to Corner.

  If Lock Handles is cleared (the default), SHIFT-Move “breaks” the handle, allowing it to move independently.

  If Lock Handles is set, the handles remain locked in their coplanar relationship.

When vectors have been moved with the Corner setting, you can return them to planar by toggling back to Coplanar. The plane is established based on the new positions.

Transforming Interior Vertices

Using program defaults, you can select only vertices and vectors on the outer edge or boundary of a patch. This default is known as Auto Interior.

In some cases, you might want to move the interior vertices. At *Patch* level, you can change the default on a patch-by-patch basis by choosing Manual Interior. This lets you select and transform individual interior vertices.

Warning: If you return a patch to the default, changes due to Manual Interior are lost.

To transform interior vertices:

1. Select one or more patches.

2. Right-click the selection and choose Manual Interior from the pop-up menu. The check mark moves from Auto Interior, the default, to Manual Interior.

3. Switch to Vertex level. The interior vertices appear as yellow squares.

4. With the Vectors filter set, transform the interior vertices of the selected patches.

Note: Manual Interior also affects interior edges. See “Editing at Edge Level.”

Chapter

12

Editing Meshes

In 3D Studio MAX, mesh editing is what lets you "sculpt" simple primitives into complex objects—a sphere into a head, or a cylinder into an airplane.

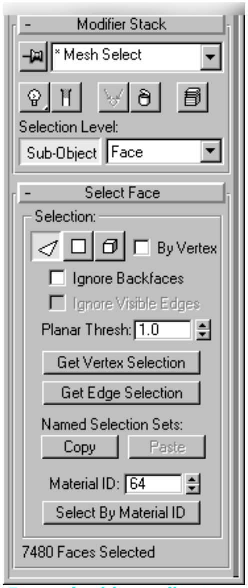
The tools for mesh editing are introduced in this chapter. These tools convert objects into triangular meshes for sub-object selection. There are dedicated editing options for each of three mesh components:

- Vertex
- Face
- Edge

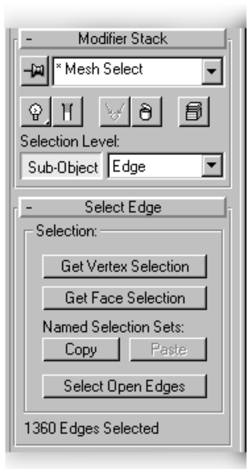
You can also transform and Shift-Clone selected mesh components, as well as apply modifiers to them. You can work at any scale, from a large terrain surface to a single vertex, face, or edge.

This chapter assumes a general knowledge of using modifiers and the modifier stack. See chapter 6, "Basics of Creating and Modifying Objects."

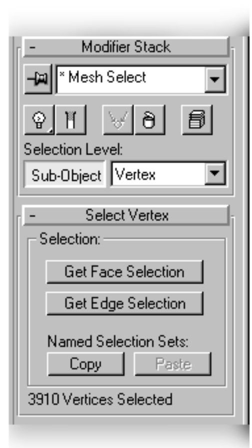
Sub-Object Meshes



Face sub-object rollout



Edge sub-object rollout



Vertex sub-object rollout

This chapter presents the features available for selecting and editing mesh geometry. It assumes a basic understanding of how to apply a modifier and make sub-object selections. It also assumes you know how to use the modifier stack.

- The first series of topics discuss the editable mesh option.
- A final topic covers Mesh Select and the other mesh-based modifiers you can use selectively in place of an editable mesh.
- This topic discusses the operations that apply to any sub-object selection of mesh geometry.

You can find the necessary background for this chapter in chapter 6, “Basics of Creating and Modifying Objects.”

Selecting and Transforming

In selecting and transforming sub-object geometry, you use standard techniques:

- Clicking any vertex, face, or edge selects it.

  Given the complexity of sub-object geometry, the Select By Name tool is a good choice for careful selection. For example, to avoid unintentional transforms, you can use Select By Name to make the selection, then switch to Move, Rotate, or Scale once you’ve finished.

- Holding down CTRL lets you add to or subtract from the selection with single clicks.
- Holding down ALT lets you a subtract from the selection with single clicks, or with Window/Crossing selections.
- Beginning a selection outside the object starts a region selection. Holding down CTRL dur-

ing region selection lets you add to the selection.

Once you've made a sub-object selection, you can use the SPACEBAR to lock the selection while you're working with it.

Using Sub-Object Selection

With either an editable mesh or a Mesh Select modifier, you can store three separate sub-object selections—one for each selection level (vertex, face, and edge). These selection sets are saved with the file. With sub-object selections, you have these options:

- Choose one of the selection sets to pass geometry up the stack to other modifiers. Only one selection set is active at time.
- Change to one of the other selection sets at any time.
- Use named selection sets for sub-object geometry you want to reuse.

  In modeling a head, for example, you might have a number of different vertex selections for forehead, nose, and chin. Such selections can be difficult to recreate, so named sets give you easy access to the original selection when you want to rework a particular area.

Cloning Sub-Object Geometry

Shift-Cloning a selection of vertices or faces displays the Editable Mesh dialog. You answer Yes or No to the question, "Do you want to create a new object?"

- If No, the selection is cloned in its new position and remains part of the original object.
- If Yes, the Detach dialog appears. Here you name the cloned copy, which becomes a plain mesh object, entirely separate from the original object.

Animating Sub-Object Geometry

When you work with an editable mesh, you can directly transform and animate a sub-object selection. In effect, the selection works like any other object.

Using XForm Modifiers for Animation

When you apply a Mesh Select modifier, there are no animation controllers assigned to the sub-object selection. This means that the selection has no way to "carry" the transform information needed for animation.

To animate a sub-object selection using Mesh Select, you apply either an XForm or Linked XForm modifier to the selection.

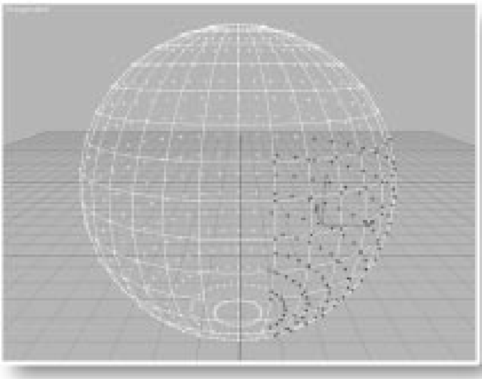
These modifiers provide the necessary controllers for animating the effects of transforms. In a sense, they give "whole-object status" to the sub-object selection.

XForm—Lets you directly animate transforms on a sub-object selection. Creates a gizmo and center for the sub-object selection. Both can be animated, with the center acting as a pivot point for the selection.

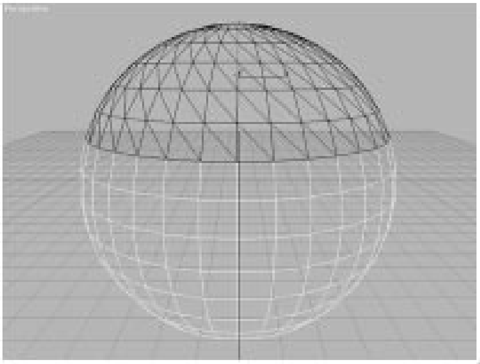
Linked XForm—Lets you choose another object to control the animation. The sub-object selection is linked to the "control object." When you transform the control object, the sub-object selection follows accordingly.

See "Using Mesh-Based Modifiers" for an introduction to the Mesh Select modifier. See online reference for full details on the XForm modifiers.

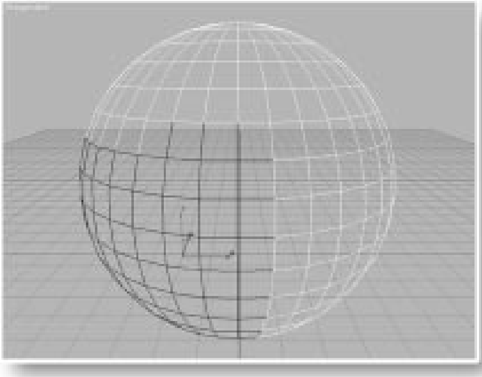
Using an Editable Mesh



Sub-object vertices selected



Sub-object faces selected



Sub-object edges selected

In 3D Studio MAX, the editable mesh option is the primary means of converting an object into a triangular *mesh* suitable for editing. This kind of mesh is made of three components: *vertices*, *faces*, and *edges*.

You can convert the following objects into an editable mesh:

- Geometric primitives or splines, with or without modifiers.
- Objects converted using mesh-based modifiers or the Collapse utility on the Utility panel.
- Imported mesh objects from other programs.
- NURBS surfaces.

Converting to Editable Mesh

The procedure depends on whether modifiers have been applied to the object selected for conversion.

To convert to an editable mesh:

1. Select an object and open the Modify panel.

2. Click Edit Stack in the Modifier Stack rollout.

- If the object has no modifiers, choose Editable Mesh in the pop-up menu. This converts the object directly to an editable mesh.
- If the object has one or more modifiers, click Collapse All in the Edit Modifier Stack dialog. This collapses the existing object to an editable mesh.

General Options

Once converted to an editable mesh, you have these general options:

- Use controls on the Edit Vertex, Edit Face, and Edit Edge rollouts to modify a sub-object selection. See following topics.

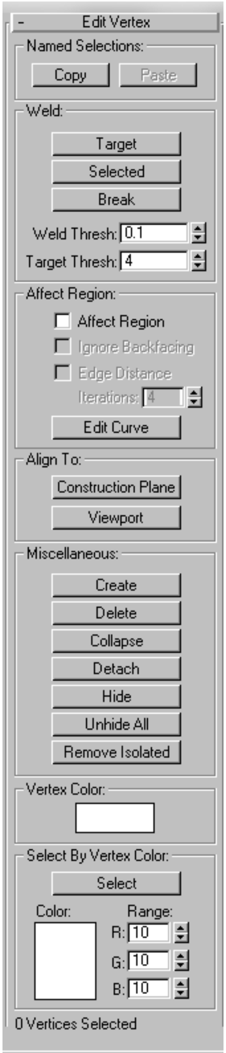
- Apply one or more modifiers to a sub-object selection.
- Pass a sub-object selection up the stack.
- Transform or Shift-Clone a sub-object selection. These are the same techniques you would use for any object.
- At the object level, attach other objects to an editable mesh.
- At the Face selection level, alter the surface characteristics of selected faces. See chapter 19, "Adjusting Normals and Smoothing."

Effect of Converting to Editable Mesh

Converting an object to an editable mesh removes all parametric controls, including the creation parameters. For example, you can no longer increase the number of segments in a box, slice a circular primitive, or change the number of sides on a cylinder. Any modifiers you apply to an object are collapsed as well. After conversion, the only entry left on the stack is "Editable Mesh."

- Before converting to an editable mesh, be sure you have the basic geometry established in preparation for detailed modeling.
- If necessary, undo the conversion and adjust parameters.

Editing Vertices



Edit Vertex rollout when object is converted to editable mesh

Vertices are points in space. They define the structure of faces. When vertices are moved or edited, the faces they form are affected as well. Vertices can also exist independently; such isolated vertices can be used to construct faces but are otherwise invisible when rendering.

You edit vertices by selecting the Vertex level of an editable mesh. The Edit Vertex rollout appears.

The following headings highlight some of the main features of the vertex editing with an editable mesh. See online reference for additional details and procedures.

Welding Vertices

There are two methods for *welding* or joining vertices.

Target method—Lets you pick a vertex and move it to find another vertex—a "weld target."

Selected method—Lets you join a selection of vertices based on a threshold setting. Vertices within the threshold are welded together into one or more vertices at the center of the selection.

Using either method, vertices must be on the same object. To weld vertices on different objects, you need to *attach* the objects to one another before welding. See "Attaching and Detaching Objects."

Contouring a Region of Vertices

Controls in the Affect Region area let you set up the equivalent of a "magnetic field" around a selected vertex. Unselected vertices within the field are drawn along smoothly as you transform the selected one. In effect, you produce a mountain from a molehill.

With this very versatile feature, you can sculpt the vertices of any mesh—for example, drawing a sphere into an egg, or gently curving a flat surface into hills and valleys. A large, flat box with many segments makes a good test case.

- A single vertex works well for many purposes. Moving a vertex along a single axis is the most useful for smoothly raising and lowering surfaces.
- For multiple vertices, you can also use Rotate or Scale.

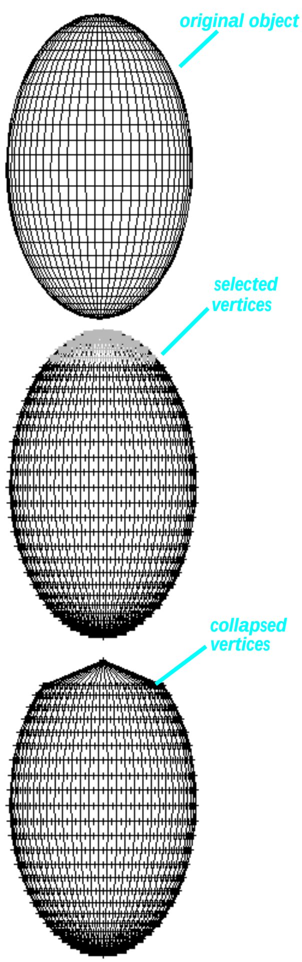
Aligning Vertices to Active Grid or Viewport

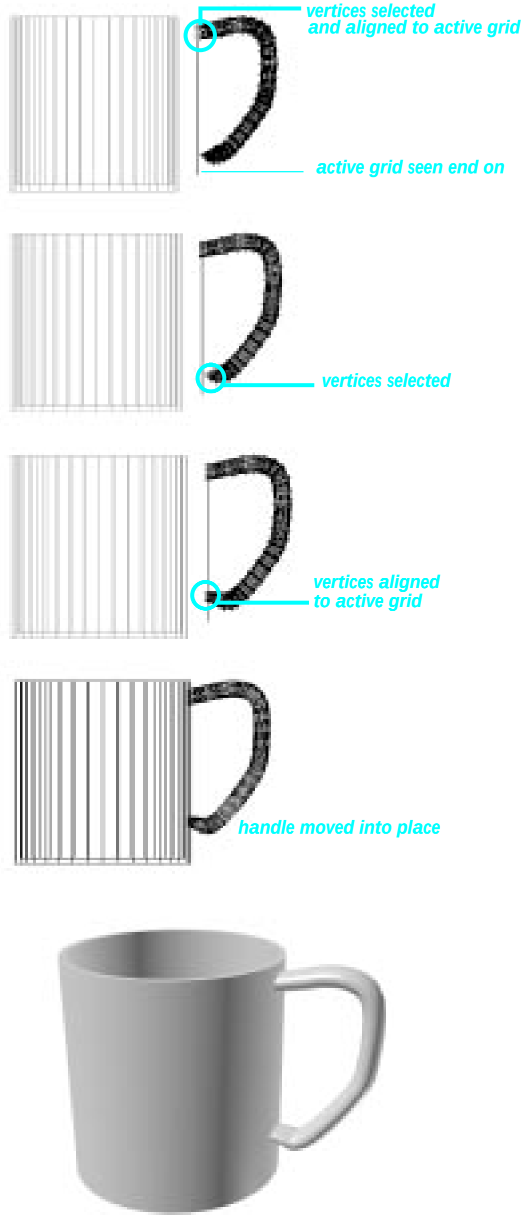
There are two options in the Align To area:

Construction—Aligns selected vertices to the current construction plane, whether you're working with the home grid in the active viewport or an active grid object.

Viewport—Aligns selected vertices to the plane of the active viewport.

- In an orthographic viewport, the effect is the same as aligning to the construction plane when the home grid is active.


original object
selected
vertices
collapsed
vertices


vertices selected
and aligned to active grid
active grid seen end on
vertices selected
vertices aligned
to active grid
handle moved into place

Hiding Vertices

You can hide vertices to make the remaining vertices easier to work with, then unhide them as needed. Hidden vertices remain part of the object but cannot be selected for transforming or applying modifiers.

- Be aware that hidden vertices are still selected after hiding. To avoid inadvertently deleting them, click outside the object or select other vertices.
- Add to the hidden vertices by repeatedly selecting and hiding vertices. If you hide all vertices of an object, the faces of the object remain.
- Use Unhide All to return all hidden vertices to visibility.

Hidden vertices cannot be selected, but they are saved with the Vertex-level selection set. For example, suppose you hide some vertices and select others, then go to another level. When you return to the Vertex level, the same vertices are again hidden and selected.

Creating Vertices

You can create new, isolated vertices that become part of the current object.

- To do so, click Create and click in any viewport. A single vertex is created. Each click creates another vertex.

These vertices are invisible during rendering unless used to construct faces. Possible uses include:

- Create vertices and turn them into faces with the Build Face option on the Edit Face rollout.
- Create vertices in a pattern, then pull vertices from the main object and snap them to the ones in the pattern. You can then weld the vertices to eliminate duplicates.

Deleting, Removing, and Collapsing Vertices

You delete, remove, and collapse vertices as follows:

- To delete vertices, select one or more vertices and click Delete.
- To remove any vertices unattached to other geometry, click Remove Isolated. This works regardless of the current selection.
- To collapse vertices, select the vertices and click Collapse. The vertices are replaced by a single vertex at the averaged center of the selection set. If you collapse all the vertices of a sphere, for example, the single remaining vertex is at the sphere's center.

Editing Faces

A face is the smallest possible mesh object—a triangle formed by three vertices. Faces provide the renderable surface of an object. While a vertex can exist as an isolated point in space, a face cannot exist without vertices.

You edit faces by selecting the Face level of an editable mesh. The Edit Face rollout appears.

Selection Methods

Three selection methods are represented by icons in the Selection area. Click one of these icons to set the active method. You can switch between methods as you make your selection. Hold down CTRL to add to and subtract from the current selection.

Face—A click selects a single triangular face. This is the smallest unit at the Face level.

Polygon—A click selects coplanar faces of a polygon, as determined by the Planar Threshold setting and edge visibility. See online reference.

The polygon, composed of two triangular faces, is what you usually see in wireframe as a four-sided "facet."

Element—A click selects an *element*—the contiguous faces of an object. For many objects, this is same as selecting the entire object.

For a text object, each letter is a separate element. Attached objects are also made up of elements, unless their vertices have been welded to form a single object.

Extruding Faces

Extrude moves selected faces along a normal and creates new faces that form the sides of the extrusion, connecting the selection to the object.

Use standard selection techniques to select faces, then extrude them by dragging or by direct entry.

Extrusion is along the averaged normal of each contiguous group of faces. If you extrude multiples of such groups, each group moves along its own averaged normal.

Tessellating Faces

Tessellation is useful for increasing local mesh density while modeling. You can subdivide any selection of faces. Two tessellation methods are available:

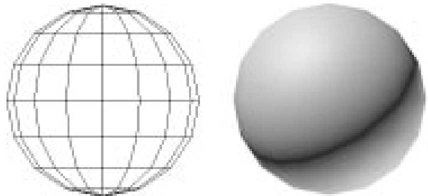
Face Center—Divides each selected face internally. This method has no effect on unselected faces.

Edge—Divides each selected face internally and also any unselected faces sharing an edge with the selection set. Has a tension setting to round tessellated faces. This methods helps reduce rendering irregularities between the selected tessellated faces and the surrounding faces.

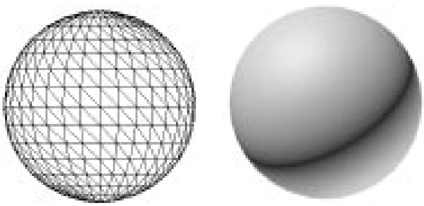
Exploding Faces

Explode converts selected faces into either separate, named mesh objects or elements.

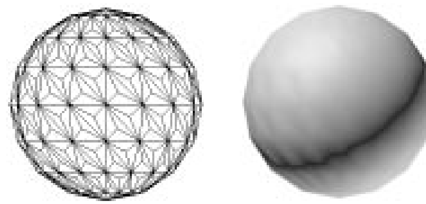
Both methods use a *threshold setting* to determine which faces are joined. This value represents the angle between adjoining faces. A setting of 0 makes every face a separate object or



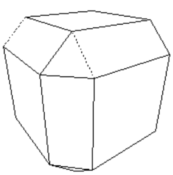
Original object



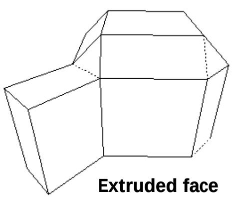
Edge Tessellated



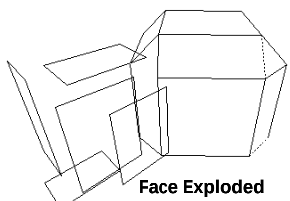
Face-Center Tessellated



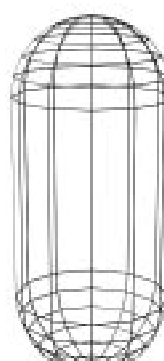
Original object



Extruded face



Face Exploded
as object and
moved



Original object

Rounded lower
faces made planar

element. A setting of 180 forces all faces into a single object or element.

Explode Into Objects—The selected faces become one or more separate mesh objects, depending on the threshold angle. Multiple objects all use the base name you supply and are numbered sequentially.

Exploded objects are completely separate copies from the original editable mesh, but lack any parametric data. They can be transformed and modified, or converted into editable meshes.

Explode Into Elements—The selected faces break into separate mesh elements, made up of one or more faces, each with its own local coordinates.

Each mesh element can be transformed and modified separately at the sub-object Face level. When you return to Object level, however, the elements act as part of the whole object when it is transformed or modified.

Making Faces Planar

While modeling, you might want to make a flat side on a object. You can do this with Make Planar.

To make faces planar:

- Select the faces and click Make Planar. The faces are flattened into a single plane.

In normal practice, you would use a contiguous selection set. If the selection includes faces on various parts of the object, the faces are still made planar, but with distorting effects on the rest of the geometry.

Building Faces

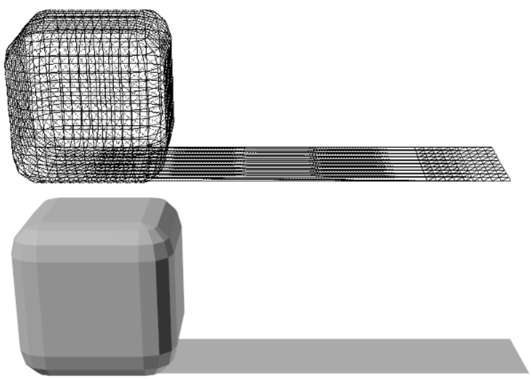
By building new faces, you can “stitch” isolated vertices into the fabric of a mesh, or create new faces from existing mesh vertices.

To build faces:

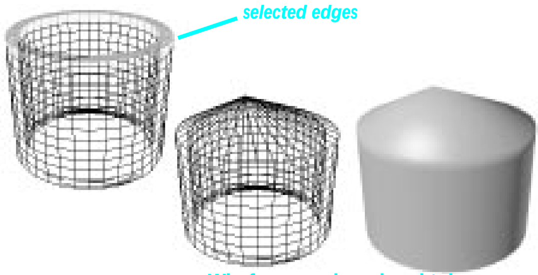
1. Click Build Face. All vertices in the object are highlighted, including any that have been isolated by deleting faces.
2. Drag a rubber-banding line from any vertex to a second vertex and release to form the base of a triangle. The direction in which you choose vertices determines the direction that face normals point.
3. Click a third vertex to complete the face. Repeat this process for other faces.

The vertices of different objects do not connect. To build faces between two or more objects, first *attach* the objects to one another, making them into a single object. See “Attaching and Detaching Objects.”

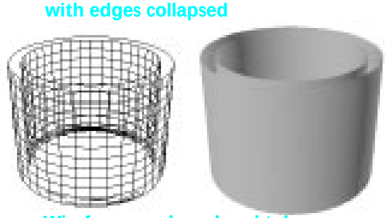
Editing Edges



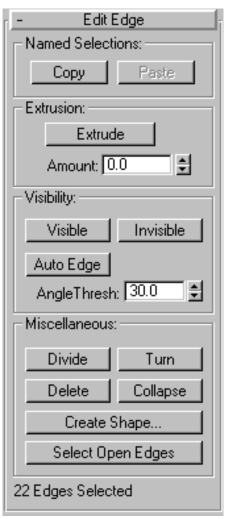
Wireframe and rendered object with extruded edge



Wireframe and rendered tube with edges collapsed



Wireframe and rendered tube with edges deleted



An edge is a line, visible or invisible, forming the side of a face and connecting two vertices. Two faces can share a single edge.

You edit edges by selecting the Edge level of the Edit Mesh modifier. The Edit Edge rollout appears.

Extruding Edges

The Extrude option moves selected edges along a local Z axis and creates new faces that form the sides of the extrusion, connecting the selection to the object.

Extrusion is along an averaged axis for each contiguous group of edges. If you extrude multiples of such groups, each group moves along its own averaged axis.

You use standard selection techniques to select edges, then extrude them by dragging or by direct entry.

For both methods, the next extrusion begins where the previous one ended. In other words, you can “extrude an extrusion,” unless you make changes to the resulting selection set.

While Extrude is on, use CTRL if you need to add to or subtract from the initial selection.

Setting Edge Visibility

While working at Edge level, you can set the visibility of selected edges as follows:

- Use Visible or Invisible to simplify wireframe display in the viewports.
- Use Auto Edge to determine visibility based on a threshold angle between faces.
- Set the degrees of the angle. A small angle means that only the edges between almost planar faces are invisible.

This setting has no effect on the geometry, only its visibility, and does not affect surface rendering. It does, however, determine the rendering of wireframe materials—only visible edges are rendered.

Edge visibility interacts with other features:

- At the Face level, Polygon selection uses visible edges for selection boundaries. When you click an area of invisible edges, the entire area selects as a single "facet."
- At object level, edge visibility interacts with the setting for Edges Only (as set on the Display panel, Display Optimization rollout). In a wireframe view, invisible edges disappear entirely when Edges Only is set (the default for each object). When cleared, invisible edges appear as broken lines.

Deleting and Collapsing Edges

You can select edges and either delete or collapse them. Both methods remove the edges from the geometry.

- After *deleting*, the selected edges disappear, along with the faces sharing these edges, leaving an opening in the surface.

  Deleting a single edge removes its two adjoining faces.

- After *collapsing*, the selected edges are deleted, and the surrounding edges and faces are rearranged to fill in the deleted area.

  Collapsing a single edge removes its two adjoining faces.

Dividing and Turning Edges

There are two options for fine-tuning the geometry of individual edges.

- *Dividing an edge* adds a vertex, splitting the edge into two half-length edges. New edges are created to meet the new vertex, generating additional faces.
- *Turning an edge* reorients it to new vertices. Turning an edge a second time flips it back to its original position. In general, the face count is unchanged. However, by repeatedly turning edges, you can cause edges to overlap into complicated and confusing geometry. This is a tool to use sparingly.

Using Create Shape

Create Shape turns an edge selection into a separate shape, an editable spline.

To use Create Shape:

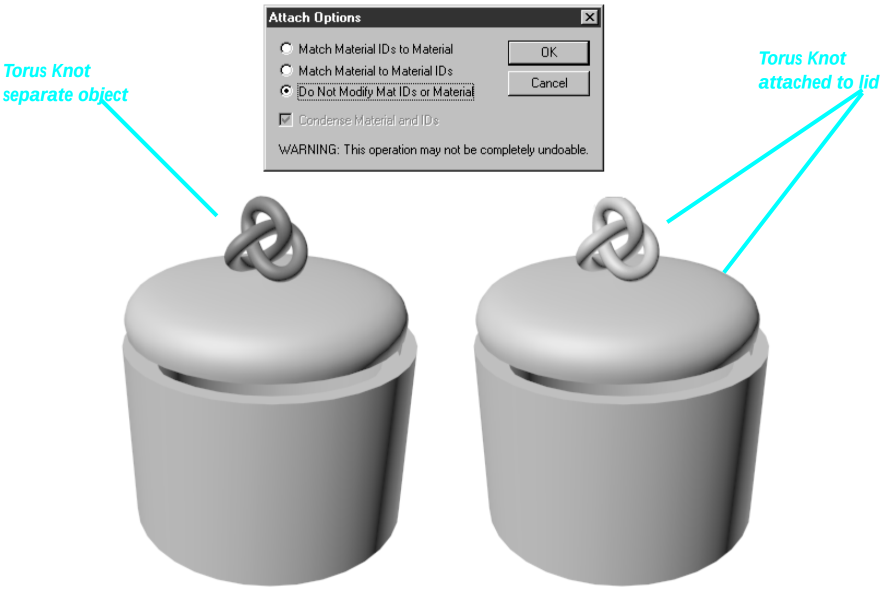
1. Select the edges you want to reproduce as a shape. Click Create Shape to display its dialog.
2. Choose between a *Smooth* or *Linear* shape type. *Smooth* produces rounded edges and corners.

   The new shape is exactly coincident with the vertices in the selected edges.

Selecting Open Edges

Clicking on Select Open Edges selects all edges with only one face. This usually identifies where missing faces exist in the mesh.

Attaching and Detaching Objects



In 3D Studio MAX, you use features of an editable mesh for both attaching and detaching objects.

- Attachment works at the *object* level of the editable mesh. It creates a combined object that keeps all the characteristics of the original object.

  At the mesh level, the original and attached objects are technically *elements*. Their meshes are not contiguous, even if vertices are snapped together. If you weld the vertices between elements, you produce a true, single *object*. See “Editing Vertices” earlier in this chapter for information on welding.

- Detachment works at the *sub-object* face or vertex level of an editable mesh. It creates either a separate mesh object or an element of the original object.

Attaching Objects

Attachment produces a combined object based on a selected editable mesh.

- The editable mesh retains its name and stack structure. Any animation keys are unchanged in the new combined object.
- Objects selected for attachment become part of the editable mesh object. An attached object loses its name, stack structure, and any animation keys. Its shape is determined by its current frame when the attachment occurs.

To attach objects:

1. Select an object that has been converted to an editable mesh.
2. Turn off the Sub-Object selection button. The Edit Object rollout appears with options for Attach and Multiple Attach.

3. Click Attach, and then click the object you want to attach. An Attach Options dialog lets you decide how you want to handle materials and material IDs.

   When both objects appear in white highlight, attachment is complete.

4. While Attach is on, click any other objects you want to attach. These objects become part of the combined object. Any transformation or modification now affects the entire object.

To attach multiple objects:

- Click Attach Multiple to display a standard dialog for selecting multiple objects by name and category. After making a selection, you see the same Attach Options dialog.

When using either the Attach or Multiple Attach option, be sure to position objects where you want them *before* attaching.

You might typically attach objects that touch or overlap, but this is not required. You can attach objects wherever they're located in your scene.

For many purposes, you can *group* objects instead of attaching them. Grouping unites a scattered set of objects into a single unit, but retains the parametric qualities of each member object. You can also add and remove objects in a group. See chapter 3, "Selecting Objects".

Detaching Objects

You can separate any part of an object at the face or vertex level into an element of the object or as a detached mesh object. Except for its missing geometry, the original object is unaffected.

To detach objects:

1. Select an object that has been converted to an editable mesh.

2. Turn on Sub-Object, set the selection level to Face or Vertex, and select the geometry you want to detach.

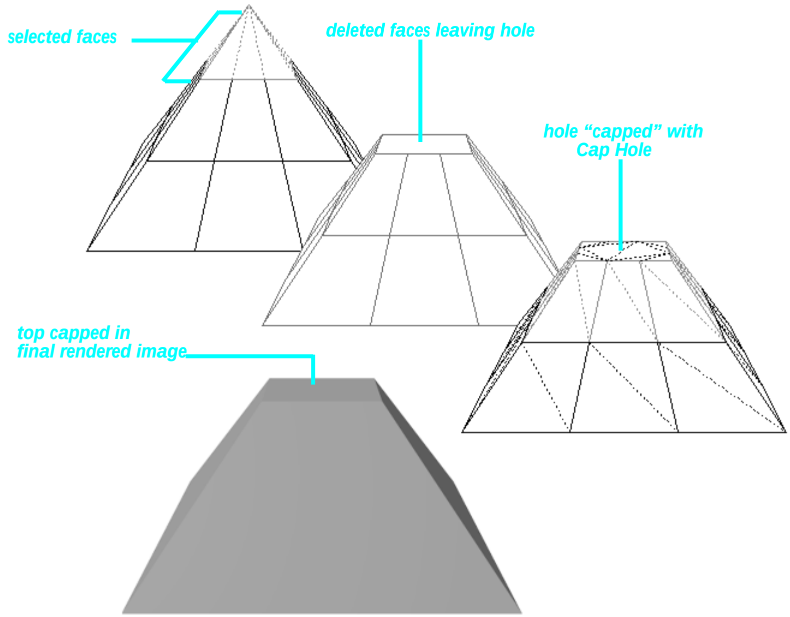
3. On the Edit Face or Edit Vertex rollout, in the Miscellaneous area, click Detach.

   The Detach dialog appears with two options:

- Enter a name for the detached object. The selected geometry deselects and changes color. It is now detached, and can be transformed and modified separately from the original object. The new object is a simple mesh that you can attach to other objects.

- Set Detach To Element. This grays out the name field. The selected geometry becomes an element of the original object.

Using Mesh-Based Modifiers



3D Studio MAX supplies a number of modifiers that duplicate the functionality contained in the Edit rollouts of an editable mesh.

- Use to access and modify sub-object geometry for a parametric object. Creation parameters are unaffected. As with other modifiers, their effect is passed up the stack.
- Use, where appropriate, in conjunction with an editable mesh.

Mesh Select

Mesh Select is a basic selection tool. It makes sub-object selections of mesh geometry available to other modifiers in the stack. It also returns the sub-object selection set in the stack to the object level.

As with an editable mesh, there are separate rollouts for vertex, face, and edge, and you can make separate selections for each mesh component.

- Each rollout contains Get buttons that retrieve the selection sets of the other components. For example, at face level, vertex and edge selections appear as face selections.
- Each rollout contains buttons for copying and pasting named selection sets.
- Transform buttons in the toolbar are automatically turned off. Only Select Object is active, because this modifier is for selection only. Use Xform and Linked XForm for transforms and animation. See online reference.
- Show End Result in the Modifier Stack rollout is automatically turned off. Click and hold this button to see the effect of sending a geometry selection up the stack.

- Selection sets are maintained even if you change the creation parameters, such as adding more segments to an object.

Turning Off Sub-Object Selection

An important use of Mesh Select is to turn off a sub-object selection currently on stack. The selection can be passed up the stack by an editable mesh or another Mesh Select modifier.

To turn off sub-object selection:

1. Add a new Mesh Select at the point in the stack where you want to end sub-object selection.
2. Click Sub-Object to turn it off. This returns selection on the stack to the whole object. The new Mesh Select no longer has an asterisk next to its listing in the stack.

Survey of Other Mesh-Based Modifiers

These are the primary modifiers used to manipulate mesh geometry. Many of these modifiers duplicate functionality found in editable meshes. Some contain additional features.

See online reference for details and procedures.

Delete Mesh—When added to the stack, deletes the current mesh selection from Mesh Select or an editable mesh. Removing this modifier restores the selection.

Affect Region—Translates an area of vertices to form a bubble or indentation over the surface of an object. Sets a falloff distance around selected vertices and smoothly pulls adjoining vertices. Works on any renderable object.

Face Extrude—Moves selected faces along a normal and creates new faces that form the sides of the extrusion, connecting the selection to the object.

- Unlike the same function in an editable mesh, you can animate all parameters.

Tessellate—Increases the mesh density of selected faces. If no sub-object face selection is on the stack, the entire object is tessellated.

- Unlike the same function in an editable mesh, you can show tessellation as polygons (two triangles) as well as faces (single triangles). Polygons are sometimes called *quads*.

Cap Holes—Builds faces to repair holes in an object or sub-object selection passed up the stack. A *hole* is defined as a loop of edges, each of which has only one face.

- Cap Holes works best on reconstructing planar holes, but also does a reasonable job on non-planar holes.

Preserve—Compares an original mesh object with an edited copy. Preserves, within limits, the edge lengths, face angles, and volume of the original.

- Pushing and pulling vertices at the sub-object level typically stretches the edge and alters the face angles, resulting in irregular topology. Preserve generates more regular edge lengths and a “cleaner” mesh.

Edit Mesh

Edit Mesh is a modifier that contains many of the controls found in an editable mesh. It is included for compatibility with 3D Studio MAX Release 1.

Edit Mesh has a high memory overhead. Unless you need to maintain compatibility, use the newer editable mesh or a combination of mesh-based modifiers to work with mesh geometry.

Chapter

13

Creating Compound Objects

Morphs and Booleans are *compound objects* that create new geometry by combining other geometric objects.

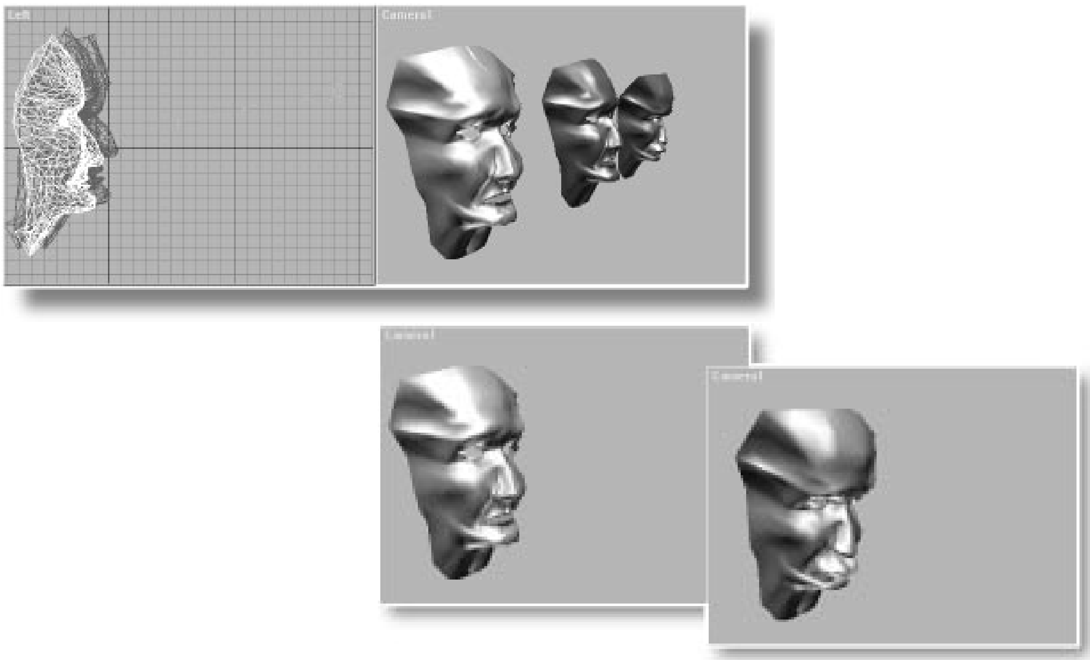
Morphs animate the vertices of one object to match the vertices of another object. Morphing is a 3D animation technique. You can morph among several objects—the original object changes its form successively as the animation plays.

Booleans combine objects using the Boolean operations of union, intersection, or subtraction (difference). For example, if a cylinder intersects a box, subtracting the cylinder creates a box with a hole in it. Boolean operations can be animated.

Other compound objects introduced in this chapter are:

- Conform
- Connect
- Scatter
- Shape Merge

Creating Morphs



Seed object, target objects, and resulting animation

Morphing is a 3D animation technique similar to *tweening* in 2D animation. In the simplest case, one object appears to change into another object.

A morph object combines two or more objects by interpolating the vertices of the first object to match the vertex positions of another object.

- The original object is the *seed object*.
- The object into which the seed object morphs is the *target object*.

Required Conditions

Before you can create a morph, the seed and target objects must meet these conditions:

- Both objects must be mesh objects.
- Both objects must have an equal number of vertices.

If these conditions don't apply, the Morph button is grayed out.

You can morph one seed into multiple targets. The form of the seed object changes successively to match those of the target objects as the animation plays.

You can use any kind of object as a morph target, including an animated object or another morph object, as long as the target is a mesh that has an equal number of vertices to the seed object.

Creating a Morph Object

You create morphs from the Create panel. The Morph button is enabled if the scene geometry allows you to create a morph.

Details for each step follow the general procedure. Notice that you don't click the Morph button until the third step.

General procedure to create a morph:

1. Set up the geometry to use in the morph.
2. Select the object to use as morph seed.
3. Under Geometry, choose Compound Objects. Under Object Type, click Morph.
4. Select the morph target objects at keyframes.

Setting Up Geometry

Make sure that the seed and target objects have the same number of vertices.

- When you create lofts to use as morph seeds and targets, make sure that Morph capping is on and Adaptive Path Steps is off.
- All shapes in the loft object must have the same number of vertices, and each shape should have both Adaptive and Optimize set to off. See chapter 9, “Creating Lofts.”
- Other shape-based objects, such as those with Extrude or Lathe modifiers, should also have Adaptive and Optimize set to off.

Selecting a Seed

The seed should be the only object selected. The Morph button is grayed out unless the scene contains morphable geometry and a single seed object is selected.

Warning: The selected object is permanently converted to a morph object as soon as you click Morph, *whether or not* you proceed to select a target object. The only way to restore the original object is to undo the Morph click.

Selecting a Target

This is the point at which the morph process begins.

To select a target:

1. After selecting a seed, click Morph.

   The name of the seed object appears at the top of the Morph Targets list in the Current Targets rollout.

2. In the Pick Targets rollout, choose the method for creating targets: Reference, Move, Copy, or Instance. The following topic, “Choosing Target Objects,” explains the purpose of each method.
3. Click Pick Target, then click to select one or more target objects in the viewports.

   As you select each target, its name is added to the Morph Targets list. If an object is not a valid target, 3DS MAX doesn’t let you select it.

Creating Additional Morph Keys

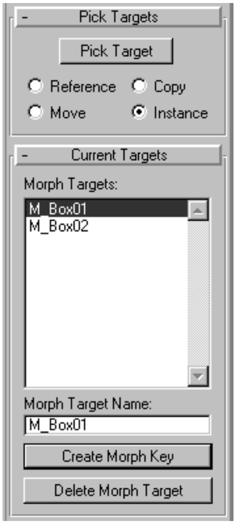
If you select a target object while you are not at frame 0, creating the target also creates a morph key. You can create additional morph keys from targets you’ve already selected. The Animate button does *not* have to be on to set morph keys.

To select additional morph keys:

1. Drag the Time Slider to the frame where you want to place the morph key.
2. Highlight the name of a target object in the Morph Targets list. The Create Morph Key button is enabled when a target object name is selected.
3. Click Create Morph Key. 3DS MAX places a morph key at the active frame.

To preview the effect of the morph, drag the Time Slider back and forth. You can view and edit the morph keys in Track View, which also lets you view the morph’s target object parameters. See chapter 25, “Basic Track View Use.

Choosing Target Objects



When you pick target objects, you designate each target as one of the following types:

- Reference
- Copy
- Instance
- Move
  (the object itself)

You make this choice in the Pick Targets rollout for Morph objects.

Options on the Current Targets rollout let you rename and delete targets, and set morph keys.

Choosing a Target Type

The target type you choose specifies how the morph object uses the target object. You can use an animated object or another morph as the target of a morph.

Base your selection on how you want to use the scene geometry *after* you create the morph.

Copy—The morph object uses a copy of the target. The original target object is unchanged.

Use Copy when you want to reuse the target geometry for other purposes in the scene.

Instance—The morph object uses an *instance* of the target object. If you later change the original target object, the morph object changes as well; and if you change the morph target, the original object changes.

Use Instance to synchronize morphing with animated changes to the original target object.

Reference—The morph object uses a *reference* to the target object. If you later change the original target object, the morph object changes as well; but if you change the morph target, the original object does not change.

Move—The morph object uses the target object itself, moving it into the morph. The original target object *disappears* and is no longer available after the morph is created.

Use Move if you've created the target geometry to be only a morph target, and have no other use for it.

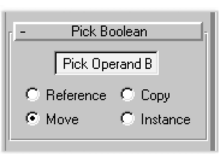
Other Target Options

When you highlight a name in the Morphs Target list, you have three options for that target:

- Rename the target in the field below the list.
- Delete the target.
- Create a morph key for that target at the current frame.

Creating Booleans

Boolean objects: union, intersection, and subtraction



A Boolean object is a combination of two objects which have had a Boolean operation performed on them on them. For geometry, the Boolean operations and their results are as follows:

Union—The resulting object contains the volume of both original objects. The intersecting or overlapping portion of the geometry is removed.

Intersection—The resulting object contains only the volume that was common to both original objects (in other words, where they overlapped).

Subtraction (or "difference")—The resulting object contains the volume of one original object with the intersection volume subtracted from it.

The two original objects are called the *operands*.

- The first object selected is called *operand A*.
- The second object selected is called *operand B*.

Creating a Boolean Object

You create Booleans from the Create panel. This is the general procedure:

To create a Boolean object:

1. Under Geometry, choose Compound Objects from the list. The Boolean button is enabled if an object is selected.

   The selected object is operand A. Operand A *becomes* the Boolean object; it loses its original type even if you don't perform a Boolean operation on it.

2. Click Boolean. In the Parameters rollout, the name of operand A appears in the Operands list.
3. In the Pick Boolean rollout, choose the copy method for operand B: Reference, Copy, Instance, or Move. These methods are described below.
4. In the Parameters rollout, choose the Boolean operation to perform: Union, Intersection, Subtraction (A–B), or Subtraction (B–A).
5. In the Pick Boolean rollout, click Pick Operand B.
6. Click in a viewport to select operand B. The software performs the Boolean operation.

Notes on the Boolean Object

Keep in mind these points:

- The operand objects remain as sub-objects of the Boolean object. By modifying the creation parameters of the Boolean's operand sub-objects, you can later adjust operand geome-

Two-color object created by differently colored operands

try to change or to animate the Boolean result.

- Occasionally the Boolean result can have face normals that point the wrong way. You can adjust normals by using the Normal modifier. See chapter 19, “Adjusting Normals and Smoothing.”

Copy Method for Operand B

When you pick operand B, you designate it as one of the following types: Reference, Move (the object itself), Copy, or Instance. You make this choice in the Pick Boolean rollout for Boolean objects.

The type you choose determines how the Boolean object uses the object B geometry. Base your selection on how you want to use the scene geometry *after* you create the Boolean.

Copy—The Boolean object uses a copy of the operand B object. The original B object is unchanged. Use Copy when you want to reuse the operand B geometry for other purposes in the scene.

Instance—The Boolean object uses an *instance* of the operand B object. If you later change the original B object, the Boolean object also changes; and if you change the Boolean object’s B operand, the original B object changes.

Use Instance to synchronize animation of the Boolean object with animated changes to the original B object.

Reference—The Boolean object uses a *reference* to the operand B object. If you later change the original B object, the Boolean object changes as well; but if you change the Boolean object’s B operand, the original B object does not change.

Object B geometry becomes part of the Boolean object regardless of which method you use.

Move—The Boolean object uses the operand B object itself. The original B object *disappears* and is no longer available after the Boolean is created. Use Move if you’ve created the operand B geometry only to create a Boolean, and have no other use for it.

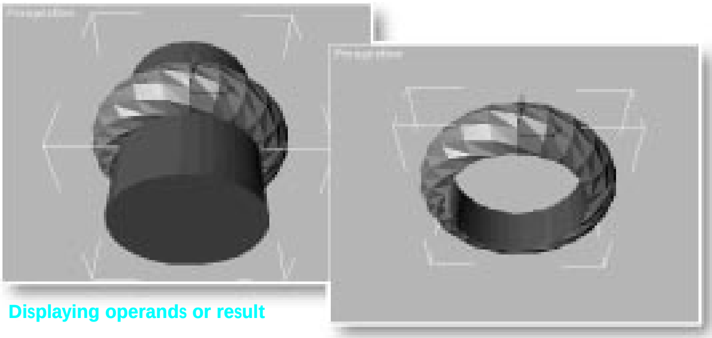
Boolean Faces and Materials

Creating a Boolean by intersection or subtraction can create new faces. A new face is given the material ID number of the faces that created it.

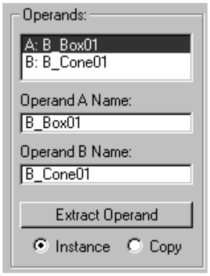
- For example, if you subtract one sphere from another, the hollow in the remaining sphere retains the material ID of the sphere that was subtracted.
- If the material IDs differ between the operands, the Boolean result has multiple IDs, and you can render it with multiple colors or materials by assigning it a Multi/Sub-Object material.

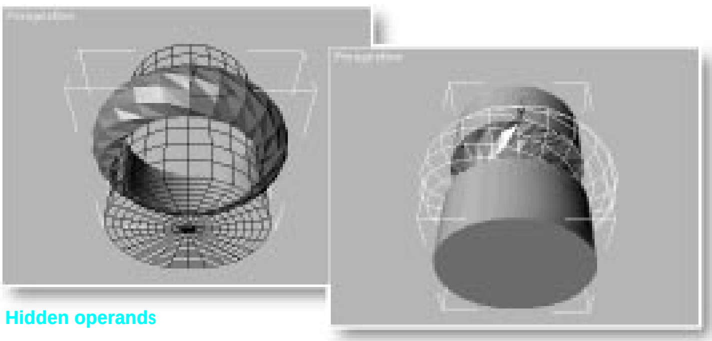
For more about multiple materials and material IDs, see chapter 21, “Designing Mapped Materials.”

Displaying Results and Operands

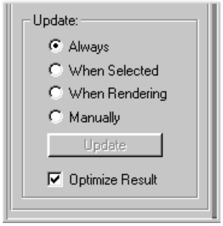


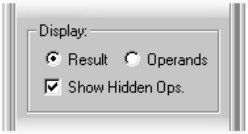
Displaying operands or result





Hidden operands





Options on the Parameters rollout can help you visualize the result of a Boolean operation. Visualizing can be tricky, especially if you want to modify or animate the Boolean.

- The Operands area has fields for naming operands A and B. Use names you can recognize easily.

  When you access the Operands area in the Modify panel, you can choose either operand and extract a copy (instance or reference) of the original geometry. One use of this feature is to check exactly what geometry went into the Boolean.

- The Display area helps you visualize how the Boolean is put together. The display controls have no effect until you've created the Boolean.

Result—Choose to view the result of the Boolean operation—that is, the Boolean object itself.

- Because you usually create Boolean objects from overlapping objects, if the B object isn't removed (you don't use the Move option), it often obstructs your view of the completed Boolean, or vice versa.
- You can also move the Boolean or the B object to better see the result.

Operands—Choose to view the operands instead of the Boolean result.

Show Hidden Ops—Set to view the "hidden" operands in shaded viewports. This parameter has no effect if the viewport is not shaded, or if operands are displayed.

- Operand geometry remains part of the compound Boolean object, although it isn't visible or renderable.
- When this parameter is set, 3D Studio MAX displays the operand geometry as wireframes in shaded viewports.

Tip: When operands are hard to see in a viewport, you can use the Operand list to select one or the other. Click the name of operand A or B to select it.

Boolean Update Options

By default, Booleans are updated whenever you change the operands. When your scene contains a complicated, animated Boolean—or more than one—this can impede performance. The Update parameters provide some alternate methods that can help performance.

Always—Updates Booleans immediately whenever you change an operand, including the original object of an instanced or referenced B operand. This is the default behavior.

When Selected—Updates Booleans only when you select them.

If your scene contains multiple Boolean objects that you need to update, select all objects to update the Booleans, then select none.

When Rendering—Updates Booleans only when you render the scene. With this option, viewports won't show current geometry. If in doubt, update Booleans manually.

Manually—Updates Booleans when you click Update. The Update button is enabled only when Manually is the active option.

Optimize Result—When set, 3D Studio MAX removes coplanar faces when it updates Boolean geometry. (The edge between coplanar faces can be visible in renderings.) Default=set.

When Operands Don't Overlap

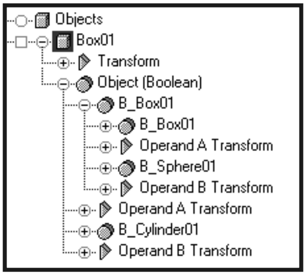
Normally Boolean operations are for objects that overlap in 3D space. If the operands don't overlap, 3DS MAX still performs the operation, but the result might not be what you expect.

Union—Both operands become part of the Boolean object, but the two parts are separated in space.

Intersection—The Boolean result is invisible because the intersection is empty.

Subtraction—The Boolean object has the same geometry as the original first operand; the operand being subtracted has no effect.

Using Boolean Trees



You can use a Boolean object as an operand when you create a Boolean object.

Using a Boolean as a Boolean operand creates a Boolean *tree*. One Boolean is nested within another, creating a hierarchical or "tree-like" structure. The tree can have multiple levels of nesting.

You can view the tree by using Track View. See chapter 25, "Basic Track View Use."

In this example, a box and sphere form the first Boolean at the bottom of the tree. This box-sphere object is then Booleaned with a cylinder to produce the final object. In this case, using default names, the final object is still called Box01 because the box was used as operand A at each step.

Navigating the Tree

When you select a multi-Boolean object in a viewport, you select the top level of the tree. You can navigate downward through the tree from the Modify panel. This is the general procedure to reach a lower-level Boolean operand.

To navigate through a tree:

1. Select the object containing the Boolean tree and open the Modify panel.

2. On the Parameters rollout, select an operand from the Operand list.

3. Open the modifier stack. Depending on the particular tree, one or both operands can show multiple Booleans in the stack.

4. Choose the next-lower Boolean from the stack. The operands for this level appear in the Operand list.

5. Repeat these steps to descend the tree. Reversing the procedure moves up the tree.

You can also navigate the tree directly by selecting any level in Track View.

Animating Booleans

You animate Boolean objects or their operands as you do other objects—by turning on Animate, moving to a keyframe, then applying transforms or modifiers. See chapter 22, “Animation Concepts and Methods,” for a complete introduction.

In the Modify panel for the Boolean object’s creation parameters, you can change the settings or the Boolean operation itself.

Transforming and Modifying Operands

As a compound object, a Boolean is composed to at least two other objects, its operands. You access the operands at the *sub-object* level. In other words, the original objects exist as subobjects of the Boolean object.

At the sub-object level, you can apply both transforms and modifiers to the original objects that make up the Boolean. These effects can be animated.

For example, by moving a cylinder operand that creates a hole in a Booolean object, you move the hole itself through the geometry of the object.

You can also apply modifiers to the operand. A Taper applied to the operand cylinder would change the shape of the hole.

To apply modifiers:

1. Turn on Sub-Object in the Modifier Stack rollout.
2. Select an operand by clicking its name in the Operand list.
3. Apply a transform or modifier to the operand.

Changing Operand Parameters

You can directly change and animate the creation parameters of operands.

To change and animate creation parameters:

1. Turn on Sub-Object in the Modifier Stack rollout.
2. Select an operand by clicking its name in the Operand list.
3. Choose the operand’s creation parameters by choosing the name of its object type from the list in the Modifier Stack rollout.
4. Change the operand’s creation parameters.

Animation Maintained in Booleans

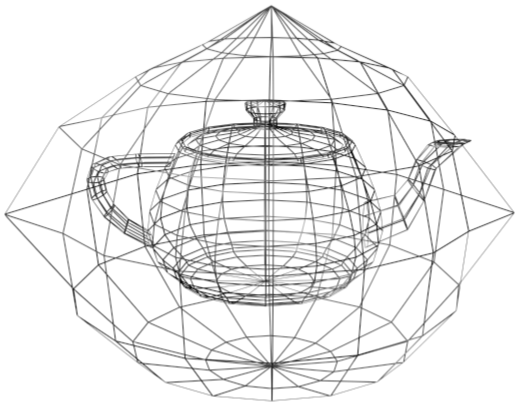
If an object you use as an operand is already animated, the animation is maintained in the Boolean. This was not true in 3D Studio MAX Release 1. The transform controller of the original object is cloned and assigned to operand B of the Boolean object. The animation is unchanged, but is now relative to the Boolean.

Possible Animation Artifacts

When you create Booleans, you sometimes see small edge objects or “slivers” where the two operands intersected. This effect tends to be more apparent when you animate Boolean operations. To avoid this effect, do one of two things:

- If the operation isn’t animated or if it occurs in only a few places, adjust the geometry by hand until the edge artifacts disappear.
- If the artifacts appear during an animation, apply a linear position controller to operand B. Then adjust the position of operand B in *every* frame of its animated range. See chapter 27, “Working with Controllers.”

Creating Conform Objects



This topic and those that follow present four additional types of compound object available in the Compound Object subcategory of the Create panel. Each type is briefly introduced with its terminology, setup options, and a general procedure for creating the object. See online reference for settings and other details.

Creating Conform Objects

The Conform compound object begins as two objects. Conform projects the vertices of one object onto the other, deforming the geometry of the projected object. There is also a space-warp version of Conform.

- One object, called the *Wrapper*, is selected first. The vertices of this object "wrap around" the other object, called the *Wrap-To*.
- Both objects must be mesh objects or objects that can be converted into meshes. If the selected Wrapper object is invalid, the Conform button is grayed out.

Note: Conform allows morphing between two objects containing different numbers of vertices. See online reference for the Vertex Projection Direction controls.

General Procedure

Begin by positioning the two objects. Typically, one might be inside the other, or the two might be close together.

To use Conform:

1. Select the Wrapper object and click Conform.
2. Choose an option in the Vertex Projection Direction area.

   If you choose Use Active Viewport, activate the viewport that looks in the direction you want to project the vertices. The projection occurs inward, away from the viewport.

   Other options are based on the geometry of either the Wrapper or Wrapped-To, or the Z axis of another object.
3. Choose Copy, Instance, or Reference to determine how the Wrap-To object is cloned. Move uses the original object for Wrap-to and does not make a clone.
4. Click the Pick Wrap-To Object button and select an object. For accuracy, use Select By Name in the toolbar to display a Pick list of available objects.

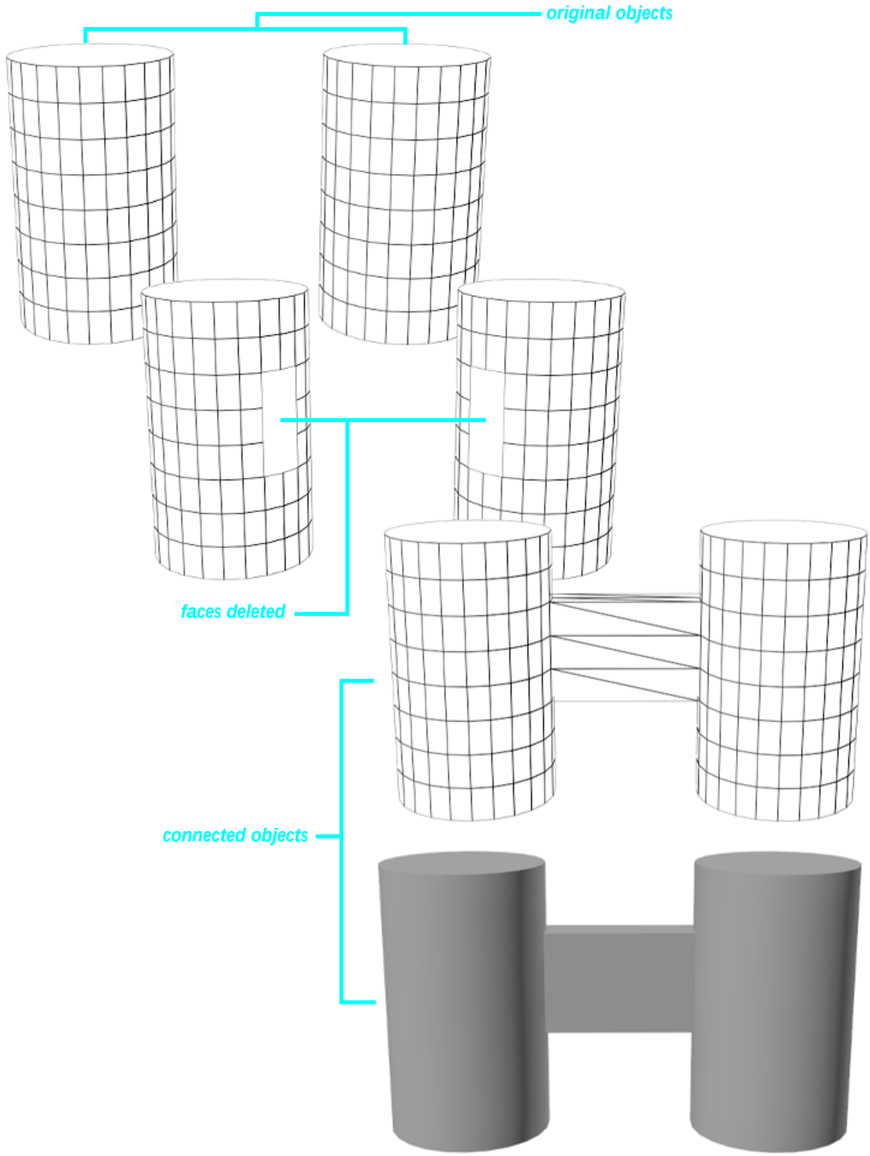
   The software now calculates the result.

Adjustments

You can make the following adjustments:

- By default, the distance between the two objects is quite close. In the Wrapper Parameters area, set Standoff Distance to a higher number to see the wrap effect more clearly.
- If you want to convert only the Wrapper object to an editable mesh after making final adjustments, hide the Wrap-To object by setting this option at the bottom or the Parameters rollout.

Creating Connect Objects



The Connect compound object builds a "bridge" between missing faces on two or more objects. A number of parametric controls let you refine the geometry of the bridge.

Connect is similar to Boolean. Selected objects becomes operands of the Connect object.

Objects must be mesh objects or objects that can be converted into meshes.

General Procedure

Begin by deleting faces on two objects. The resulting compound object builds new faces between these “holes” in the geometry.

Optionally, you can make holes in more than two objects and build a number of connecting bridges.

To create Connect objects:

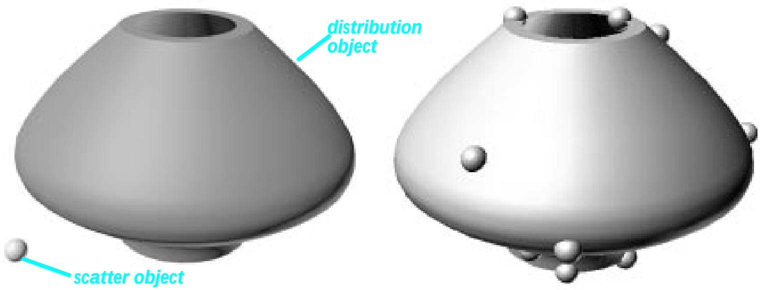
1. Position the objects so that the holes are in rough alignment, leaving a clear path for the construction of new faces.
2. Select one of the objects and click Connect.
3. Click Pick Operand and select a second object. Faces are generated between the two holes.
4. To connect other objects, click Pick Operand and repeat the selection for each object. New faces generate each time you do this.

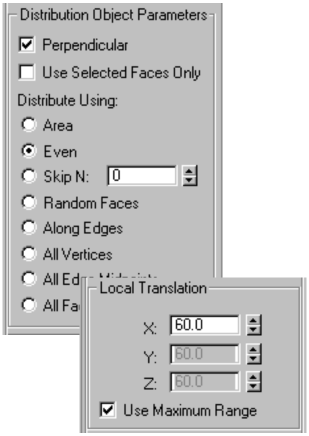
Adjustments

You can make the following adjustments:

- Increase Segments to produce a denser mesh in the connecting bridge.
- Adjust Tension to alter the curvature of the bridge. Positive tension draws the bridge inward and makes a somewhat smoother connection to the faces at either end.
- Set Bridge to smooth the edges of the new faces. Set Ends to smooth the transition between old and new faces. You can set both for maximum smoothing effect. The effect of these settings is only visible in a rendered viewport or scanline rendering.

Creating Scatter Objects





Before and after creating scatter objects

The Scatter compound object randomly scatters a selected object, called the *source object*, in either of two ways:

- Over the surface of a second object, called the *distribution object*.
- As a random array in space. Only the source object is required for this option.

In both cases, many parameters for size and distribution, as well as a seed value for randomness, can be adjusted after the initial scatter.

Source and distribution objects must be mesh objects or objects that can be converted into meshes. If the selected source object is invalid, the Scatter button is grayed out.

General Procedure

Begin by creating the object you want to scatter, and optionally a distribution object. Separate steps are provided for the two options.

To scatter over a distribution object:

1. Select the source object and click Scatter.

2. On the Scatter Object rollout, Use Distribution Object is set by default.

3. Choose Copy, Instance, or Reference to determines how the distribution object is cloned. Move uses the original object and does not make a clone.

4. In the Source Objects Parameters area, set Duplicates to the number of copies you want to scatter.

5. Select the distribution object. The software now creates the scatter.

Tip: If the source object is complex, or you plan to scatter a large number, set Proxy on the Display rollout to reduce the time needed to generate trials as you adjust parameters. The default, Mesh, uses the actual geometry.

To scatter as an array:

1. Select the source object and click Scatter.

2. On the Scatter Object rollout, set Use Transforms Only.

3. In the Source Objects Parameters area, set Duplicates to the number of copies you want to scatter.

4. On the Transform rollout, adjust values for Local Translation to move the duplicates away from the source object. Then adjust rotation and scale for the scattered array.
5. If necessary, set Proxy on the Display rollout to reduce the time needed to generate trials as you adjust parameters. Set Mesh when you have the scatter effect you want.

Adjustments

The effects of these adjustments appear in the viewport as you make them.

- When using a distribution object, choose an alignment option. These options are on the Scatter Object rollout in Distribution Object Parameters area. See online reference for details.
- When scattering an array, you can force all three axes of a transform to the highest value of the three. In the Transform rollout, set Use Maximum Range. When you clear this setting, the original values return.

The following adjustments are available for both types of scatter.

In the Source Objects Parameters area, you can:

- Change the number of duplicates after the scatter.
- Adjust the base scale of the scattered objects. This becomes the starting size for scale changes made in the Transform rollout.
- Increase Vertex Chaos from 0 to randomly dislocate the vertices of the source object. This is applied to all duplicates.

On the Display rollout, you can:

- Set Proxy or Mesh, as discussed previously.
- Set a percentage of scattered objects that appear in the viewport. This has no effect on the rendered scene.
- Set a Seed number in a range of 1–32,000 as the basis for all random values. This alters the overall distribution of the scatter.

Creating Shape Merge Objects

The Shape Merge compound object projects one or more shapes onto a mesh object. The shapes are either embedded in the mesh, altering the edge and face patterns, or subtracted from the mesh.

General Procedures

Begin by positioning the shape relative to the mesh object. The shape projects along its negative Z axis, so you want this axis pointing toward the surface selected for merging.

You can position multiple shapes on various sides of an object and merge them all as part of the same Shape Merge object.

To use Shape Merge:

1. Select the mesh object and click Shape Merge.
2. Choose Copy, Instance, or Reference to determines how the shape is cloned. Move uses the original shape and does not make a clone.
3. Click Pick Shape, and then select the shape.

   A pattern matching that of the shape is now embedded in the surface geometry of the mesh object.
4. To merge other shapes, click Pick Shape and repeat the selection for each shape. As you do this, each shape in turn is listed as an operand.

Adjustments

You can make the following adjustments:

- If you're working in a smooth shaded viewport, you won't see any change in the surface geometry. Right-click the viewport label and choose Edged Faces to see the shape.
- Merge is the default, set on the Parameters rollout in the Operations area. Set Cookie Cutter to remove the shape from the mesh.
- For Cookie Cutter, set Invert to leave the shape and remove the rest of the mesh.
- For Merge, Invert reverses the sub-object mesh selection. If you apply a Face Extrude modifier to the shape operand with Invert cleared by default, the area of the shape is extruded. If you set Invert, the rest of the surface is extruded, not the shape.
- To delete an embedded shape, select the shape in the Operands list and click Delete Shape.
- In the Output Sub-Mesh Selection area, options let you specify the selection level that is passed up the modifier stack. By default, None is set, making all selection levels available on the stack. Set Vertex, Edge, or Face to limit the available selection to a single sub-mesh geometry.

You can transform a shape operand on the surface of the object. This works with either Merge or Cookie Cutter.

To apply transforms:

1. On the Modify panel, select the shape in the Operands list.
2. Turn on Sub-Object and press SPACEBAR to lock the selection.
3. Move, rotate, or scale the selection. Rotate changes the shape of the projection against the surface. You can animate these transforms.

With the selection still locked, you can extrude the shape by applying a Face Extrude modifier. Increase the Amount value to extrude, and use a negative value to indent.

Chapter

14

Modeling Particle Systems

Particle systems are useful for simulating effects such as rain, snow, dust and bubbles, but you can also use particle systems to model objects whose behavior evolve over time. Examples of this second use for particle systems include flowing liquids, and explosions.

This chapter discusses the use of various types of particle systems. See the Online Reference for details about all particle systems.

Creating a Particle System

You create particle systems when you want to model an object or effect that can best be described as a large collection of similar objects behaving in a similar fashion. Obvious examples of such effects include rain and snow, but other equally valid examples include water, smoke, ants, and even crowds of people.

Click Spray, Snow, Super Spray, Blizzard, Particle Array, or Particle Cloud on the Create panel to create a particle system. Spray and Snow exist primarily for compatibility with earlier releases of 3D Studio MAX and are superseded by Super Spray and Blizzard.

These are the basic steps for creating aparticle system:

- Create a particle emitter. All particle systems require an emitter. Some particle systems use the particle system icon as the emitter while others use an object you pick from the scene as the emitter.
- Determine the amount of particles. You set parameters such as birth rate and life span to control how many particles can exist at any given time.
- Set particle shape and size. You can select from many standard particle types (including metaballs) or you can pick an object to be emitted as a particle.
- Set initial particle motion. You can set the speed, direction, rotation, and randomness of particles as they leave the emitter. Particles can also be affected by animation of the emitter as well.
- Modify particle motion. You can further modify the motion of particles after they leave the emitter by binding the particle system to a space warp.

Rain and Snow

Create rain and snow using Super Spray, and Blizzard. These particle systems are optimized for droplet (Super Spray) and tumbling flake (Blizzard) effects. Add space warps such as wind to create spring rains or winter storms.

Bubbles

Create bubbles by using the Bubble Motion parameters of Super Spray. If you require good rendering speed, consider using constant or tetra particles. If you require bubble detail, consider using opacity mapped facing particles, instanced spheres, or metaparticles.

Flowing Water

You generate flowing liquid effects by setting Super Spray to generate closely packed metaparticles. The metaparticles blob together forming a stream. Add a Path Follow space warp to send the stream down a trough.

Explosions

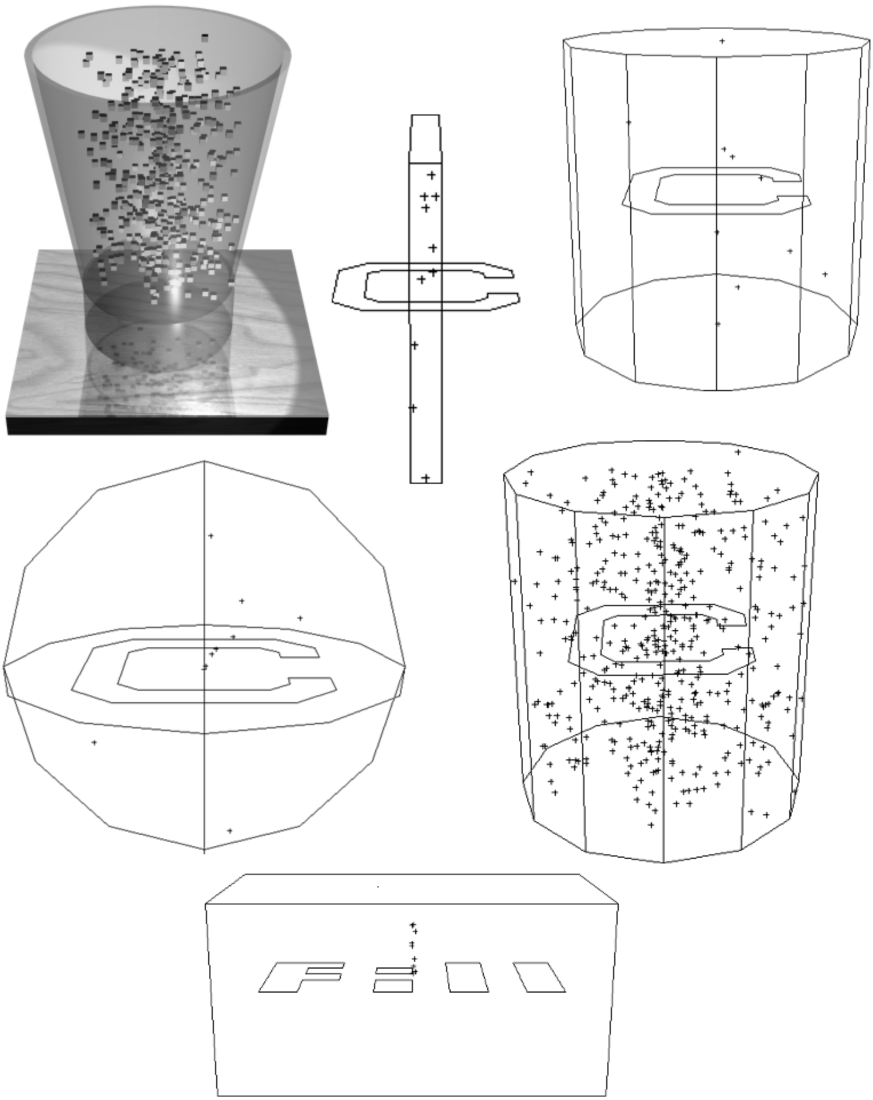
Particle Array uses another object as its particle emitter. You can set the particle type to use fragments of the emitter object to simulate the object exploding.

Volume Effects

Particle Cloud constrains its particles within a specified volume. You can use Particle Cloud to generate bubbles in a glass of soda, or bees buzzing inside a jar.

Crowds

Super Spray, Blizzard, Particle Array, and Particle Cloud all have the ability to use instanced geometry as their particle type. You can create a stream of ants, a flock of birds, or a cloud of dandelion seeds using instanced geometry particles.



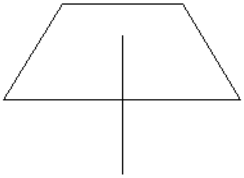
Particle Cloud Emitters constrain particles within a specific volume

Creating a Particle Emitter

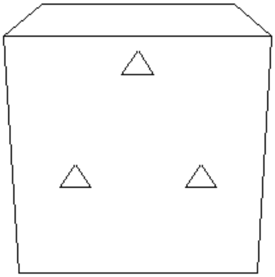
After you click one of the particle systems on the create panel, you create the particle system icon in the scene. The icon serves different purposes depending on the type of particle system you create.

- As an emitter the icon defines the starting location and direction of the particles. Spray, Snow, Super Spray, Blizzard, and Particle Cloud use the icon as the particle emitter.
- As a place holder the icon only serves to hold the parameters for the particle system. The particles are emitted from another selected object. Particle Array and Particle Cloud use the icon as a place holder.

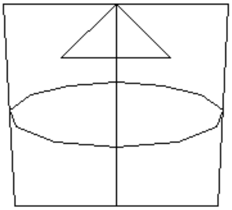
You can choose whether Particle Cloud uses its icon or another selected object as the emitter.



Spray, Snow, or Blizzards



PArray



PCloud

Creating Emitter Icons

Drag in a viewport to set the size and orientation of the particle emitter icon for Spray, Snow, Super Spray, Blizzard, and Particle Cloud.

- Spray, Snow, Blizzard, and Particle Cloud use the icon size as the area of particle emission. Super Spray emits particles from it center regardless of its icon size.
- All of the particle systems align the particle direction with Z axis of the creation grid.

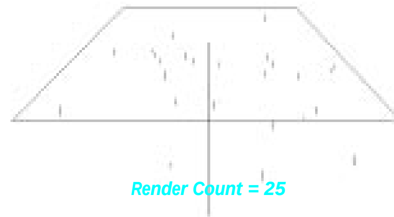
Creating Place Holder Icons

Drag in a viewport to set the size of the place holder icon for Particle Array and Particle Cloud.

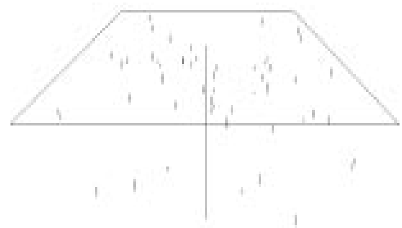
After placing the particle system icon, click Pick Object on the Basic Parameters rollout, to pick the object you want to use as the particle emitter. The size and location of the particle system icon has no effect on the particles.

Controlling Particle Amount

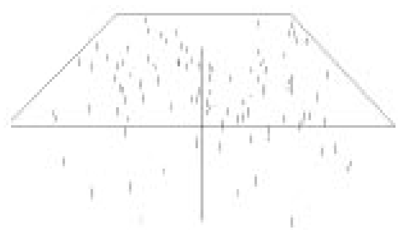
Particle systems quickly generate hundreds or thousands of particles. You control how many particles are created by setting parameters for the maximum number of particles allowed, particle birth rate, and particle life span.



Render Count = 25



Render Count = 50



Render Count = 75

Limiting Total Number of Particles

You can limit the number of particles created by setting a single parameter. Which parameter you use varies with which particle system you create.

Render Count—Spray and Snow use this parameter to limit the number of particles rendered on any frame.

- When the number of particles reaches the Render Count value, particle creation is suspended until some particles die.
- When enough particles die, particle creation resumes until Render Count is reached again.

Use Total—Super Spray, Blizzard, Particle Array, and Particle Cloud use this parameter (on the Particle Generation rollout) to limit the number of particles created over the life of the particle system. Use Total controls the particle birth rate to evenly distribute particle creation.

Viewing Particles in the Viewports

You can limit the number of particles displayed in the viewports to improve display performance. Spray and Snow use a Viewport Count parameter, while the remaining particle systems display a percentage of the total particles.

Set the display percentage to 100 percent if you want to see the same number of particles as will be rendered in your scene.

Setting Particle Birth Rate and Life Span

You set parameters for particle birth rate, life, and start and end times to control the speed and duration of the particle system.

Setting Particle Birth Rate

You set a rate parameter to control how many particles are created on each frame of animation. Which parameter you use varies with which particle system you create.

Birth Rate—Spray and Snow use this parameter. The Birth Rate drops to 0 when the total particles in the scene equal the Render Count. When enough particles die the Birth Rate resumes.

If the Birth Rate is high compared to the Render Count, particles are created in repeated bursts.

Use Rate—Super Spray, Blizzard, Particle Array, and Particle Cloud use this parameter. When you enable Use Rate the Count parameter is ignored and particles are continually created at the set rate.

Setting Life Span

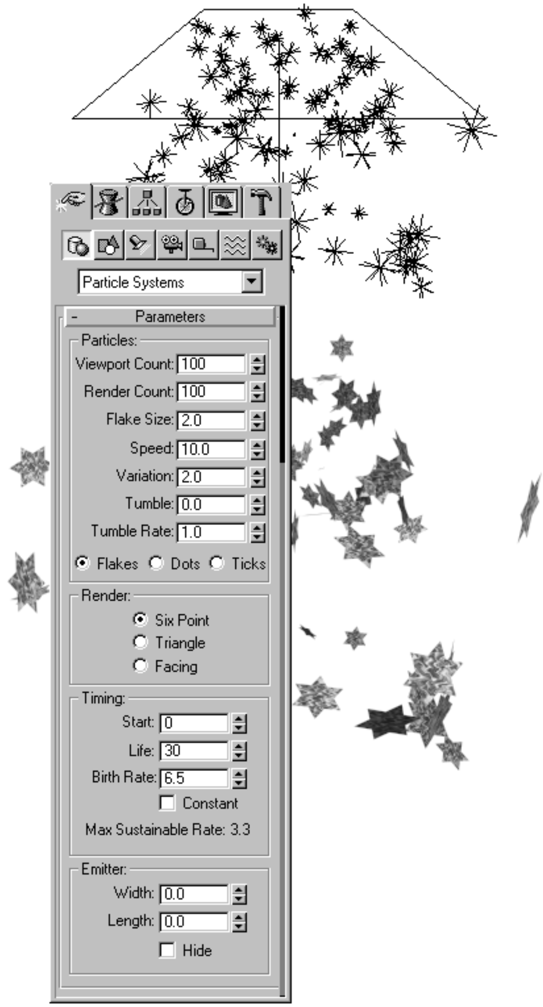
Set the Life parameter of a particle system to specify how many frames a particle lasts after its birth.

Setting Particle System Duration

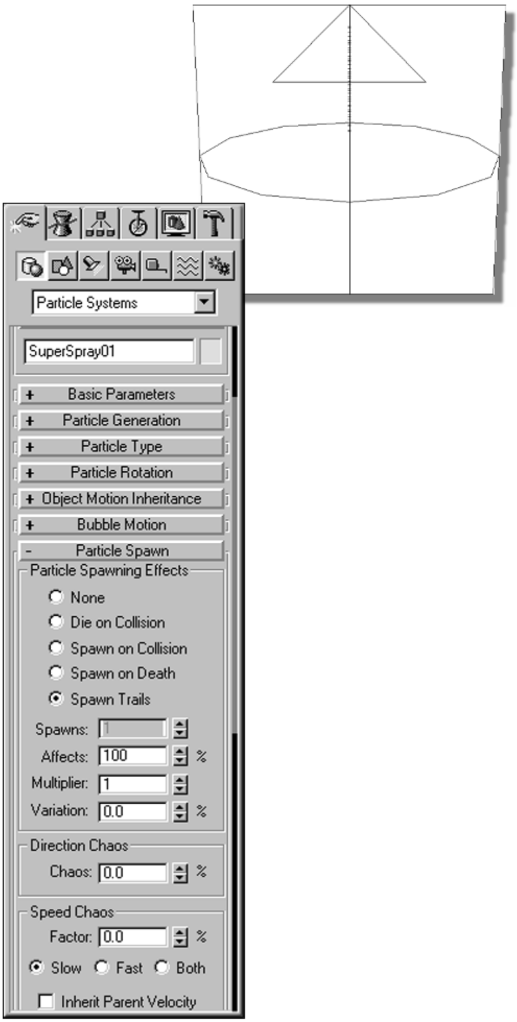
You can also limit the number of frames in which a particle system is active. Which parameters you use varies with which particle system you create.

Start—Spray and Snow use this parameter to set when particle creation starts. You stop particle creation by animating the Birth Rate parameter to 0.

Emit Start/Emit Stop—Super Spray, Blizzard, Particle Array, and Particle Cloud use these parameters to control when particle creation starts and stops.



Spawning Particles



You can also set Particle Spawn parameters to cause the creation of more particles as the original particles die or collide with deflectors. Spawned particles are not considered part of the Use Total particle count.

Spawning is available for only Super Spray, Blizzard, Particle Array, and Particle Cloud.

Choosing When Particles Spawn

You choose from three options to spawn particles. You can also shorten particle life spans without spawning or disable spawning all together. The options that spawn particles are:

Spawn on Collision—Particles spawn when they collide with a deflector space warp.

Spawn on Death—Particles spawn at the end of their life.

Spawn Trails—Particles spawn on each frame of their life.

Limiting Spawned Particles

Spawning can quickly load your scene with many extra particles. You set parameters in the Particle Spawn rollout to control how many particles are capable of spawning, the number of spawn generations, and how many particles are created when a particle spawns. See the Online Reference for details about particle spawning.

Setting Other Spawning Options

You have many options to control the following properties of spawned particles.

- Direction and speed can vary randomly.
- Scale can vary randomly.
- Instanced geometry particles can mutate from one instance to another.
- Life span can vary for each generation.

Choosing Particle Shape and Size

You have complete control over the size and shape of particles. You can also specify that particles grow and fade over their life span. Particle size parameters are found on the Particle Generation rollout. Particle shape parameters are found on the Particle Type rollout.

Setting Particle Size

You set parameters to control particle size and growth.

Size—Usually specifies particle size in units. Size can also represent pixels or a percentage for some particle types.

Variation—A percentage by which particle size can vary. You use variation to get a realistic mix of large and small particles.

Grow For—Controls the rate at which particles grow up to their specified size after they are created.You use this parameter to simulate natural effects such as bubbles growing as they reach the surface.

Fade For—Controls the rate at which particles shrink down to nothing before they die. You use this parameter to simulate natural effects such as sparks fading to ash.

Choosing Standard Particles

Choose one of the standard particle types for quick rendering and good general purpose results. Standard particles are very effective when combined with texture and opacity mapped materials. A few examples include:

- Choose Tetra particles for raindrops or sparks.
- Choose Facing or Special particles with an appropriate opacity map for bubbles or snow flakes.
- Choose Triangle particles with noise opacity for steam or smoke.

Choosing Metaball Particles

Choose metaparticles when you want a close-up view of a liquid particle system. The particles are made from metaballs and you control the properties for how they merge together. Metaparticles take extra time to render but are very effective for spraying and flowing liquids.

Choosing Instanced Geometry Particles

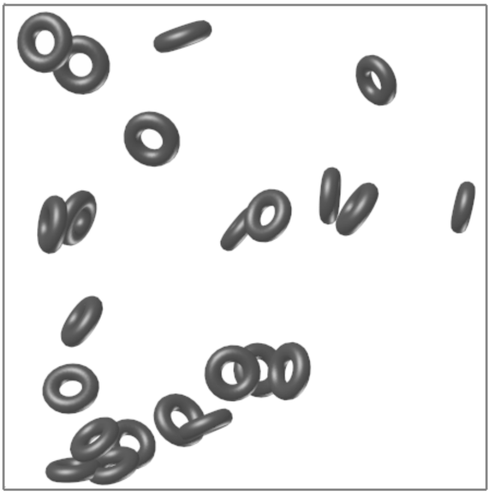
Choose instanced geometry when you want particles to be identical instances of another object in your scene. Instanced geometry particles are extremely effective for creating crowds, flocks, or flows of very detailed objects. A few examples include:

- Instance a red blood cell and use Super Spray to animate blood flowing in an artery.
- Instance a bird and use Particle Cloud to animate a flock of birds flying.
- Instance a rock and use Particle Cloud to animate an asteroid field.

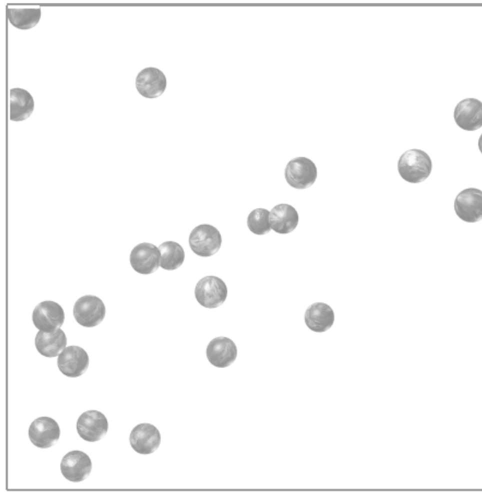
Choosing Object Fragment Particles

Object Fragment is only available with Particle Array. Choose it when you want to fracture the particle emitter object and use the pieces as particles. Object Fragments are useful for animating explosions and shattering collisions.

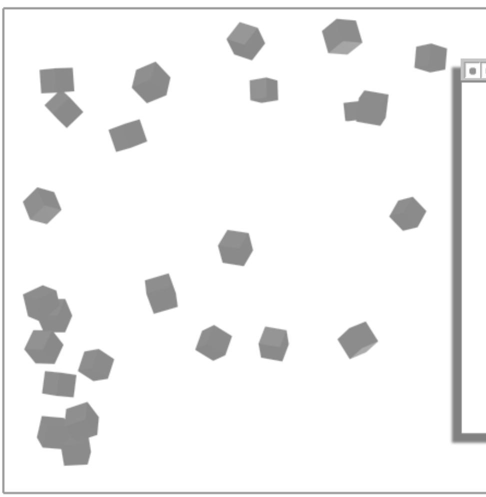
All of the fragments are created on the Emit Start frame. The Use Rate, Use Total, and Emit Stop parameters are disabled.



Instanced Geometry

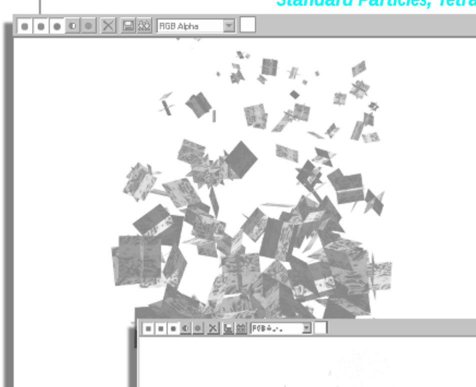


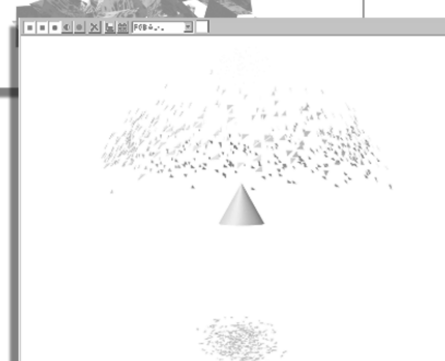
Instanced Geometry



Standard Particles, Cube

Standard Particles, Tetra





Object Fragment Particles

Controlling Particle Motion

You control initial particle motion by animating the particle emitter and setting parameters for particle motion.

Animating Emitter Motion

If you animate the position or rotation of the particle emitter, you can affect particle motion in the following ways:

- Position and direction of particles are determined by the position and orientation of the emitter at the time the particle is created. If the emitter is moving through the scene, particles are scattered along the emitter's path.
- Set parameters on the Object Motion Inheritance rollout to add motion of the particle emitter to the motion of particles. If the Influence parameter is zero, particles leave the emitter at the same speed and direction as if the emitter had been stationary at the time the particle was created.

To illustrate the effect of object motion inheritance, imagine that you are sitting in a moving train and you throw a ball out the window:

- The real world result is that the ball initially travels sideways at the speed with which you threw the ball and forward at the speed of the train. If it were not for gravity and wind, the ball would follow an angled trajectory out and forward from the point where it was thrown. Object Motion Inheritance at 100 percent simulates this condition.
- If Object Motion Inheritance were set to zero percent, the ball would travel straight out from the train with no forward motion.

Setting Particle Speed and Direction

All particle systems have parameters for speed and variation.

- Specify the number of units travelled per frame in the Speed parameter.
- Set the Variation parameter to randomize speed so that all particles are not travelling at the same rate.

Particles travel along the normal vector of the spot where they are emitted. Some particle systems have additional parameters you can set to vary the travel direction and dispersion angle away from the emitter normal vector.

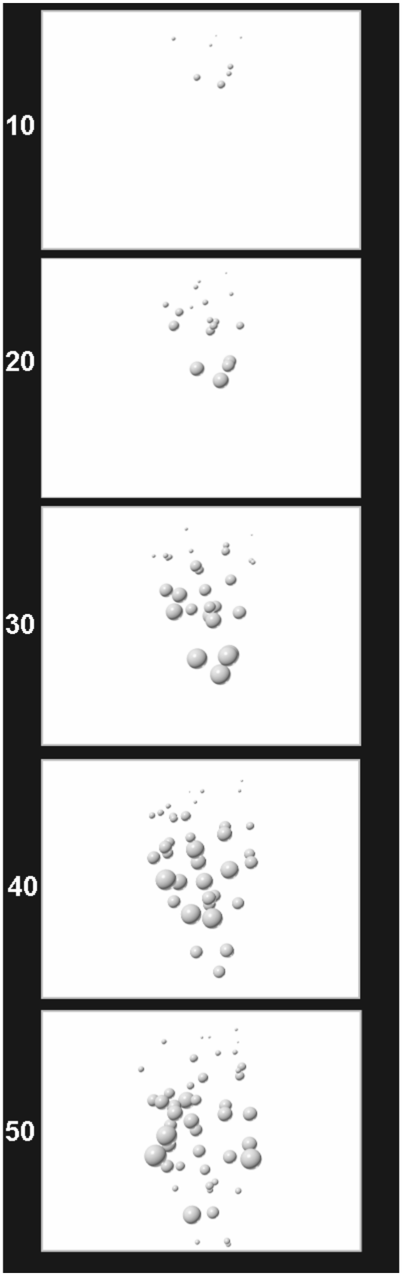
Setting Particle Rotation

Super Spray, Blizzard, Particle Array, and Particle Cloud all contain a rollout named Particle Rotation. You set the Particle Rotation parameters to specify rotation rate, variance, and direction of the rotation axis.

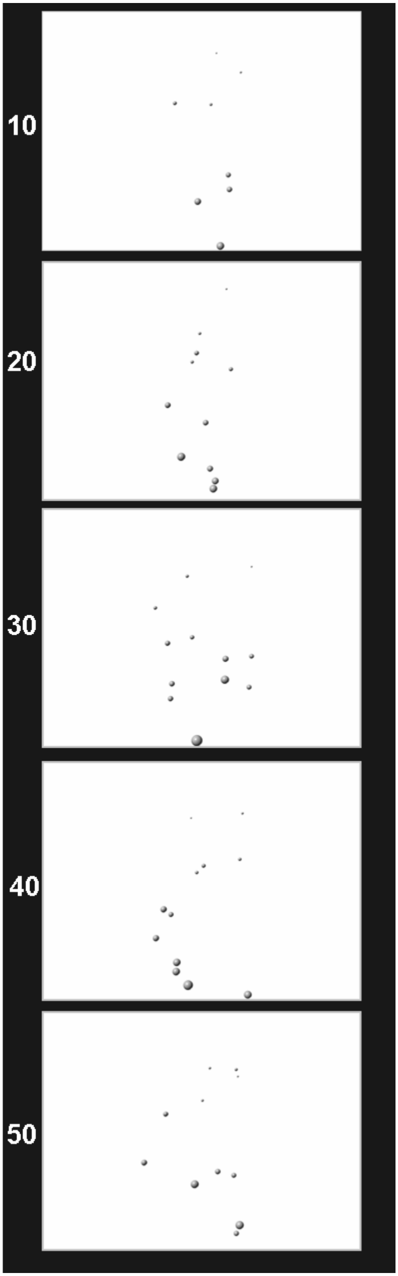
Adding Bubble Motion

Super Spray, Particle Array, and Particle Cloud all contain a rollout named Bubble Motion.You set the Bubble Motion parameters to simulate the wobbling effect you see in bubbles rising underwater. This effect is useful when the particles are set to rise in thin streams.

Bubble motion is similar to a wave form, and the Bubble Motion parameters set the amplitude, period, and phase of the bubble "wave."

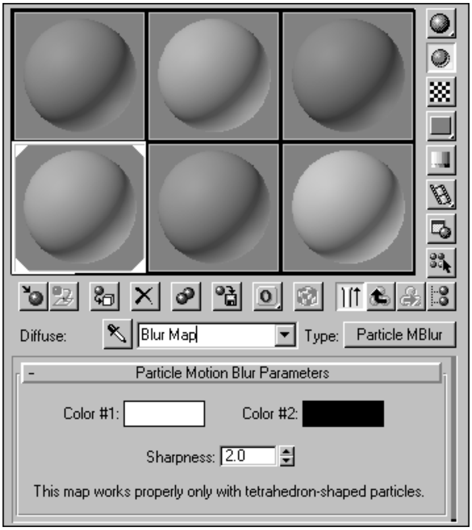


Speed setting = 10



Speed setting = 50

Applying Motion Blur to Particles



Particle MBlur parameters in Material Editor

Particles often move at high rates of speed. In such cases you might want to add motion blur to the particles to enhance their motion.

- Super Spray, Particle Array, and Particle Cloud all support a motion blur technique specific to particle systems.
- You can also apply the standard effects of Object and Scene motion blur to particle systems.

Generating Particle Motion Blur

Particle motion blur is combination of varying the opacity and the length of particles based on their speed. You generate particle motion blur by coordinating material assignment with Particle Rotation parameters.

- Assign the Particle MBlur map as an opacity map of the material assigned to the particle system.
- On the Particle Rotation rollout, select Spin Axis/MBlur and set Stretch greater than zero.
- Select any particle type except Constant or facing particles.

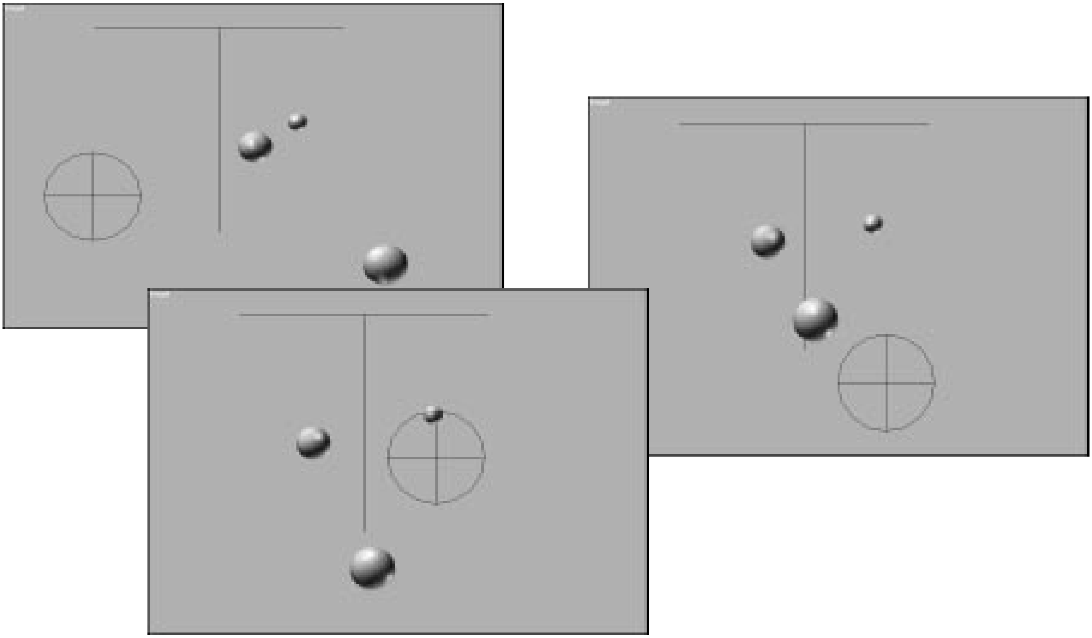
This type of motion blur is very efficient because it uses object scaling and mapping rather than using multiple rendering passes.

Using Object and Scene Motion Blur

You can also assign Object or Scene motion blur to particle systems.

- Standard particles work with both Object and Scene motion blur.
- Metaparticles, Object Fragments, and Instanced Geometry only work with Scene motion blur.

Binding Particles to Space Warps



Placement of particles in relation to a space warp

You can further affect the motion of particles by binding them to space warp objects. Use space warps of the Particles & Dynamics category with particle systems. See chapter 15, "Creating Space Warps."

Note: Particle systems cannot participate in dynamic simulations. Use deflector space warps to simulate collision dynamics.

Deflecting Particles

Some space warps, called deflectors, act as collision objects for particles. Particles can bounce, stick, or slide across the surface of a deflector.

- Deflector bounces particles off of a flat plate.
- SDeflector bounces particles off of a sphere.
- UDeflector bounces particles off of a mesh object.

Applying Forces to Particles

Some space warps apply external forces to particle motion. The speed and direction of the space warp force is added to the speed and direction of the particles.

- Gravity and Wind simulate natural forces.
- Particle Bomb simulates explosive shock waves.
- Motor, Push, and Displace simulate forces of torque and linear resistance.

Constraining Particle Motion

The remaining space warp constrains particle motion. Use Path Follow to cause particles to travel along a spline or NURBS curve.

Chapter

15

Creating Space Warps

Space warps are objects that generate “force field” effects on other objects in the scene. The space warps themselves do not render. You use them to affect the appearance of other objects.

Some space warps deform object geometry by generating ripples, waves, or explosions. Other space warps are meant for use with particle systems and dynamic simulations. They simulate natural effects such as wind blowing snow about, or a rock in the path of a waterfall.

This chapter discusses the uses of space warps. See the Online Reference for details about all space warp objects.

Creating and Binding Space Warps

You create a space warp and bind other objects in your scene to it to generate "force fields" that affect the appearance and movement of the bound objects.

Types of Space Warps

You generate two very different effects depending on which type of space warp you create.

- Geometric/Deformable and Modifier-Based space warps behave somewhat like modifiers. They deform the geometry of bound objects except that a space warp applies its effect in *world space*, rather than *object space*, as a modifier does.
- Particles & Dynamics space warps affect the motion of bound particle systems and of objects in dynamic simulations. These space warps push, pull, spin, and deflect particles and objects.

See the Appendix, "Differences Between Object Space and World Space" for information about the different types of spaces.

Creating Space Warp Objects

You create a space warp the same as any other object. Click a space warp button on the Create panel and then drag in a viewport to define the base size of the space warp.

You can transform and animate the space warp as you do other objects.

Binding Objects to Space Warps

You *bind* an object, or selection of objects, to a space warp to apply the space warp effect. A space warp has no effect on objects unless the objects are bound to it. When an object is bound to a space warp, the space warp binding appears at the top of the object's modifier stack. A space warp is applied *after* any transforms or modifiers.

When you bind multiple objects to a space warp, the space warp affects all the objects equally. However, each object's distance and spatial orientation from the space warp changes the effect. Moving or otherwise transforming an object bound to a space warp changes the effect.

You can also bind an object to multiple space warps. Multiple space warps appear in an object's stack in the order you apply them.

Changing Binding Parameters

Some kinds of space warps have *binding parameters*. These parameters are found on the Modify panel when you select the space warp binding in the modifier stack of a bound object. Changing the parameters changes how a space warp affects the bound object.

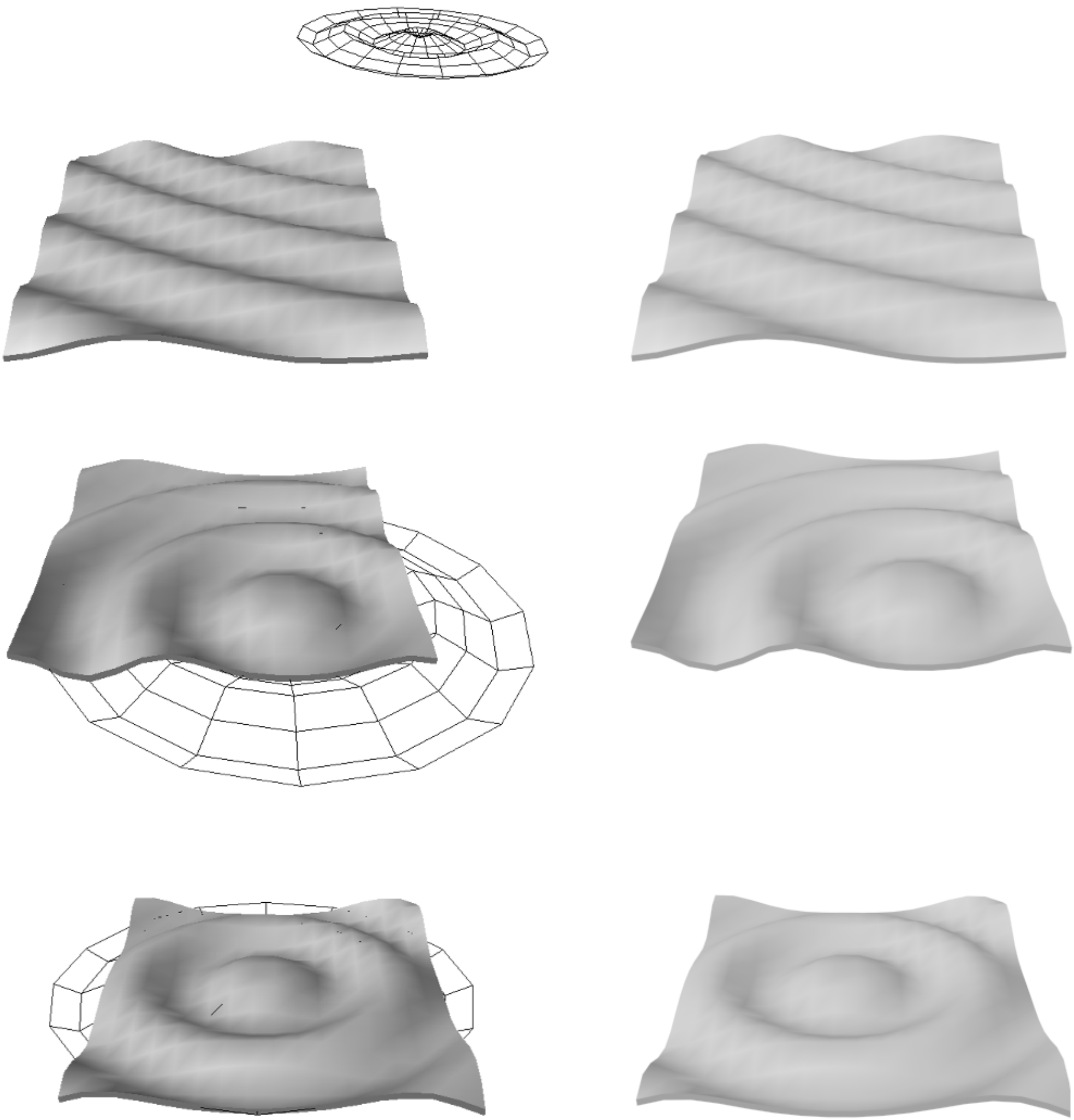
Unbinding Objects

You remove a space warp binding from an object by deleting the binding from the object's modifier stack. Deleting the binding has no effect on the space warp itself.

Space warps are bound to objects individually. When you bind multiple objects to a space warp, the binding appears in each object's stack. You can't unbind a selection of bound objects, even if you bound the objects as a selection.

To unbind multiple objects from a space warp, do one of the following:

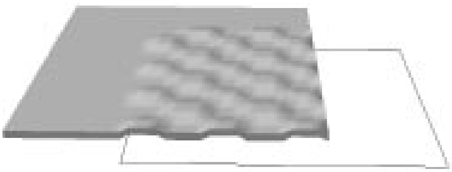
- Delete the space warp binding from each object individually.
- Make a copy of the space warp and then delete the original space warp.



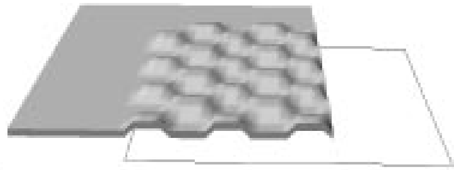
Objects in viewports with ripple deformation space warps on the left. Rendered image of each object on the right

Warping Geometry

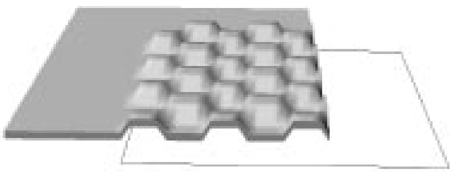
Displacement warp with checkered map



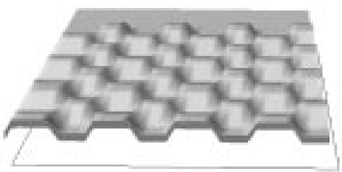
strength set at 2



strength set at 4



strength set at 6



displacement warp moved

You can warp the geometry in your scene using Geometric/Deformable or Modifier-Based space warps. Both categories of space warps work in an identical way.

The primary difference between the two categories is that Geometric/Deformable space warps have custom gizmos, some of which also include sub-objects, while Modifier-Based space warps all use a box as their gizmo.

An object must be deformable, such as a standard primitive, for these space warps to have the correct effect. For example, lights, helpers, and particle systems are not deformable so these space warps have little effect on such objects.

Using Geometric/Deformable Space Warps

You create Geometric/Deformable space warps to generate the following effects:

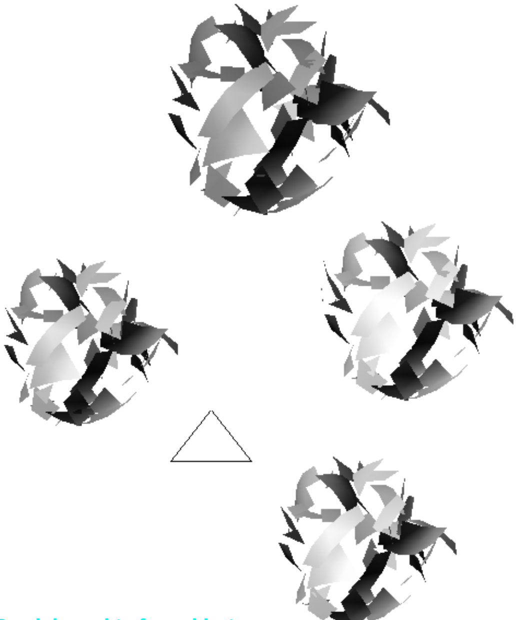
Free Form Deformation (FFD)—Use Box or Cylinder FFD to deform objects by positioning control points of the FFD lattice. The offset position of the control points from the original lattice source generate the deformation of the bound objects.

Wave Form Deformation—Use Wave or Ripple to pass a wave through objects. Wave creates a linear wave while Ripple creates a concentric ring wave through the bound objects.

Surface Displacement—Use Displace to push against a surface using a planar, cylindrical, or spherical gizmo. The displacement effect can be uniform across the gizmo or it can vary based on the gray scale value of a bitmap. By rotating, scaling, and moving the gizmo, you can change the displace effect.

Surface Matching—Use Conform to push bound object vertices in a specified direction until they hit a target object or travel a specified distance. The bound objects end up flattening themselves against the surface of the target object.

Explosion—Use Bomb to explode objects into individual faces that fly away from the center of the Bomb gizmo. For many explosive effects



Bomb bound to four objects

you may want to consider using the Particle Array particle system rather than Bomb.

Using Modifier-Based Space Warps

You create Modifier-Based space warps when you want to duplicate modifier deformation but apply the deformation in world space.

The Modifier-Based space warps are Bend, Taper, Noise, Twist, Skew, and Stretch.

Unlike modifiers, the deformation caused by Modifier-Based space warps varies as the bound objects move around the scene.

Warping Particles and Dynamic Simulations

You create particle system space warps to apply forces to particle systems and to objects in dynamic simulations.

Binding Particle Systems to Space Warps

You bind particle system icons to particle system space warps to apply the effect to emitted particles. The particle system icons are not affected by the space warp. See chapter 14, “Modeling Particle Systems” for information about creating particle systems.

Particle system space warps include the following effects:

Natural forces—Use Gravity and Wind to simulate these natural forces.

- You can change the strength and direction of Gravity to simulate non-Earthly conditions.
- Wind includes parameters for strength and turbulence.

Explosions—Use Particle Bomb to generate explosive shock waves. The shock wave scatters particles in front of it. If you bind a Particle Array particle system, set to use object fragments, to Particle Bomb space warp, you can create convincing explosive effects.

Mechanical forces—Use Motor, Push, and Displace to simulate forces of torque and resistance.

- Motor applies torque to a particle system, spinning the particles around its center.
- Push applies a constant linear force in a single direction over a specified range.
- Displace applies a force radiating from a planar, cylindrical, or spherical gizmo. The force can be constant or can vary based on the grayscale value of a bitmap.

Deflection—Use the deflectors as collision objects for particles. Particles can bounce, stick, or slide across the surface of a deflector.

- Deflector bounces particles off a flat plate.
- SDeflector bounces particles off a sphere.
- UDeflector bounces particles off a mesh object.

Motion Constraint—Use Path Follow to constrain particle motion along a spline or NURBS curve.

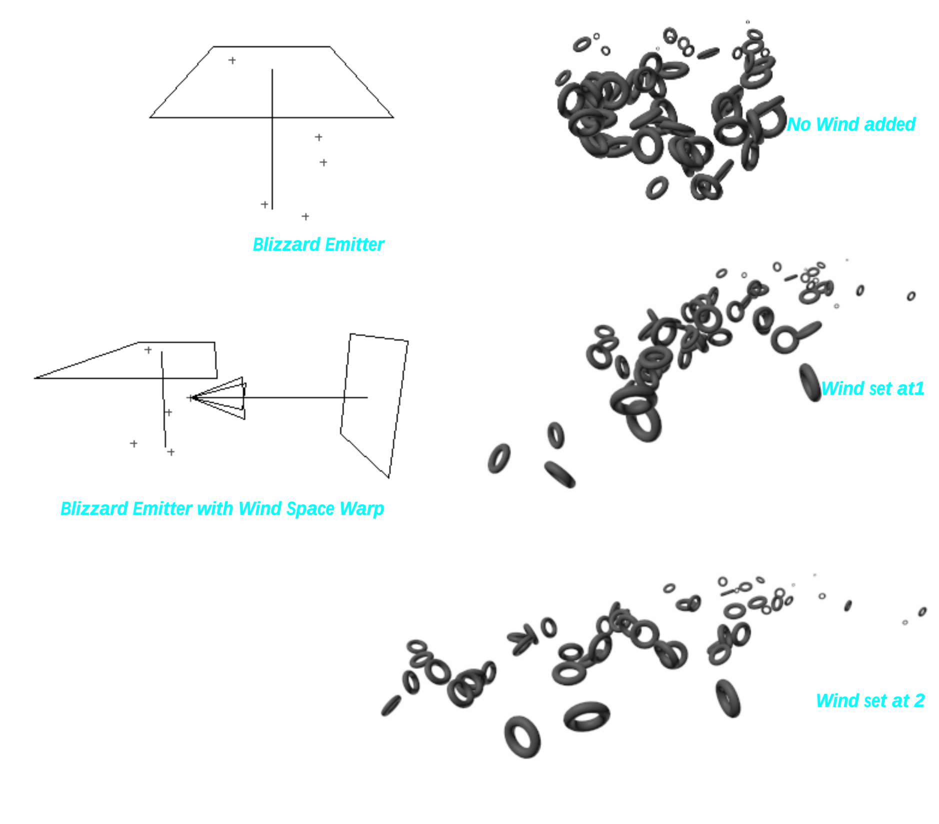
Adding Space Warps to Dynamic Simulations

Some of the Particles & Dynamics space warps can be added to dynamic simulations. You do not need to bind objects to the space warps in this case. Instead you add the space warp as an effect that participates in the simulation. See chapter 28, “Animating Dynamic Simulations.”

Particle system space warps that can be used in dynamic simulations include Gravity, Wind, Particle Bomb, Motor, and Push.

These space warps, with the exception of Push, have much the same effect on objects in a dynamic simulation as they do on particles. Push acts as a point force, like pushing against an object with your finger, rather than spreading its force over a specified range.

Blizzard particles with and without Wind



Chapter

16

Creating Systems

A *system* combines objects, linkages, and controllers to produce an object set that has a behavior as well as geometry. Systems enable you to create animations that would be much more difficult or time-consuming to produce using other 3D Studio MAX features independently.

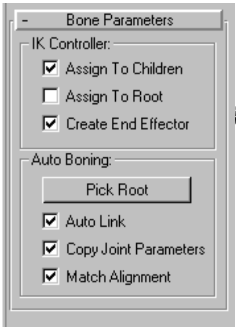
Ranging from simple object generators to full-scale subsystem programs, systems are primarily intended for use as plug-in component software.

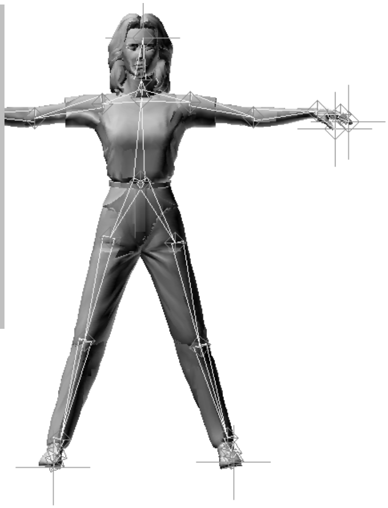
3D Studio MAX includes three systems:

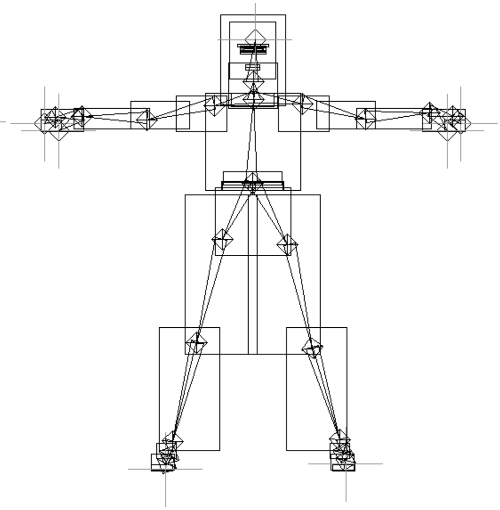
- Bones
- Ring Array
- Sunlight

This chapter introduces these systems. See online reference for further details.

Bones System







A Bones system is a jointed, hierarchical linkage of "bone" objects that demonstrates kinematics. Each "bone" is a parametric wireframe object. You can't render bones or modify individual bone objects.

Bones and the New IK

In 3D Studio MAX, the Bones system forms the basis for the New IK. Using this feature, you create Bones with an IK controller to apply IK solutions procedurally across all frames.

See chapter 24, "Using Inverse Kinematics," for a complete introduction to the New IK as well as standard IK.

Using Auto Boning

You can use Auto Boning to automatically assign bones to objects in an existing hierarchy. Auto Boning is handy when you want to convert a regular linked hierarchy to a bone structure using the IK controller.

For details on creating hierarchies with Bones and using Auto Boning, see chapter 23, "Building Hierarchies."

Creating Bones Without Controllers

Using the default settings for Bones, you automatically create a New IK system, with an IK controller applied to the transform of each bone below the root.

The following procedure creates a "plain" Bones hierarchy *without* IK controllers for use with forward kinematics and standard IK. The first bone you create is at the top of the hierarchy. The last bone you create is at the bottom.

To create a Bones system without controllers:

1.  On the Create panel, click Systems. On the Object Type rollout, click Bones to display the default creation parameters for Bones.
2. In the IK Controller area, clear Assign To Children. This grays out the other options in this area, disabling all New IK controls.
3. Click in a viewport to define the base of the hierarchy. Drag to define the length of the first bone.
4. Continue to click and drag to create additional bones in a hierarchical chain. Right-click to close the chain.

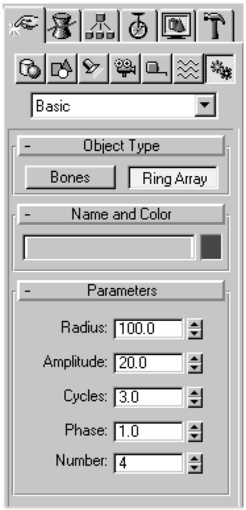
Transforming Bones Without Controllers

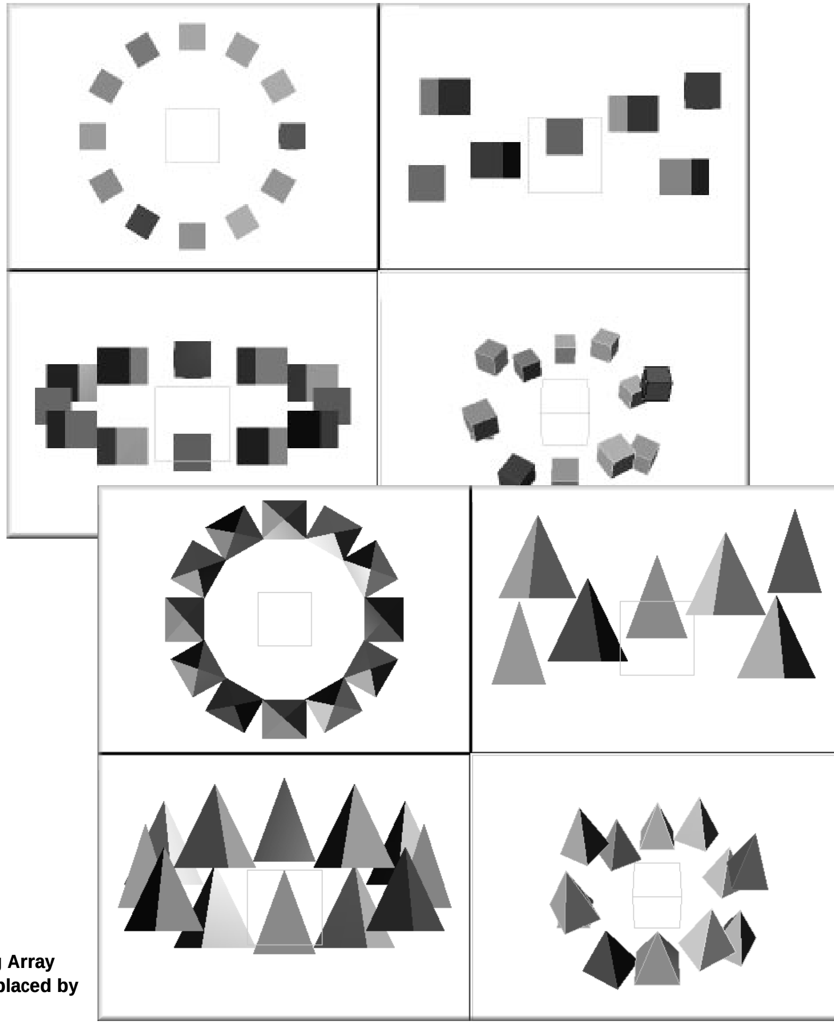
You can move, rotate, and scale the bones in a Bones system created without controllers. The effect depends on whether Inverse Kinematics (IK) is active or not.

- *With IK off*, when you transform an individual bone you are actually transforming the "joint" at its end. Moving a bone moves the joint and changes the bone's length.
- *With IK on*, the length of each bone remains the same, and the system behaves like an armature.

In either case, bones remain connected to each other. Transforms you apply to one bone can propagate to other bones in the system. With IK off, the propagation is always to bones that are lower in the hierarchy. With IK on, the propagation can be in either direction along the hierarchy.

Ring Array System





Top: Original Ring Array
Bottom: Boxes replaced by pyramids

A ring array is a dummy object linked as a parent to a ring of boxes arranged along a sine curve. You can vary the number of boxes and animate the ring array's parameters. You can also replace the boxes with other objects.

Creating a Ring Array System

By default, a ring array consists of four boxes evenly spaced in a circle around a dummy object. You cannot change the number of boxes after you create the array. The following procedure therefore includes a step to do this.

To create a Ring Array system:

1. On the Create panel, click Systems. On the Object Type rollout, click Ring Array.
2. Change the value for Number to increase or decrease the number of boxes generated. The minimum is one box.
3. Drag in a viewport to set the center and radius of the array. A dummy object (wireframe cube) appears at the center, with boxes surrounding it.

Ring Array Parameters

After creation, Ring Array parameters appear on the Motion panel (not the Modify panel). The following parameters control ring arrays:

Radius—Sets radius of the ring. You set the initial value when you drag to create the ring array.

Amplitude—Sets amplitude of the ring's sine curve in active units. Amplitude is a height offset from the local origin of the center dummy object.

Cycles—Sets number of cycles in the ring's sine curve. When Cycles equals 0.0, the ring is flat. When Cycles equals 1.0, the ring is tilted. Greater values increase the number of peaks in the curve.

Phase—Offsets the phase of the wave. That is, it has the effect of moving the wave along the circumference of the ring. Whole values have no effect—only fractional values do.

Number—Sets number of boxes in the ring.

Animating the Ring Array

You can animate the ring array parameters. However, you can't animate the number of boxes in the ring.

To animate a Ring Array system:

1. Select any box in the array.
2. Turn on the Animate button and move to a non-zero frame.
3. Adjust Ring Array parameters on the Motion panel.
4. Repeat for other frames.

Putting Other Objects in the Ring

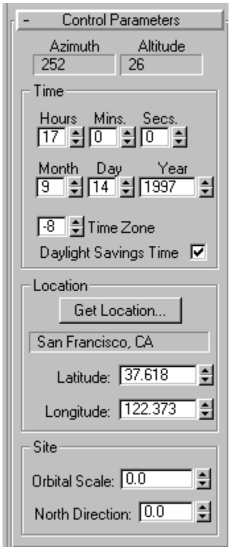
To replace the boxes in the ring with other kinds of objects, you use Cut and Paste in Track View.

To replace boxes with other objects:

1. In the Track View hierarchy, select the object container of the object you want to put in the ring.
2. Click Copy.
3. Select the object container of one or more ring array boxes.
4. Click Paste. On the Paste dialog, choose Copy or Instance. If you want to replace every box, set Replace All Instances. Click OK.

For more information about copying and pasting in Track View, see chapter 25, "Basic Track View Use."

Sunlight System



The Sunlight System creates and animates realistic "sunlight" in a 3D Studio MAX scene.

The "sun" is a Target Direct light, targeted on a non-renderable compass rose showing the cardinal points. In this system, the sun is linked as a child to the compass rose.

The sun follows the geographically correct angle and movement of the sun over the earth at a given location. You can choose location, date, time, and compass orientation. Date and time can be animated.

Shadows are rendered accurately, making this system suitable for shadow studies of proposed and existing structures.

Creating a Sunlight System

You generally want to create a Sunlight system on the ground plane (representing the surface of the earth). These are the basic steps.

To create a Sunlight system:

1. Set a viewport to Perspective or Top.

2.  On the Create panel, click Systems. On the Object Type rollout, click Sunlight.

3. Click down at a point on the ground plane of your scene. Drag to set the radius of the compass rose. Release and drag to set the Orbital Scale of the sun.

4. Click to complete the system. The light representing the sun is positioned according to the current time settings on your computer.

The Orbital Scale can be any convenient distance, since the light produces parallel illumination wherever it's located. The non-renderable compass rose should be large enough to easily read the cardinal directions.

Setting Control Parameters

After creation, the parameters for the sun appear on the Motion panel. As you make parameter changes, the "sun" shifts in the scene to correspond to the new settings. See online reference for detailed procedures.

Setting Geographic Location

You can locate your scene anywhere in the world—by picking from a map, a list of major cities, or by entering a latitude and longitude.

Setting Date and Time

The Time Zone is preset if you have already chosen a location. Daylight Savings Time is cleared unless your computer's operating system supplies this data, but can be set manually.

To set date and time:

- In the Time area, set the month, day, and year.
- Set hours, minutes, and seconds. This setting uses 24-hour mode (6:00:00 is 6 A.M. and 18:00:00 is 6 P.M.).

Azimuth and Altitude read out automatically for the current date, time, and location, and cannot be changed.

Setting Orientation

For real-world projects, the orientation of North in your scene is critical. If you have a site plan, make sure the North of your scene corresponds.

To set North, do one of the following:

- Set North Direction as an angle off zero. By default, North is 0 and points along the positive Y-axis of the construction plane. A value of 90 points North along the positive X-axis.
- Select the compass rose object and rotate it in the XY plane.

Animating Date and Time

Animation is a basic way to study the effect of sunlight in your scene. You can animate sun motion and shadows during a single day or between dates, or both.

Before animating, the location, date, time, and compass orientation should be set for your scene. Realistic results depend on these settings.

Animation follows standard procedures. To animate time over a day, you might use sunrise, or 6 A.M. (6:00:00), as the starting time, and sunset, or 6 P.M. (18:00:00), as the ending time.

Rendering Sunlight and Shadows

To see the shadow-casting effects of a Sunlight system, you need to render the scene. You can render to an animation or set up your scene to render one image at a time.

Rendering a single frame gives you an accurate “snapshot” for documentation or client presentations.

Rendering an animation lets you see the motion of sunlight and shadow in your scene, or shows effects that wouldn't be apparent in stills.

Note: The Target Direct light has a default for Falloff that might need to be increased. If the hotspot isn't wide enough, you won't see rendered shadows for all objects in your scene. See online reference for this procedure and other rendering tips.

Chapter

17

Lighting Your Scene

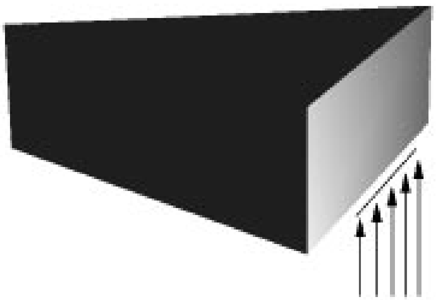
Lights are 3D Studio MAX objects that simulate household or office lamps, the light instruments used in stage and film work, the sun itself, and so on. Different kinds of light objects cast light in different ways, simulating different kinds of real-world light sources.

When there are no lights in a scene, 3D Studio MAX shades it or renders it with default lighting. You add lights to give the scene a more realistic appearance. Good lighting enhances the clarity and three-dimensionality of a scene.

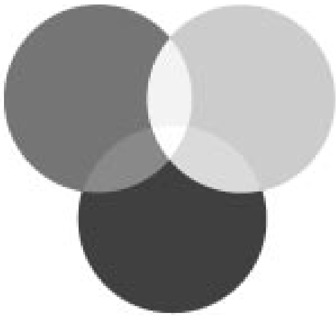
Light objects replace the default lighting. As soon as you create a light, the default lighting is turned off. If you delete all the lights in the scene, default lighting is turned back on. The default lighting consists of two invisible lights: one is above and to the left of the scene, and the other is below and to the right.

See the Online Reference for details about creating light objects.

Properties of Lighting Your SceneLight



Angle of incidence affects intensity



Additive mixing of colored lights

When you light a 3D Studio MAX scene, it can be helpful to know how light naturally behaves.

When light rays strike a surface, some of the rays are reflected enabling us to see the surface. The appearance of a surface depends on the light that strikes it combined with the properties of the surface material: color, smoothness, opacity, and so on. See chapter 20, “Designing Basic Materials” for information about specifying material properties.

Intensity

The intensity of light at its point of origin affects how brightly the light illuminates an object. A dim light cast on a brightly colored object shows dim colors.

Angle of Incidence

The more a surface inclines away from a light source, the less light it receives and the darker it appears. The angle relative to the light source is known as the *angle of incidence*.

When the angle of incidence is 90 degrees, the surface is illuminated with the full intensity of the light source. As the angle of incidence diverges from 90 degrees, the intensity of illumination decreases.

Attenuation

In the real world, light diminishes over distance. Objects far from the light source appear darker; objects near the source appear brighter. This effect is known as *attenuation*.

In nature, light attenuates at an inverse square rate—that is, its intensity diminishes in proportion to the square of the distance from the light source. It is common for attenuation to be even greater when light is dispersed by the atmosphere, especially when there are dust particles in the atmosphere, or fog or clouds.

Radiosity and Ambient Light

The light an object reflects can illuminate other objects. This effect is known as *radiosity.* The more light a surface reflects, the more light it contributes to illuminating other objects.

Radiosity creates *ambient light*. Ambient light has a uniform intensity and is uniformly diffuse. It has no discernible source and no discernible direction.

Color and Light

The color of light depends partly on the process that generates the light—a tungsten lamp casts orange-yellow light; a mercury vapor lamp casts cold blue-white light; and sunlight is yellow-white. Light color also depends on the medium the light passes through. For example, clouds in the atmosphere tint daylight blue, and stained glass tints light a highly saturated color.

Light colors are *additive colors;* the primary light colors are red, green, and blue (RGB). As multiple colored lights mix together, the total light in the scene gets lighter and eventually turns white.

Color Temperature

Color *temperature* describes a color in terms of degrees Kelvin. This is useful for describing the color of light sources and other color values that are close to white. The following table shows color temperatures for some common types of light, with the equivalent hue number (from the HSV color description).

If you use these hue numbers for lights in a scene, set the value to full (255) and then adjust the saturation to meet the needs of your scene. Mentally we tend to correct light color so that objects appear to be lit by white light; usually the effect of color temperature in a scene should be subtle.

Light source	Color temperature	Hue
Overcast daylight	6000 K	130
Noontime sunlight	5000 K	58
White fluorescent	4000 K	27
Tungsten/halogen lamp	3300 K	20
Incandescent lamp (100–200 W)	2900 K	16
Incandescent lamp (25 W)	2500 K	12
Sunlight at sunset or sunrise	2000 K	7
Candle flame	1750 K	5

Guidelines for Lighting

The guidelines for lighting used by photographers, filmmakers, and stage designers can also help you set up the lighting for scenes in 3D Studio MAX.

Your choice of lighting depends on whether your scene simulates natural or artificial light. Naturally lit scenes such as daylight or moonlight, get their most important illumination from a single light source. Artificially lit scenes often have multiple light sources of similar intensity. Both kinds of scenes require multiple secondary lights for effective illumination.

Natural Light

Sunlight and moonlight have parallel rays coming from a single direction. The direction and angle vary depending on the time of day, the latitude, and the season.

In clear weather, the color of sunlight is a pale yellow: for example, RGB values of 250, 255, 175 (HSV 45, 80, 255). Cloudy weather can tint sunlight blue, shading into dark gray for stormy weather. Particles in the air can give sunlight an orange or brownish tint. At sunrise and sunset, the color can be more orange or red than yellow.

3DS MAX includes a *Directional* light object to simulate the sun. A single Directional light is appropriate as the main light source for sunlit scenes.

A Directional light can also simulate moonlight, which is white but dim compared to the sun.

Artificial Light

Artificial light uses multiple light sources. The following guidelines are for creating well lit, legible scenes.

- The subject of a scene should be lit by a single bright light, known as the *key light*. Position the key light in front of the subject and slightly above.
- In addition to the key light, position one or more other lights to illuminate the background and the side of the subject. These are known as *fill lights*. Fill lights are less bright than the key light.
- When you use only one fill light, the angle at ground level between it, the subject, and the key light should be approximately 90 degrees.

Key lighting and fill lighting emphasize the subject of a scene. They also emphasizes the three-dimensionality of the scene.

A *spotlight* is usually best for the key light, and either spotlights or *Omni lights* are good for creating the fill lighting. Ambient light can be another element of your fill lighting.

You can also add lights to emphasize secondary subjects in a scene. In stage terminology, these lights are known as *specials*. Special lights are usually brighter than the fill light but less bright than the main key light.

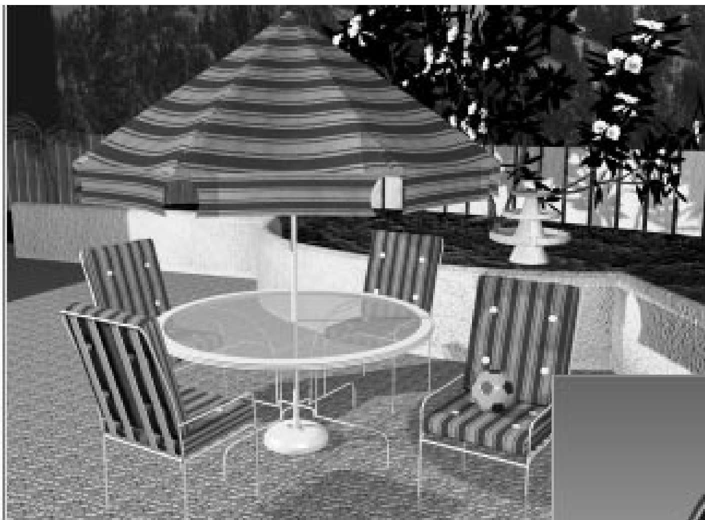
Ambient Light

Ambient light in 3D Studio MAX simulates background radiosity.

The color of ambient light tints the scene and usually should be the complement of the color of the principal light source for the scene.

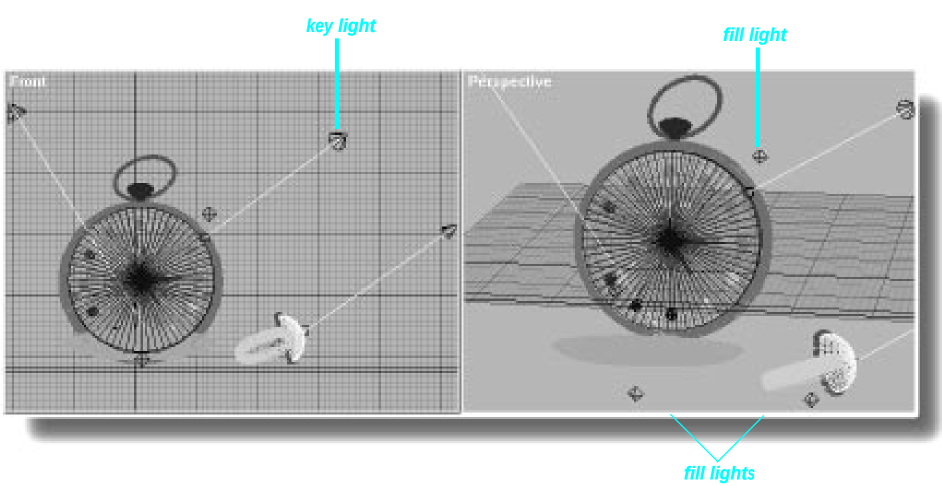
Because 3DS MAX doesn't calculate radiosity based on light bounced from objects in the scene, don't rely on ambient light to create the full effect of radiosity. Usually you need to add additional light objects to create a well-lit scene.

The intensity of ambient light affects contrast as well as overall illumination—the higher the intensity of ambient light, the lower the contrast. This is because ambient light is completely diffuse, so the angle of incidence is equal for all faces.

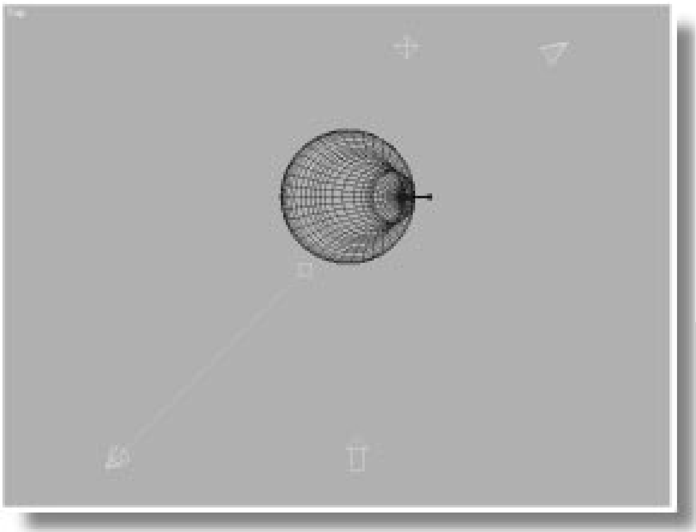


Outdoor scene in daylight

Special light to emphasize portions of a scene



Creating Light Objects



You create light objects from the Lights category on the Create panel. Click a light button, then click or drag in a viewport. You can then set the light parameters on the Create panel or later select the light and set the parameters on the Modify panel.

You can create these light types:

Omni lights—Unfocused light in all directions. An Omni light is like an unshaded light bulb or a star.

Directional lights—Parallel light rays in a single direction as the sun does at the surface of the earth.

Spotlights—Focused beam of light toward a specified target point.

All lights can cast shadows and project images.

Light objects replace the default lighting. As soon as you create a light, the default lighting is turned off. If you delete all lights in the scene, the default lighting is restored.

Unless you turn on shadow casting, light illuminates each object equally, as if no other objects were in the scene. Shadows are a result of geometry blocking light, and if shadows are turned off, light passes through objects (even though they might be rendered as solid).

Free versus Target Lights

You can create directional lights and spotlights as either *free* or *target* lights. The forms of the lights differ in how you place them in your scene, and each has its own benefits.

Free lights have the following properties:

- You create a free light by clicking in a viewport. Where you click locates the light. The light is aimed along the negative Z axis of the viewport you click in.
- Free lights are easier to animate when the light is moving, especially when you want the light to follow a path and bank on corners.

Target lights have the following properties:

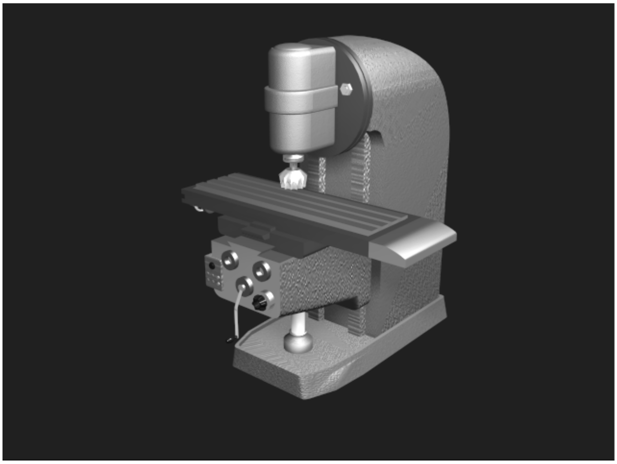
- You create the light by dragging in a viewport. Where you begin the drag locates the light. Where you release the drag locates the target.
- You can position and animate the light and the target independently, but the light always points at its target.
- Target lights are easier to aim and animate when the light remains still and tracks a moving object.
- You can select both a light and its target by clicking the line connecting the two.

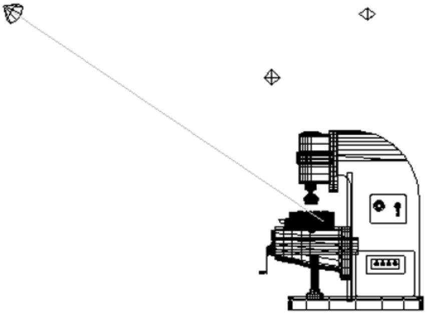
Light Object Icons

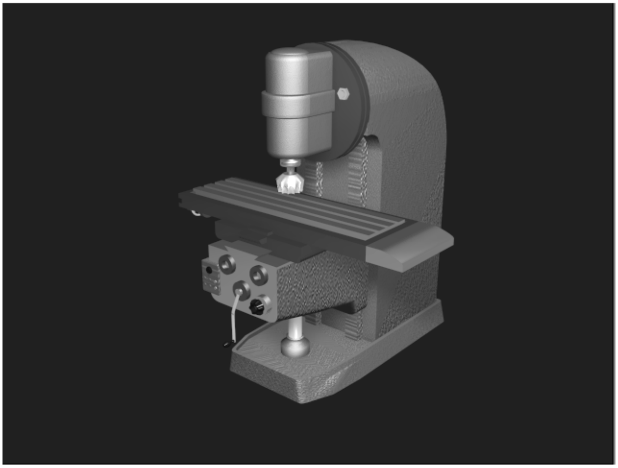
Light objects are visible in viewports unless you turn off their display. However, the geometry that appears in the viewport is only an icon meant to show you where the light object is located and how it is oriented. You cannot render light objects, and the display of light object icons is not scaled when you change the scale of the viewport.

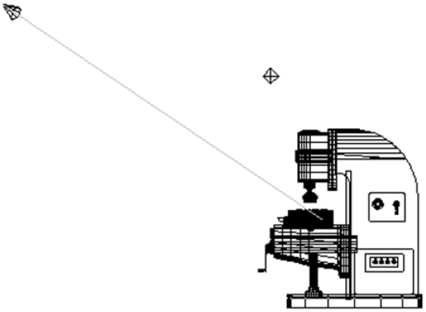
Note: To change the size of light icon display, choose File > Preferences, and on the Viewports panel, change Non-Scrolling Object Size.

Positioning Lights









Once you have placed lights in your scene, you can use transforms to change a light's position or orientation.

Transforming Lights

Use transforms on light objects as follows:

Move—  Changes the position of lights. You can also change the position of light targets.

Rotate—  Use Rotate to change the orientation of lights.

You can't rotate a target light about its local X or Y axes. Instead, use Move to move the light or its target. Rotating the light about is local Z axis can be useful if the light uses a rectangular beam or projects a bitmap.

Rotating a plain omni light has no effect as they cast light uniformly in all directions. However, rotating an omni projector light causes the projected image to rotate.

Scale—  Scaling spotlights and directional lights changes the size of their light beam and attenuation ranges. Scaling Omni lights only changes the attenuation ranges.

Light viewports are another convenient way to transform and change parameters of Spotlights and Directional lights. See chapter 2, “Viewing and Navigating 3D Space” for information about using Light viewports.

Tip:  When you adjust lights, it can be useful to turn on Degradation Override. If Override is off and 3D Studio MAX begins to display shaded viewports in wireframe, you can't see the result of the changes you make to lights.

Placing Highlights

You use Place Highlight to position a light to create a specular highlight at a designated point on an object. Place Highlight is one of the buttons on the Align flyout.

Place Highlight moves or rotates the selected light object to aim it at a face on an object you pick. The light maintains its original distance from the face.

Tip: Place Highlight works with any kind of selected object. You can also use Place Highlight with a selection set that contains more than one object. All objects maintain their initial distance from the face.

Setting Light Color and Intensity

Color and intensity are the two properties most people think of when they try to describe a light. You have many options to work with when setting color and intensity for your lights.

Choosing a Color

You change a light's color in the Color area of the General Parameters rollout.

- You can directly edit the light color using the RGB and HSV fields on the rollout.
- You can click the color swatch and edit the light color using the color dialog.

See the Appendix, "Working with Color" for information about color models.

Controlling Intensity

You set light intensity by changing the Value (V) parameter of the HSV color model. Once you have defined a color for a light, you can think of the Hue (H) and Saturation (S) parameters as defining the basic color of the light.

Think of the Value parameter as being like a dimmer switch:

- At a Value of 0, the light is dark, or off.
- At a Value of 255, the light is at maximum intensity.

The Multiplier parameter also increases or decreases a light's intensity. You should leave the Multiplier set to its default of 1.0, except for special effects.

- High positive Multiplier values wash out colors. If you set a spotlight color to red, then increase its Multiplier to 10, the light is white in the hotspot and red in the falloff area.
- Negative multiplier values result in "dark light"—the light darkens objects by subtracting the light color instead of adding light.

Turning Lights On and Off

You can turn a light on and off using two different methods.

- To turn a light on or off for the entire animation, select or clear the On checkbox on the General Parameters rollout.
- To animate a light turning on or off, animate the Value parameter.

Increasing Contrast

You change the Contrast parameter to control the contrast between the diffuse and ambient areas of an illuminated surface. Set Contrast greater than zero to increase contrast for special effects, such as the harsh light of outer space.

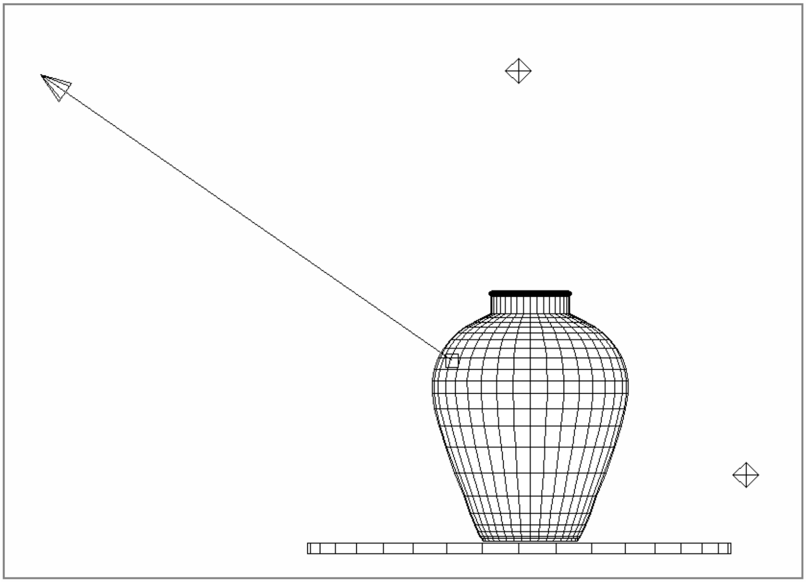
Limiting Light Effect

You can also control whether a light illuminates only the specular areas of a surface, the diffuse areas, or both.

- Affect Diffuse and Affect Specular are normally selected. This means the light illuminates the entire surface of an object.
- Clear Affect Diffuse to prevent the light from illuminating the diffuse surface property.
- Clear Affect Specular to prevent the light from illuminating the specular surface property.

When Affect Diffuse is clear, you can use one light to color the specular highlights of an object, while not coloring its diffuse area.

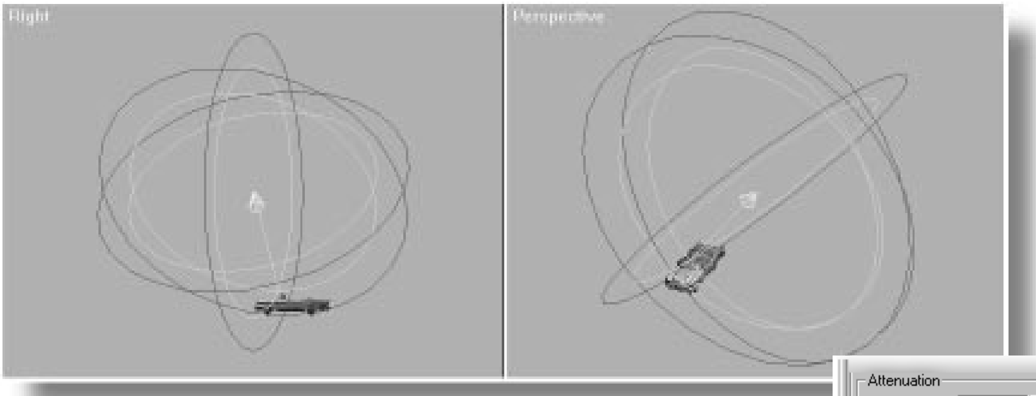
When Affect Specular is clear, you can use a second light to color the diffuse portion of the surface while not creating specular highlights.



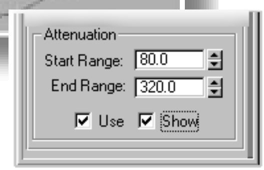
Default V value

Value of V doubled

Setting Attenuation and Decay



Attenuation range display



Attenuation parameters

You use attenuation and decay to restrict how far light travels from a light object and how quickly the light intensity drops off.

Setting Light Attenuation

You can attenuate both the near and far ends of lights. Think of attenuation as a way to force light to fade in and fade out over a specified distance.

Near Attenuation—The Start parameter sets the distance from the light source where light begins to fade in. The End parameter sets the distance where light reaches its full intensity.

Far Attenuation—The Start parameter sets the distance from the light source where light begins to fade out. The End parameter sets the distance where light fades out completely.

Light intensity varies linearly between the start and the end of the range.

Enabling Decay

You enable Decay to reduce the intensity of light over the distance of its throw.

- You can use Far Attenuation to reduce the intensity of light over distance. The calculation is fast but the intensity reduces linearly, which may not always look realistic.
- You use Decay when you want light intensity to reduce at a more realistic, curved rate.

Choose one of three decay options:

None—Disables Decay. Light maintains full strength from source to infinity, assuming that Attenuation is not in use.

Inverse—Applies an inverse decay equal to 1/d, where d = distance. Distance is measured from the light source, or from the End of Near Attenuation if attenuation is used.

Inverse Square—Applies an inverse-squared decay equal to *(1/d)2*. This formula is the “real-world” decay of light, but you might find it too dim in the world of computer graphics. You can compensate by increasing the light Multiplier value or, you can turn up the Global Lighting Level value in the Environment dialog.

The point at which the decay begins depends on whether or not you use Attenuation.

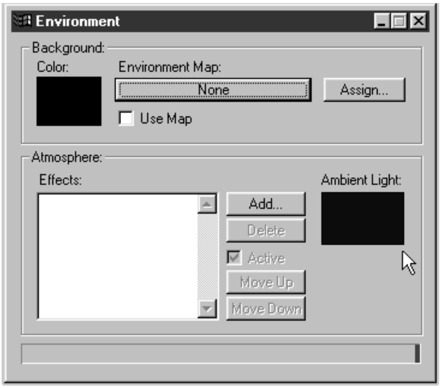
- When Attenuation is not used, decay begins at the source of the light.
- When Attenuation is used, the decay begins at the End position or Near Attenuation.

Once the beginning point is established, the decay follows its formula to infinity, or until the light is cut off by the Far Attenuation settings. Because decay continues to calculate dimmer and dimmer values as the distance of the light throw increases, it's a good idea to use Far Attenuation to eliminate unnecessary calculations.

Note: With no Attenuation or Decay, an object can paradoxically appear to grow brighter as it moves away from the light source. This is because the angle of incidence more closely approaches 90 degrees for more of the object's faces.

Setting Ambient and Global Light Values

You set the ambient light value to simulate background radiosity. Change ambient light in the Environment dialog. You can also change two global light parameters to tint or multiply all lights in the scene.



Environment dialog

Setting Ambient Light

The color of ambient light tints the scene. For most renderings, the color of ambient light should be the complement of the color of the main light source for the scene.

The intensity of ambient light affects contrast as well as overall illumination—the higher the intensity of ambient light, the lower the contrast. This is because ambient light is completely diffuse, so the angle of incidence is equal for all faces.

Tip: Because 3DS MAX doesn't calculate radiosity based on light bounced from objects in the scene, don't rely on ambient light to create the full effect of radiosity. Usually you need to add additional light objects to create a well-lit scene.

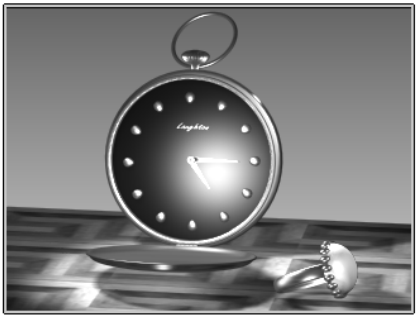
Setting Global Lighting Parameters

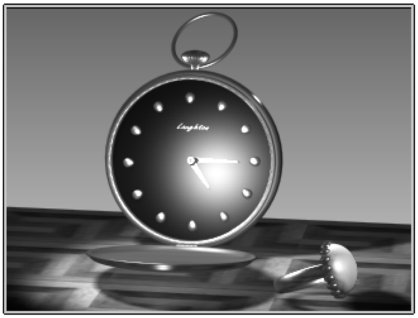
You can globally tint or multiply all lights in your scene using the following global lighting parameters:

Tint—Click the color swatch and change the color from white to add that tint color to the color of all lights. White has no effect on lights.

Level—Change value to multiply the intensity of all lights in the scene, except ambient light. A level of 1 preserves the normal light settings. Higher numbers raise the lighting, and lower numbers reduce the lighting.

Including and Excluding Objects from Lights





You can define a list of which objects to exclude or include from a light. By default, all objects in your scene are included in the illumination from a light.

Objects excluded from a light aren't illuminated by the light. Excluding objects helps you simulate, for example, a shade above an Omni-like light bulb. It is also useful for reducing the light cast on some objects when you render a scene without shadows.

You can also exclude objects from shadow-casting instead of illumination. This can help save rendering time. Choosing Both excludes a light from both illumination and shadow-casting.

The exclusion control is in the General Parameters rollout for all kinds of lights.

Adjusting Light Beams

You create spotlights and directional lights to create beams of aimed light.

- Spotlights create cone-shaped, focused beams of light. Flashlights, track lights, follow spotlights in theaters, and car headlights are common examples of light with cone-shaped focused beams.
- Directional lights create focused parallel beams of light. Laser beams and sunlight streaming through a hole in the clouds are examples of lights with focused parallel beams.

Changing Light Hotspot and Falloff

You set the hotspot and falloff parameters to control how much of the scene is illuminated by a directional light or spotlight. 3DS MAX constrains the falloff to always be greater than the hotspot angle.

- Spotlights measure hotspot and falloff as angles.
- Directional lights measure hotspot and falloff as radii.

Light diminishes at the edge of the light beam. When a light beam falls on a surface, the area of maximum illumination is known as the *hotspot*. The hotspot is usually surrounded by an area of dimmer light known as the *falloff* area.

The greater the difference between the edge of the hotspot and the outer edge of the falloff, the softer the edge of the light beam. If the hotspot and falloff are equal, the edge of the light beam is sharp.

Changing Light Shape

You can choose the shape of the light beam to be circular or rectangular.

- Circular beams work best for spotlights, flashlights, and similar light fixtures.
- Rectangular beams work best for projectors, and special fixtures like ceiling troffers.

Changing Light Aspect Ratio

You set the *Aspect ratio* to describe the shape of a rectangular beam as a ratio of width divided by height. As you change the Aspect Ratio the height of the light beam changes, and the width of the beam remains the same. Aspect ratio is most frequently used to describe image displays such as projections or television screens.

The Bitmap Fit button is another way to set the aspect ratio, but it is for use with projections, and is described in the topic “Projecting Images” later in this chapter.

Enabling Overshoot

Select Overshoot to cause a directional light or spotlight to ignore the falloff area and cast light over the entire scene.

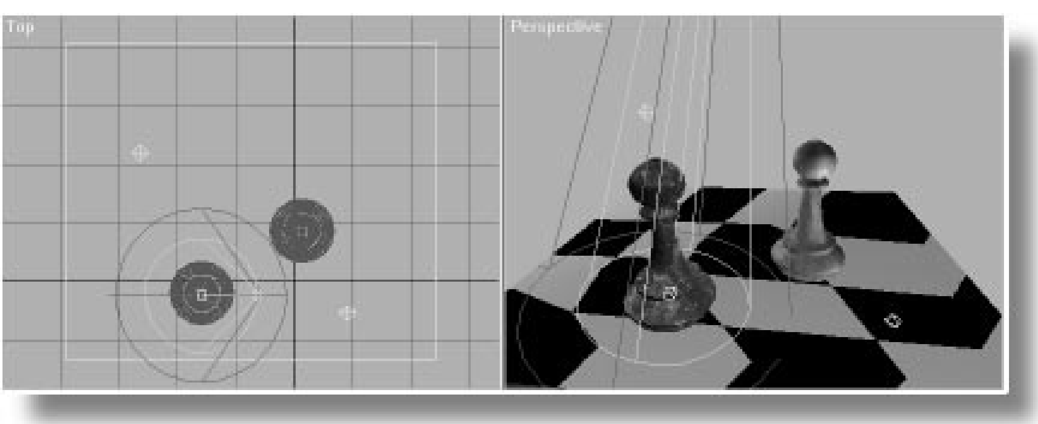
The falloff area is still used by the light to limit the area of cast shadows and projection.

Overshoot is useful when you want to light a large area but need to cast shadows in only a small part of that area. Set the falloff to include the area where shadows must appear, and then turn on Overshoot to light the rest of the scene. This technique can reduce the size of shadow maps and thereby improve rendering speed.

wide screen = 1.85:1

video = 1.333:1

Typical aspect ratios

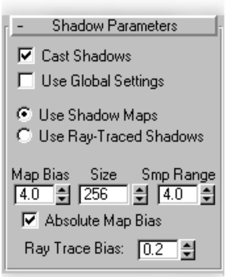


Cone of focused light with hotspot and falloff

Casting Shadows


Scene without shadows


Scene with shadows



Shadows can enhance the realism of a scene and increase the three-dimensionality of its appearance. For some purposes, such as architectural studies, shadows can be indispensable. However, shadows also increase rendering time, sometimes considerably.

All lights can cast shadows. Be careful when using shadow-casting omni lights because they must perform six shadow calculations—one for each orthogonal direction. Spotlights and directional lights only perform one shadow calculation for the direction in which they are aimed.

Note: With Overshoot turned on, spotlights cast light in all directions, like an omni light, but cast shadows only within the falloff cone. Directional lights cast light throughout the scene, in one direction, but cast shadows only within the falloff area.

Global Versus Local Shadow Control

You can choose to use a global setting for shadows, or to control shadows cast by individual lights.

- When Use Global Settings is on, the other shadow controls (except Cast Shadows) are set to the values used by all other shadow-casting lights in the scene that have Use Global Settings selected.
- Changing the affected parameters for one light with Use Global Settings checked, changes them for *all* lights with Use Global Settings selected.

Methods of Generating Shadows

You also choose which method the renderer uses to generate shadows, either for a particular light or for all shadow-casting lights in the scene.

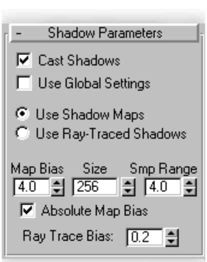
3DS MAX uses two different methods for generating shadows. Each has advantages and disadvantages.

Shadow Maps—A *shadow map* is a bitmap that the renderer generates during a pre-rendering pass of the scene. Shadow maps don't show the color cast by transparent or translucent objects. However, shadow maps can have soft-edged shadows, which ray-traced maps cannot.

Ray-Traced Shadows—The renderer generates *ray-traced* shadows by tracing the path of rays sampled from the light source. Ray-traced shadows are more accurate than shadow maps, and they can show the color cast by transparent and translucent objects. However, ray-traced shadows always have hard edges, which might be inappropriate for scenes with soft lighting.

Controlling Shadow Mapped Shadows

Mapped shadows with object casting shadow on itself



You set multiple parameters to control the generation of shadow maps. You find the following parameters in the Shadow Parameters rollout.

Setting Map Size

Set the map size as the number of pixels for one line of a square map. The total number of pixels in the map is equal to the map size squared. The size you use for a rendering depends upon the following variables:

- Distance of the light from the shadow-casting objects. The farther the light is from the object, the larger the map needs to be.
- Size of the falloff area. The larger the light falloff area, the larger the map needs to be.
- Size of the rendering output resolution. The higher the resolution, the larger the map needs to be.

If shadows appear too coarse—aliased or even checkered— increase the map size. Large map sizes require more memory and time to render. A map size of 4096 uses 64 MB of memory: $4096 \times 4096 \times 4$ four bytes per map pixel.

The default map size of 256 is sufficient for an image in which the light source is not very far from the object and the rendering output resolution is 320×240.

Map Bias

You change the Map Bias to move shadows toward or away from shadow-casting objects.

- Increasing the bias moves the shadow away from the object.
- Decreasing the bias moves the shadow closer to the object.

For example, if a shadow-casting object intersects another object but its shadow leaves a gap at the intersection, the bias is too high.

You can also use Map Bias to avoid problems with objects that cast shadows onto themselves. If you see streaks or moire patterns on the surface of the object, the bias value is too low.

If you increase the bias so much that the shadow becomes disconnected from the object, reduce the bias and increase the shadow map Size value instead.

Sample Range

You change the Sample Range to change the softness of shadow edges. The Sample Range determines how much the shadow is averaged.

- Small values reduce the area that is averaged, bringing the edge of the shadow inward, producing sharper-edged shadows. Sharp-edged shadows can exhibit aliasing. If aliasing occurs, increase the map size.
- Large values extend the area that is averaged, moving the edge of the shadow outward, producing softer-edged shadows. Soft-edged shadows have more antialiasing. The effect is somewhat like the falloff of a soft-edged spotlight.

Rendering time increases *exponentially* as the Sample Range value increases.

Controlling Ray-Traced Shadows

You set a single parameter, Ray Trace Bias, to move ray traced shadows toward or away from shadow-casting objects.

- Increasing the bias moves the shadow away from the object.
- Decreasing the bias moves the shadow closer to the object.

For example, if a shadow-casting object intersects another object but its shadow doesn't meet properly at the intersection, the bias is too high. This effect varies with the angle of the spotlight to the object. Extremely shallow spotlight angles usually require higher bias values.

Another purpose of bias is to avoid problems with objects that cast shadows onto themselves. If you see streaks or moire patterns on the surface of the object, the bias value is too low.

If you increase the bias so much that the shadow becomes disconnected from the object, reduce the bias.

Ray traced shadows on single and multiple objects

Setting Shadow Casting Properties for Objects

Bottom scene shows some objects excluded from casting shadows

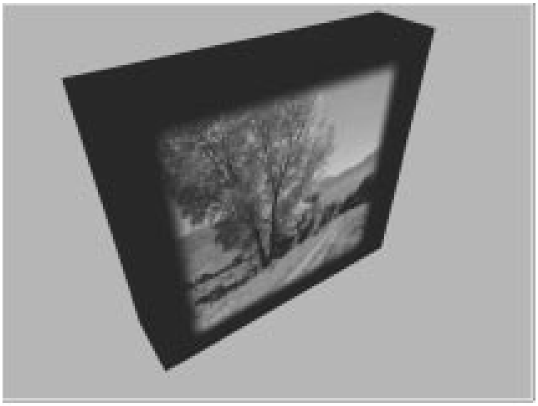
You can set whether objects cast shadows and can have shadows cast onto them. You set an object's Cast Shadows and Receive Shadow properties on the Object Properties dialog.

- Right-click an object and choose Properties to display the Object Properties dialog.

Casting and receiving shadows are two separate settings. An object can cast shadows but not receive them, receive them but not cast them, or neither cast nor receive shadows.

Note: You can also prevent objects from casting shadows by excluding them from a light. See "Excluding Objects from Lights" previously in this chapter.

Projecting Images



Projection in a scene

You can set up lights to project images, much like photographic slides, movies, or the shadow patterns (*gobos)* used in theatrical lighting.

Projections appear when you render a scene; they are not visible in shaded viewports.

The projected image is a *stand-alone map,* you use the Material Editor to preview and adjust the map. For a full description of maps and managing them in the Material Editor, see chapter 21, "Designing Mapped Materials."

If the map you choose to project is animated, the light projects a moving image. Each frame of the projected map corresponds to one frame of the rendered animation.

Note: Lights project maps only within their falloff area, even if Overshoot is turned on. See "Adjusting Light Beams."

Mapping Parameters for Projected Maps

You set mapping parameters for a projected map on the Coordinates rollout of the Material Editor. Follow the guidelines below for predictable results:

- Choose Texture as the coordinate type and choose Explicit UVW 1 as the mapping method.
- Change parameters to offset *UV* coordinates or tile or mirror the map. See chapter 21, "Designing Mapped Materials."
- Correct for aliasing, if necessary, by blurring the map. Blurring a map is also useful for creating an impressionistic effect, or a distant background.

You can set the map to be projected with Environment coordinates, such as Spherical or Shrink Wrap. The map is projected in the same way as it would be mapped to the environment.

Aspect Ratios for Projected Maps

You can project maps at any aspect ratio. The map is stretched as necessary to fill the falloff area of the projector light. This can be a problem when projecting bitmaps. To avoid distortion, you set the aspect ratio to equal the aspect ratio of the bitmap.

Use the Bitmap Fit button on the light Parameters rollout as a convenient way to match the aspect ration of a projector light to the aspect ration of a bitmap.

Animating Lights

You animate lights by using transforms or changing creation parameters on different keyframes while the Animate button is on. 3D Studio MAX interpolates light transforms and parameter values between keyframes, as it does for object geometry.

Note: If you adjust a spotlight viewport while the Animate button is on, you animate the light.

Moving and Rotating Light Objects

The following methods can be used to move and rotate light objects.

- Moving an Omni light can be useful when it's a "practical" light within a scene—that is, a light that appears in the scene itself. You do this by combining the light with a self-illuminating geometric object.
- Changing the parameters of a sunlight system—which uses a directional light—can simulate different times of day.
- Use free spotlights when the spotlight is to move within the scene. Free spotlights are especially intended to be animated along a path, using a path controller. Unlike target spotlights, free spotlights can bank as they travel. Use target spotlights when the light position is fixed.
- If you do need to move a target spotlight, link both the light and its target to a dummy object, then assign the path controller to the dummy object.
- You can use a Look At controller to have a spotlight track a moving object.

  If the spotlight is a target spotlight, its previous target is ignored.

  If the spotlight is a free spotlight, it effectively *becomes* a target spotlight, with the looked-at object the target.

Animating Light Creation Parameters

The following techniques can be used to animate light creation parameters.

- To dim or brighten a light over time, animate its Value parameter.
- To change the color of a light over time, animate its color parameters. Use a smooth tangent for color change keys unless you want the color to change abruptly.
- To make a light flash on and off, set its Value to 0 in repeated keyframes, and assign a step tangent to its color change keys.

Chapter

18

Setting Up Cameras

Cameras present a scene from a particular point of view. Cameras in 3D Studio MAX are objects that simulate still-image, motion picture, or video cameras in the real world.

With a *camera viewport* you can adjust the camera as if you were looking through its lens. Camera viewports can be useful for editing geometry as well as setting up a scene for rendering. Multiple cameras can give different views of the same scene.

If you want to animate the point of view itself—for example, to fly over a landscape or walk through a building—then you must create a camera and animate its position. You can animate other camera parameters as well; for example, you can animate the camera's field of view to give the effect of zooming in on a scene.

Characteristics of Cameras

Real-world cameras use lenses to focus the light reflected by a scene onto a *focal plane* that has a light-sensitive surface.

Focal Length

The distance between the lens and the light-sensitive surface, whether film or video electronics, is called the *focal length* of the lens. Focal length affects how much of the subject appears in the picture.

Effect of focal length on the scene:

- Lower focal lengths include more of the scene in the picture.
- Higher focal lengths include less of the scene but show greater detail of more distant objects.

Measuring Focal Length

Focal length is always measured in millimeters. A 50 mm lens is a common standard for photography.

Definition of lenses based on focal length:

- A lens with a focal length less than 50 mm is called a *short* or *wide-angle* lens.
- A lens with a focal length longer than 50 mm is called a *long* or *telephoto* lens.

Field of View

The *field of view (FOV)* controls how much of the scene is visible. The FOV is measured in degrees of the horizon. It is directly related to the focal length of the lens. For example, a 50 mm lens shows 46 degrees of the horizon.

Focal length, FOV, and perspective are interrelated in the following ways:

Relationship between focal length and FOV:

- The longer the lens, the narrower the FOV.
- The shorter the lens, the wider the FOV.

Relationship to perspective:

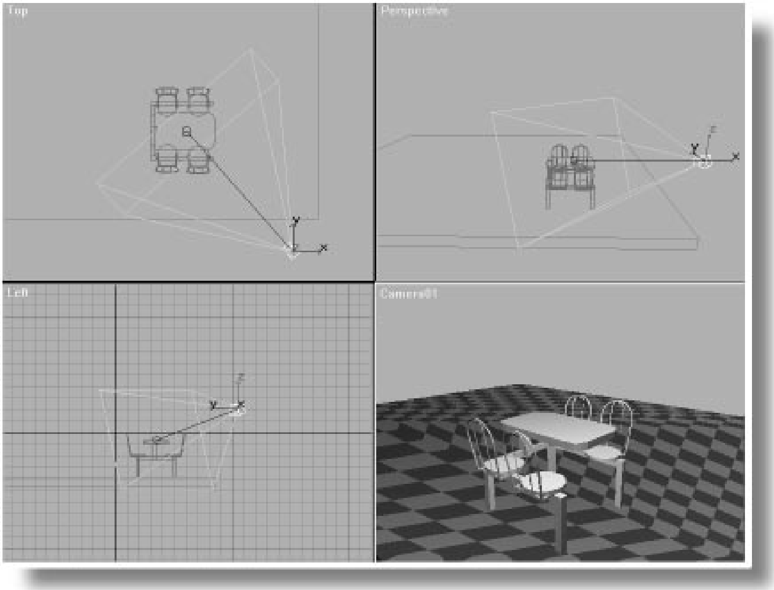
- Short focal lengths (wide FOV) emphasize the distortions of perspective, making objects seem in-depth, looming toward the viewer.
- Long focal lengths (narrow FOV) reduce perspective distortion, making objects appear flattened and parallel to the viewer.

The perspective associated with 50 mm lenses appears normal, partly because it is close to what the eye sees, and partly because such lenses are so widely used for snapshots, news photos, cinema, and so on.

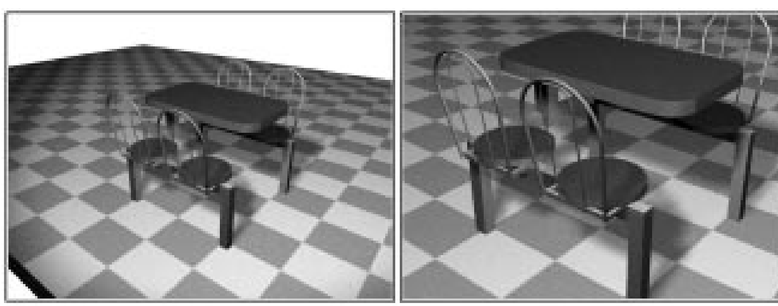
Differences from Real-World Cameras

Many other controls on real-world cameras—for focusing a lens, handling film or videotape, and so on—aren't needed for computer rendering and have no counterpart in the camera objects you use in 3D Studio MAX.

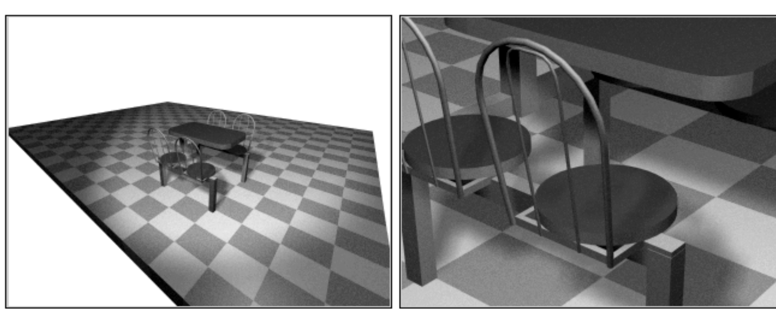
3DS MAX does have counterparts for the camera movements used in motion picture making—truck, dolly, pan, and so on. See "Using Camera Viewports."



Using a camera to frame a scene

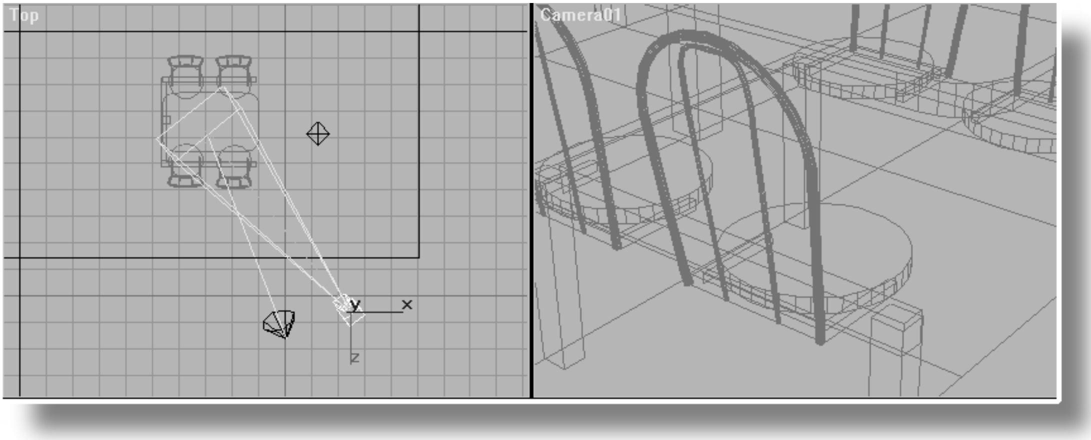


Same object at different focal lengths



Perspective distortions due to different focal lengths

Camera Objects in 3D Studio MAX



Camera and the viewplane it defines

Camera objects in 3D Studio MAX simulate cameras by projecting geometry onto the *viewplane* defined by the camera's position and orientation in the scene. To see this viewplane yourself, you use a camera viewport.

The main parameter for a camera is its FOV, or field of view, which determines how much of the viewplane is visible to the camera. (You can also specify a focal length as an alternate way of expressing the FOV.)

How to Use Cameras for Rendering

To render a scene using cameras in 3DS MAX, you follow these overall steps:

- Create the camera and aim it at the geometry you want to be the subject of your scene.
- Create a camera viewport for that camera.
- Adjust the camera's position and parameters using this viewport. If you do this in Animate mode, you can animate the camera.
- Render the camera viewport.

Kinds of Cameras

There are two kinds of camera objects in 3DS MAX.

Target Cameras—View the area around a target object you create when you create the camera.

Target cameras are easier to use when the camera does not move along a path while rendering the scene or animation.

Free Cameras—View the area in the direction the camera is aimed.

Free cameras are easier to use when the camera's position is animated along a path. They can bank as they travel the path, which Target cameras cannot.

Camera Object Icons

Camera objects are visible in viewports unless you choose not to display them. However, the geometry that appears in the viewport is only an icon meant to show you where the camera is located and how it is oriented.

You cannot shade camera objects. However, you can render their icons using Views > Make Preview. See chapter 30, “Rendering Scenes and Animations,” or online reference.

To change the display size of camera icons:

- In the Viewports panel of the Preference Settings dialog (File > Preferences), set the Non-Scaling Object Size. Default=1.0 in current units.

This also changes the size of light icons, helper objects, and other non-scaling objects in the scene.

Effect of Scale on Camera Icons

The display of camera object icons is not scaled when you change the scale of the viewport. When you zoom in on a camera, for example, the icon size does not change.

Scale transforms have the following effects on a camera object:

- Uniform Scale has no effect other than increasing or decreasing the icon size.
- Non-Uniform Scale and Squash change the size and shape of the camera’s FOV cone. You see the effect in the viewport, but the camera’s parameters do not update. See “Adjusting FOV and Focal Length.”

Hiding Cameras

As with other kinds of objects, you can hide cameras.

To hide individual cameras:

- Use Hide functions on the modeless Display Floater (Tools > Display Floater).

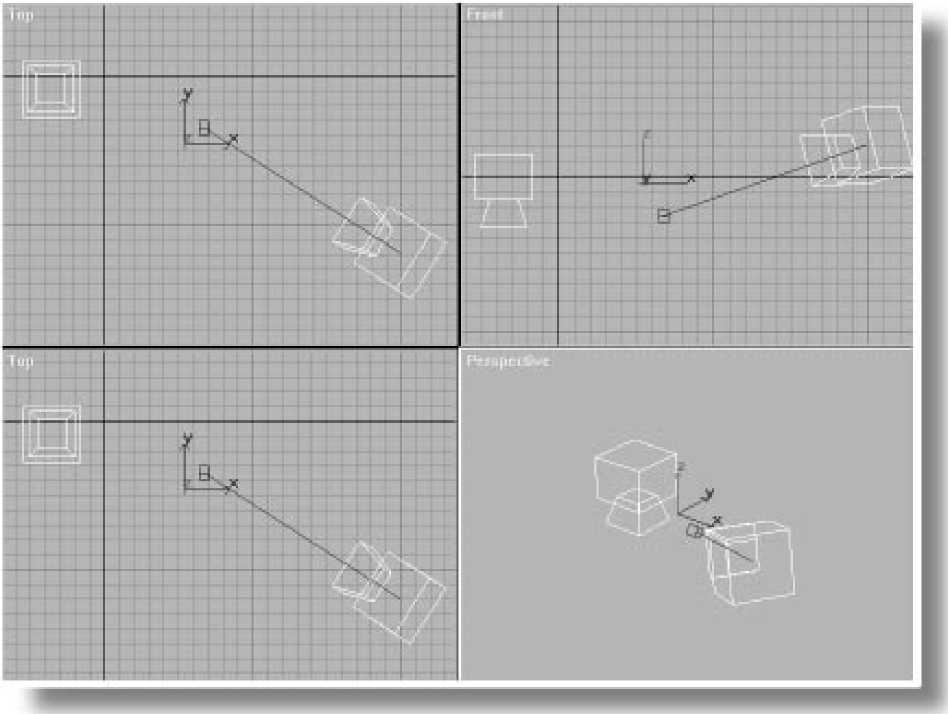
To hide all cameras:

-  On the Display panel, in the Hide by Category rollout, set Cameras to hide all cameras.

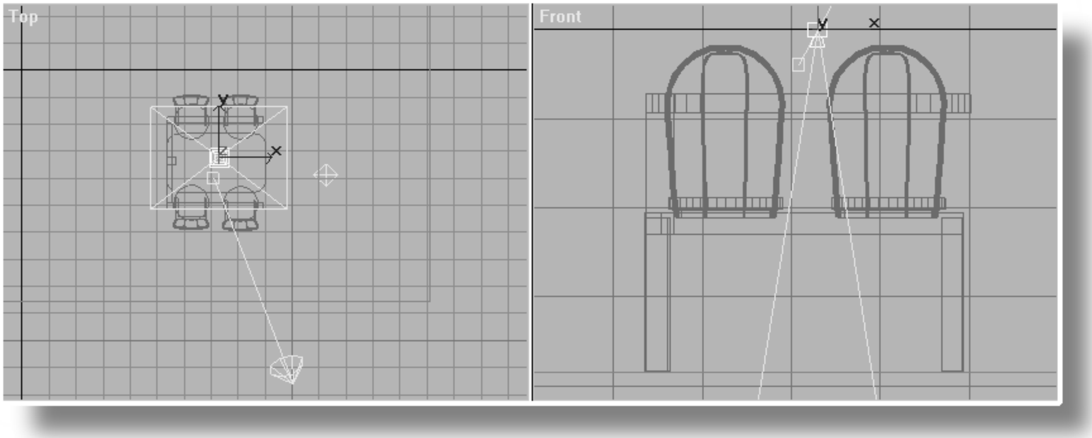
  You can also do this in the Object Level panel of the Display Floater.

When camera icons are displayed, the Zoom Extents commands include them in views. When camera icons are not displayed, the Zoom Extents commands ignore them.

Placing Cameras in a Scene



Icons for camera objects



Initial direction of a Free camera

You place a camera in a scene by creating it. The procedure differs slightly for creating a Target camera versus a Free camera.

By default, both cameras are created on the active construction plane, unless you use 3D snaps during creation. See chapter 5, “Precision and Drawing Aids” for information on snaps.

Creating a Target Camera

The procedure creates the camera first, then its target.

To create a target camera:

1.  On the Create panel, click Cameras. In the Object Type rollout, click Target.

2. Drag in a viewport. The initial point of the drag is the location of the camera, and the point where you release the mouse is the location of the target.

   The Target camera is now part of the scene. It is always aimed at the target, which is a separate 3D Studio MAX object.

3. Set the creation parameters, as described in following topics.

Creating a Free Camera

The initial direction of a Free camera is along the negative Z axis of the active construction grid of the viewport where you create it. In other words, if you click in an orthogonal viewport to create the camera, the initial camera direction is directly away from you. Here are specific examples:

- Clicking in a Top viewport aims the camera at the scene from above.
- Clicking in a Front viewport aims the camera at the scene from the front, and so on.
- Clicking in a Perspective, User, Spotlight, or Camera viewport aims the camera downward—along the negative Z axis of the World coordinate system.

To create a free camera:

1.  On the Create panel, click Cameras. In the Object Type rollout, click Free.

2. Click the viewport location where you want the camera. The kind of viewport you click determines the camera’s initial direction, as explained previously. The Free camera is now part of the scene.

3. Set the creation parameters, as described in following topics.

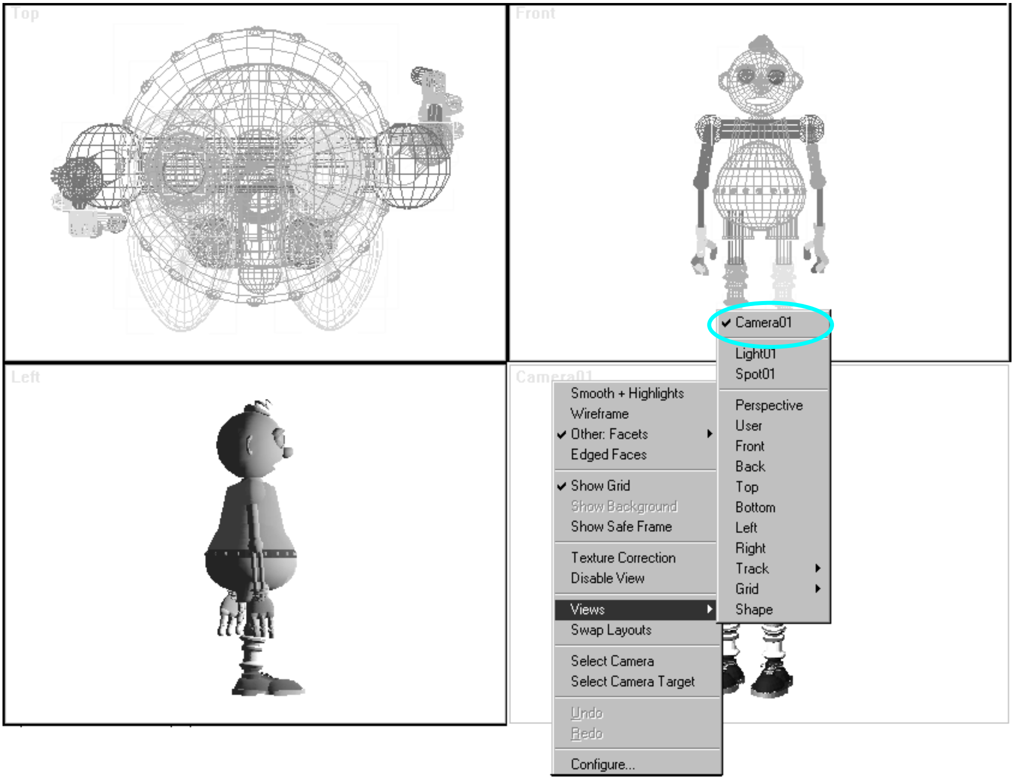
Because the camera is created on the active construction plane, where you also create geometry, you might have to move the camera before you can see objects in its camera viewport.

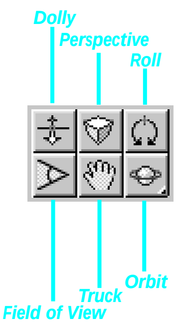
Choosing a Camera for Vertical Views

If you need a camera to look vertically upward or downward, use a Free camera. This is because 3DS MAX constrains a Target camera’s up-vector—its local positive Y axis—to be as close as possible to the world positive Z axis.

This is no problem when you are working with a static camera. However, if you animate the camera and put it in a nearly vertical position, either up or down, 3DS MAX flips the camera view to prevent the up-vector from becoming undefined. This creates sudden changes of view.

Using Camera Viewports





Camera Navigation buttons

You use camera viewports to "look through" the camera as you adjust the camera object or animate it in preparation for rendering. The viewport shows the scene from the camera's point of view. You adjust the scene with the viewport navigation buttons. Changes you make alter the camera—using these buttons is the same as Moving or Rotating the camera, or changing its creation parameters.

Changing a Viewport to Camera View

The procedure is the same for a Target or a Free Camera.

To change to a camera viewport:

1. Right-click the viewport label to display the Viewport Properties menu.

2. Choose Views. The name of each camera in your scene appears at the top of the View list.

3. Choose the name of the camera you want. The viewport now shows that camera's point of view.

Note: Making a camera viewport active does *not* automatically select the camera. To adjust a camera by using its viewport and the Modify panel at the same time, see the next topic.

Displaying Safe Frames

As in other viewports, in camera viewports you can see a display of *safe frame* areas to help you compose a scene for final rendered output.

To display safe frames:

1. Right-click the camera's viewport label.

2. In the View menu, choose Show Safe Frame.

See chapter 30, "Rendering Scenes and Animations." for the use of safe frames.

Camera Navigation Buttons

Using viewports, including camera viewports, is discussed fully in chapter 2, "Viewing and Navigating 3D Space." This topic reviews how to use the navigation buttons that apply to camera views.

 **Dolly—**Moves the camera forward or backward along its own line of sight (the camera's local Z axis). Because Dolly doesn't change the field of view (FOV), dollying changes the size of the area seen by the camera.

Drag up to dolly the camera closer to its target. The subject grows larger in the viewport.

Drag down to dolly the camera away from its target. The subject grows smaller in the viewport.

Dollying is equivalent to moving the camera while constrained to the local Z axis. It is similar to zooming a non-camera viewport.

 **Perspective—**Adjusts the perspective of the scene by dollying the camera *and* changing the field of view (FOV). The overall field of the viewport remains the same. Perspective change is most noticeable in the vertical lines of the scene and the horizontal lines that recede to a vanishing point.

Drag up or down in the viewport to change the perspective.

 **Roll—**Rotates the camera about its own line of sight (the camera's local Z axis).

Drag left or right in the viewport to roll the camera.

Rolling is equivalent to rotating the camera while constrained to the local Z axis.

 **FOV—**Changes the field of view (FOV). The greater the FOV, the more of the scene is visible and the smaller the subject.

Drag up to decrease the FOV. The subject grows larger in the viewport.

Drag down to increase the FOV. The subject grows smaller in the viewport.

 **Truck—**Moves both the camera and its target perpendicular to the line of sight.

Drag to truck the camera and its target in any direction.

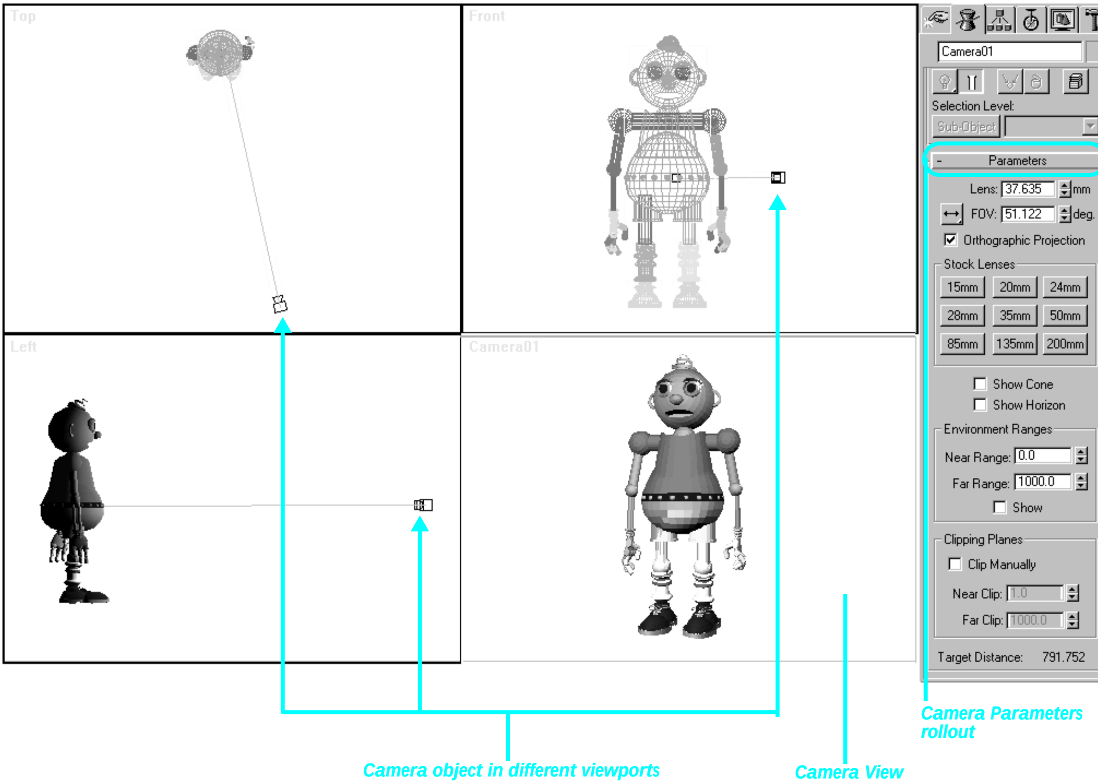
Trucking is equivalent to selecting a Free camera, or a Target camera and its target object, and then moving the camera while constrained to the local X axis. It is similar to panning in a non-camera viewport.

  **Pan and Orbit—**Use the flyout to choose either Pan or Orbit. Pan rotates the target about the camera, as if the camera were mounted on a tripod. Orbit rotates the camera about its target.

Drag in the viewport to pan or orbit the camera.

Orbit is similar to using Arc Rotate in a Perspective or User viewport.

Adjusting Cameras



You can use transforms to aim a camera and change its orientation in the scene.

Move—Adjusts the position of the camera object or the position of a Target camera's target.

Because the target is displayed as a small square, and because it is often in the same area as objects that are the subject of the camera, it can be hard to select by clicking.

To select the target:

- Select the camera object itself, right-click it, and then choose Select Target from the shortcut menu.
- Alternatively, choose Cameras from the Selection Filters list on the toolbar, and then click the target.

Rotate—Adjusts the orientation of the camera object. This transform is most useful with Free Cameras.

You can't rotate a Target camera about its local X and Y axes, because it is constrained to aim at its target. Instead, use Move to reposition the camera or its target.

You should avoid rotating a Target camera to a nearly vertical position, either up or down. Use a Free camera for these orientations.

 **Scale—**Adjusts the scale of the camera object icon. Non-Uniform Scale and Squash change the size and shape of the camera's FOV cone as well. You see the effect in the viewport, but the camera's parameters do not update. See "Adjusting FOV and Focal Length."

In a camera viewport, you can also use the navigation buttons to adjust the camera interactively. See the previous topic. Some navigation buttons, such as Dolly, are actually transforms.

Modifying Camera Parameters

Once you've placed a camera in a scene, you can use the Modify panel in conjunction with camera navigation controls to change the camera's parameters. To do this, you need to both *select* the camera and make its viewport *active*.

Note: Making a camera viewport active does *not* automatically select the camera. This is because you can use camera viewports for working on geometry as well as adjusting cameras.

To adjust a camera by using its viewport and modifiers:

1. Change a viewport to view through the camera you want to adjust. See the previous topic.
2. Select the camera in any convenient viewport, or with Select By Name on the toolbar.
3. Click the *camera viewport's label* to activate the viewport. (This technique prevents you from deselecting the camera.)
4. Open the Modify panel. Now you can adjust the camera using the camera's parameters as well as the navigation buttons.

Adjusting FOV and Focal Length

A camera's field of view (FOV) controls how wide an area the camera views. You can also adjust focal length to alter FOV.

Changing either FOV or focal length changes the other in the opposite direction. A higher FOV has a shorter focal length, and vice versa. Watch the effect in the camera's viewport as you make the changes.

Only the FOV value is saved with the camera. The focal length value is merely an alternative way to express and select the FOV.

FOV Controls on Modify Panel

These are the controls on the Modify panel for a selected camera.

Lens—Sets the camera's focal length, measured in millimeters.

FOV—Sets the arc of the camera's horizon, measured in degrees.

A flyout button next to FOV shows a horizontal arrow, indicating that the FOV angle is measured horizontally. You can change to either vertical or diagonal measurement. The change is only to the method of measurement and has no effect on the view of the camera.

Stock Lenses—Provide preset focal lengths typical of real-world cameras, measured in millimeters.

FOV Controls on Render Scene Dialog

The values shown for FOV and Lens can also be altered by setting Aperture Width in the Output Size area of the Render Scene dialog. This does not change the view through the camera, only the relationship between FOV and Lens values. See chapter 30, "Rendering Scenes and Animations" for use of the Render Scene dialog.

FOV Controls in a Camera Viewport

The following buttons affect FOV. See "Using Camera Viewports" for other details.

 **FOV—**Lets you adjust the field of view interactively in a camera viewport.

 **Perspective—**Changes the FOV in conjunction with dollying the camera.

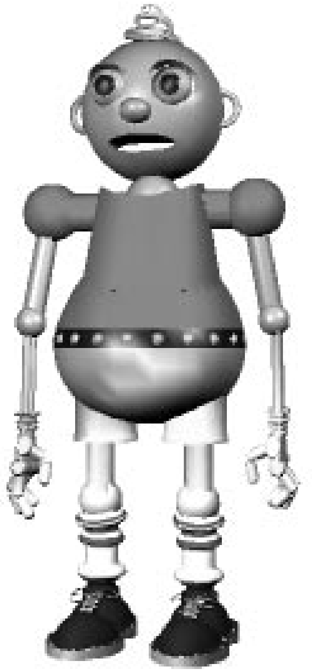
Viewing the FOV Cone

While cameras are visible, you can choose to display the cone—actually a pyramid—defined by a camera's field of view. The cone appears in viewports other than the camera's own viewport. Default=cleared.

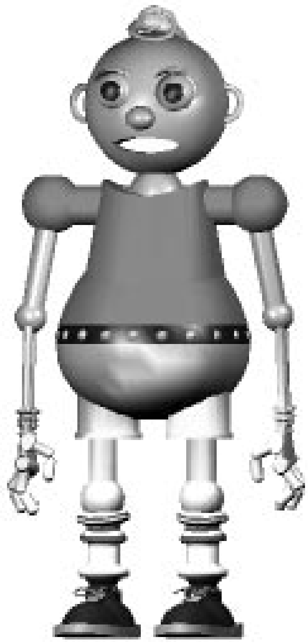
To view the FOV cone:

- Select Show Cone in the Parameters rollout. The camera's field-of-view cone appears outlined in light blue.

Note: A camera's cone is always visible while the camera object is selected, regardless of the Show Cone setting.



Mechanical Boy as seen in Camera View



Mechanical Boy as seen in orthographic projection in Camera View

Switching to Orthographic Projection

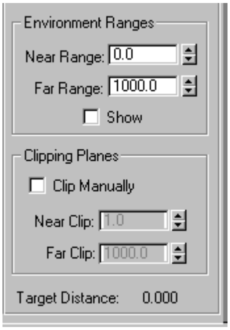
You can toggle to an orthographic projection of a selected camera's active viewport. Default=cleared.

To toggle to orthographic projection:

- Set Orthographic Projection in the Parameters rollout. The Camera view now looks like a User View

All the viewport navigation buttons are operable in orthographic projection. However, when you use the Perspective button to change FOV, the effect is cancelled by the projection—nothing seems to happen. You'll see the change when you clear Orthographic Projection.

Adjusting Clipping Planes and Environment Ranges



Clipping planes and environment ranges help you control effects related to depth in your scene. Clipping planes directly determine a camera's "depth of view," while environment ranges are set for atmospheric effects of depth when rendering.

Adjusting Clipping Planes

Clipping planes let you exclude some of a scene's geometry and view, or render only certain portions of the scene.

Each camera has a *Near* and a *Far* clipping plane. Objects closer than the Near clipping plane or farther than the Far clipping plane are invisible to the camera.

The location of each clipping plane is measured along the camera's line of sight (its local Z axis) in the current units for the scene.

Clipping Plane Parameters

Clipping planes are useful for rendering a selected portion of a scene that has a lot of complex geometry. They can also help you create cutaway views.

Clipping plane settings are part of the camera's creation parameters. By default, Clip Manually is cleared. The camera ignores the Near and Far clipping planes, and their controls are grayed out. The camera renders all geometry within its field of view.

Clip Manually—Activates the Near and Far ranges. As you set these ranges, you see their planes in the camera's cone, represented by red rectangles with crossed diagonals.

Near Clip—Positions the Near clipping plane. Objects closer to the camera than the Near distance are not visible to the camera and aren't rendered.

Far Clip—Positions the Far clipping plane. Objects farther from the camera than the Far distance are not visible to the camera and aren't rendered.

The Near value is constrained to be less than the Far value.

Using Clipping Planes

You can set the Near clipping plane close to the camera so that it doesn't exclude any geometry, and still use the Far plane to exclude objects.

Similarly, you can set the Far clipping plane far enough from the camera that it doesn't exclude any geometry, and still use the Near plane to exclude objects.

If the clipping plane intersects an object, it cuts through that object, creating a cutaway view. How much of the cutaway object is visible depends on whether the object's material is two-sided or not.

Adjusting Environment Ranges

A camera's *environment range* values are used by some atmospheric effects. Standard Fog, for example, can have a certain density at the Near Range and a different density at the Far Range—the fog effect is visible when you render that camera's viewport.

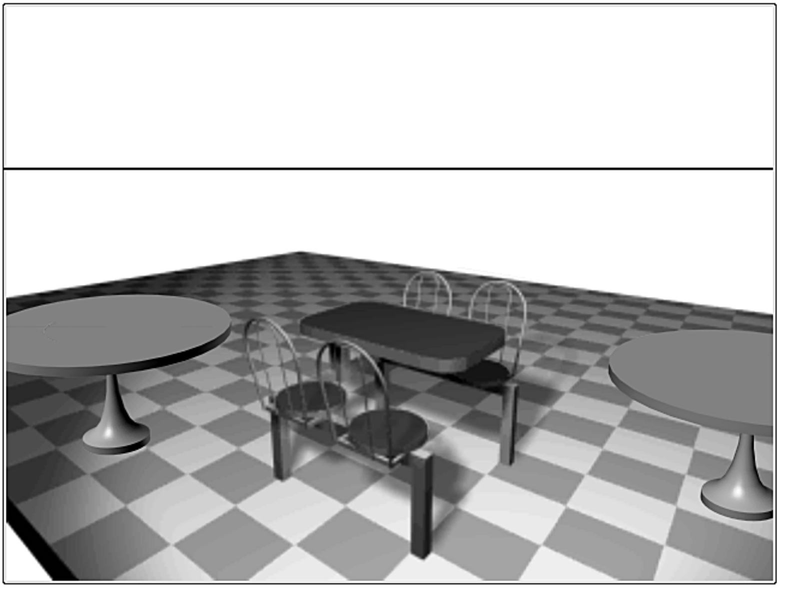
By default, the Near Range=0.0, and the Far Range equals the Far clipping plane value.

To use environment ranges:

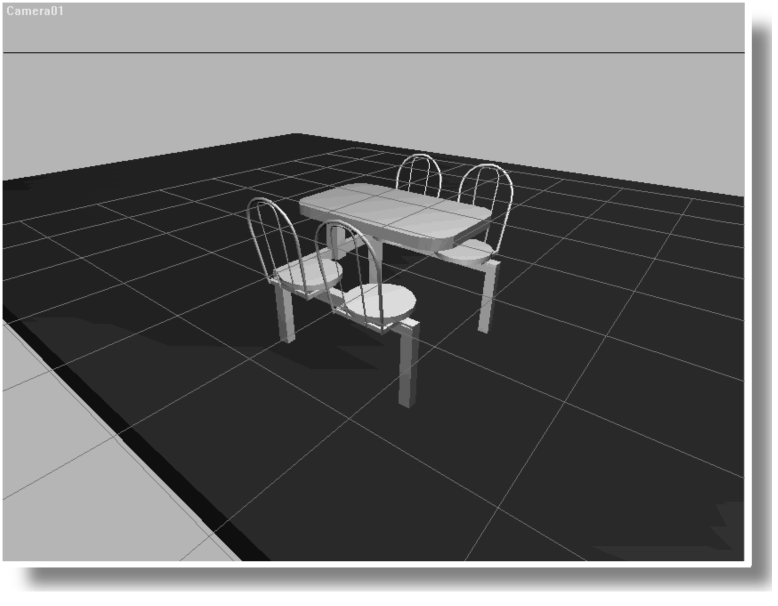
1. Adjust the value of Near Range or Far Range as needed for the atmospheric effect.
2. Select Show to see the environment ranges in viewports. The ranges appear as open, colored rectangles in the camera's cone, easy to distinguish from red clipping planes.

See chapter 29, "Creating Environments and Atmospheres," for a description of atmospheres and how some atmospheres use environment range values.

Viewing the Horizon



Matching perspective



Horizon line displayed in a camera viewport

The horizon of a scene is the edge of vision at the height of the camera, parallel with the world coordinate plane. You can view the horizon in camera viewports.

Horizon Line and Camera Level

A camera is level when it and its target are the same height from the world coordinate plane—in other words, the camera's local Z axis is parallel to the world plane.

When the camera is level, the horizon line is centered in the viewport. As the camera tilts up, the horizon line lowers; as it tilts down, the horizon line raises.

The horizon line control is in the camera's Parameters rollout.

To view the horizon:

- Select Show Horizon. A dark gray line appears at the level of the horizon in the camera's viewport. Default=cleared.

The horizon line might not be visible if the horizon is beyond the camera's field of view or if the camera is tilted very high or low.

Using the Horizon to Match Perspective

The horizon line can help you match the perspective of your scene to the perspective of a still image.

3D Studio MAX supplies a Camera Match utility for aligning a scene with the perspective of a background image. See online reference for details.

In general, use the horizon-line technique when an approximation is all you need. Use Camera Match when you have real-world measurements of the background image and want as precise a match as possible.

To match perspective using the horizon line:

1. Display the horizon line. Use it to help you adjust the camera and target so they are level.
2. Display the image in the camera viewport. Use Views > Background Image, described in chapter 29, "Creating Environments and Atmospheres."

   If the image's horizon and the camera horizon don't match, you have to offset the image, perhaps by using a paint program.
3.  Orbit the camera until the perspective of the scene roughly matches that of the still image.
4.  Adjust the camera's perspective to fine-tune the perspective match.
5.   Move the camera or target to position the scene against the background.

   If you raise or lower the camera, raise or lower the target by an equal amount, in order to keep them level and maintain the horizon.

Animating Cameras

You animate a camera by using transforms or changing its creation parameters in different keyframes while the Animate button is on. 3D Studio MAX interpolates camera transforms and parameter values between keyframes, as it does for object geometry.

This topic summarizes some possibilities and suggests some techniques. See chapter 22, “Animation Concepts and Methods,” for a complete introduction.

The following are general guidelines for choosing cameras to use in animation:

- Use a Free camera when the camera is to move within the scene.
- Use a Target camera when camera position is fixed.

Moving a Camera Along a Path

Having a camera follow a path is a common way to create architectural walkthroughs, roller coaster rides, and so on. The following are specific tips.

- If the camera must bank or tilt close to the vertical (as on a roller coaster), use a Free camera. Assign the path controller directly to the camera object. The camera follows the path, and you can adjust its point of view by adding pans or rotate transforms. This is comparable to filming with a hand-held camera.
- For a Target camera, link both the camera and its target to a dummy object, then assign the path controller to the dummy object. This is comparable to mounting the camera on a tripod on a dolly. It is easier to manage than having, for example, separate paths for the camera and its target.

Following a Moving Object

You can use a Look At controller to have the camera automatically follow a moving object.

The Look At controller makes the object replace the camera’s target. If the camera is a Target camera, its previous target is ignored. If the camera is a Free camera, it effectively *becomes* a Target camera.

While the Look At controller assignment is in effect, the Free Camera cannot rotate around its local X and Y axes, and can’t be aimed vertically because of the up-vector constraint.

As an alternative to using a Look At controller, you can link a Target camera’s target to the object.

Panning

Use a Free camera to animate pans. Animated panning is not supported for target cameras.

Orbiting

Animated orbiting is not supported for either kind of camera. Instead, create an orbit-shaped path and use a Path controller.

Zooming

Zooming moves toward or away from the camera’s subject matter by changing the focal length of the lens. It differs from dollying, which physically moves the camera but leaves the focal length unchanged.

You can zoom by animating the value of the camera’s FOV parameter.

Creating Animated Cutaway Views

You can animate the creation of a cutaway view by animating the location of the Near or Far clipping plane, or both.

Chapter

19

Adjusting Normals and Smoothing

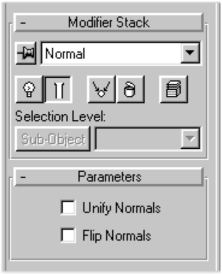
In general, you use the controls described in this chapter to prepare objects for rendering.

A *face normal* is a unit vector that defines which way a face is pointing. The direction that the normal points represents the front, or outer surface of the face. You can manually flip or unify face normals to fix surface errors caused by modeling operations or by importing meshes from other programs.

Smoothing groups define whether a surface is rendered with sharp edges or smooth surfaces. Smoothing groups are numbers assigned to the faces of an object. Each face can carry any number of smoothing groups up to the maximum of 32. If two faces share an edge and share the same smoothing group, they will render as a smooth surface. If they don't share the same smoothing group, the edge between them will render as a corner. You can manually change or animate smoothing group assignment.

See the Online Reference for details about Normals and Smoothing Groups.

Viewing and Changing Normals



Normal modifier parameters

3D Studio MAX automatically generates normals for objects you create. Usually objects render correctly using these default normals. Sometimes, however, you need to adjust the normals.

Undesired normals can appear in these objects:

- Meshes imported from other applications.
- Geometry generated by complex operations such as boolean objects, lathe objects, or lofts.

Normals are used to define which side of a face is considered the “out” side. The out side of a face is the side that gets rendered unless you are using two-sided materials.

Do one of the following to view or change normals:

- Apply a Normal modifier. If a Face sub-object selection is active, Normal applies to the selected faces. If no faces are selected, Normal applies to the entire object.
- Apply an Edit Mesh modifier, enable face sub-object mode, and then use the features on the Edit Surface rollout.
- Convert an object to an Editable Mesh, enable face sub-object mode, and then use the features on the Edit Surface rollout

See chapter 12, “Editing Meshes” for information about Editable Meshes and applying the Edit Mesh modifier.

Viewing Normals

The easiest way to view normals is to look at an object in a shaded viewport. In this case, you are not viewing the normals but rather their effect on the shaded surface. If the object looks as if it is inside-out, or has holes, then some of the normals may be pointing in the wrong direction.

You can display the normal vectors for selected faces by enabling Show Normals on the Edit Surface rollout of an Editable Mesh object or the Edit Mesh modifier.

Unifying Normals

Select Unify Normals to make normals point in a consistent direction. If an object has normals that are inconsistent—some point outward and others inward, the object will appear to have holes in its surface.

Unify Normals is found on the Edit Surface rollout and on the Normal modifier.

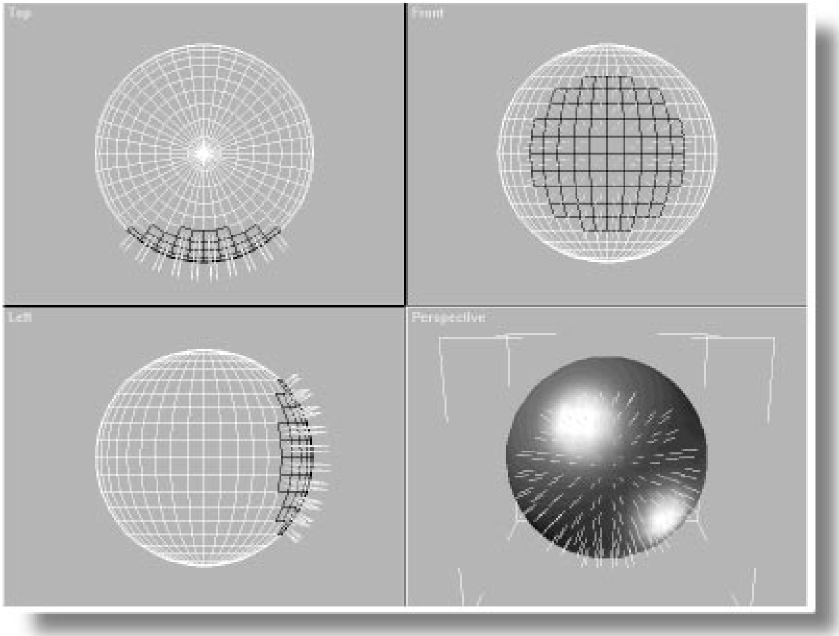
If you are animating the creation of a complex object such as a nested boolean or a loft, and you think the operation might result in inconsistent faces, apply Normal to the result and select Unify.

Flipping Normals

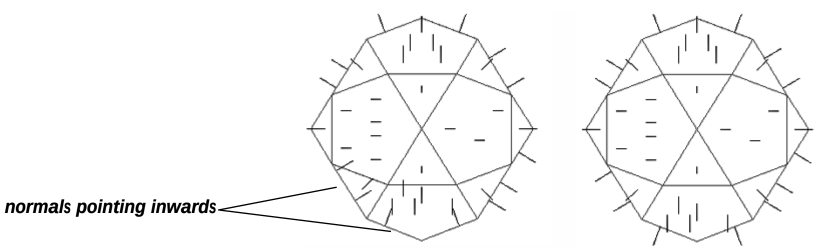
Select Flip Normals to reverse the direction of all selected faces. Flipping the normals of an object turns it inside-out.

Flip Normals is found on the Edit Surface rollout and on the Normal modifier.

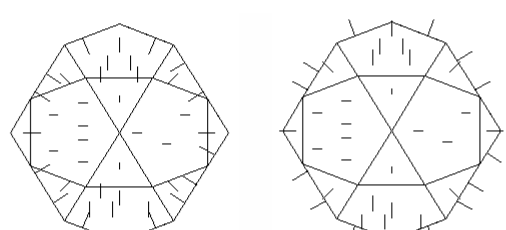
The Lathe modifier sometimes creates an object with normals pointing inward. Use the Normal modifier with both Unify and Flip checked to fix “inside out” lathe objects.



Display of object's face normals



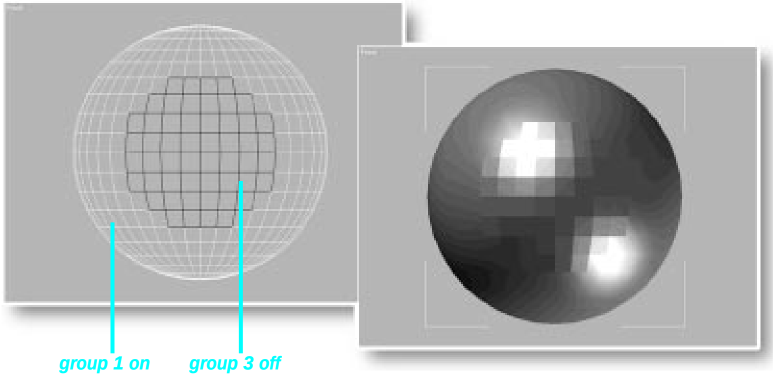
Unifying normals



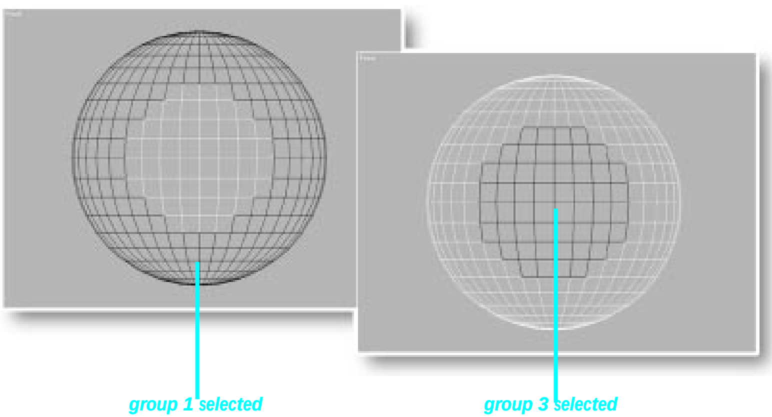
Flipped normals

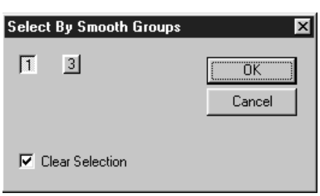
Viewing and Changing Smoothing

Smoothing Group properties in an Editable Mesh



Smoothing group assignments





Select by Smooth Groups dialog

Smoothing is a rendering property that blends edges between faces to produce a smooth curved surface. You can control how smoothing is applied to a surface so your objects can have both smooth surfaces and sharp faceted edges.

Smoothing is controlled by *smoothing groups*, which are integers from 1 to 32. Whether an edges display between adjacent faces depends on whether the faces share a common smoothing group.

- If faces have *no* smoothing groups in common, the faces have an edge between them.
- If faces have at least *one* smoothing group in common, the edge between the faces is smoothed.

Because each face has three edges, only three smoothing groups can be in effect for any face. Extra smoothing groups assigned to a face are ignored.

Do one of the following to view or change smoothing group assignments:

- Select or clear the Smooth option on the Parameters rollout of a parametric object to set default smoothing for the object.
- Apply a Smooth modifier. If a Face sub-object selection is active, Smooth applies to the selected faces. If no faces are selected, Smooth applies to the entire object.
- Apply an Edit Mesh modifier, enable face sub-object mode, then use the features on the Edit Surface rollout.
- Convert an object to an Editable Mesh, enable face sub-object mode, then use the features on the Edit Surface rollout.

See chapter 12, "Editing Meshes" for information about Editable Meshes and applying the Edit Mesh modifier.

Viewing Smoothing Groups

The easiest way to view smoothing is to look at an object in a shaded viewport. In this case, you are not viewing the smoothing groups but rather their effect on the shaded surface.

You can see the smoothing group numbers for selected faces by looking at the Smoothing Group buttons on the Edit Surface rollout of an Editable Mesh object or the Edit Mesh modifier.

Smoothing Group buttons appear as follows:

- Group numbers not used by any face in the selection, appear normal.
- Group numbers used by all faces in the selection, appear selected.
- Group numbers used by some, but not all, faces in the selection, appear blank.

Automatically Smoothing an Object

Click Auto Smooth to assign smoothing automatically. You set a Threshold angle to determine whether to smooth adjacent faces.

- If the angle between face normals is less than or equal to the threshold, the faces are assigned to a common smoothing group.
- If the angle between face normals is greater than the threshold, the faces are assigned to separate groups.

Auto Smooth is found on the Edit Surface rollout and on the Smooth modifier.

Manually Applying Smoothing Groups

You manually assign smoothing groups to a selection of faces by clicking Smoothing Group buttons on the Edit Surface rollout or the Smooth modifier. The smoothing group of each button you click is assigned to the selection.

Selecting Faces by Smoothing Group

You can also select faces according to the assigned smoothing groups. Click Select by Smooth Group on the Edit Surface rollout and then click the smoothing group of the faces you want to select.

This is a convenient way to examine smoothing groups on an object someone else created.

Chapter

20

Designing Basic Materials

A *material* describes how an object reflects or transmits light. Material properties work hand in hand with light properties—shading or rendering combine the two to simulate how the object would look in a real-world setting.

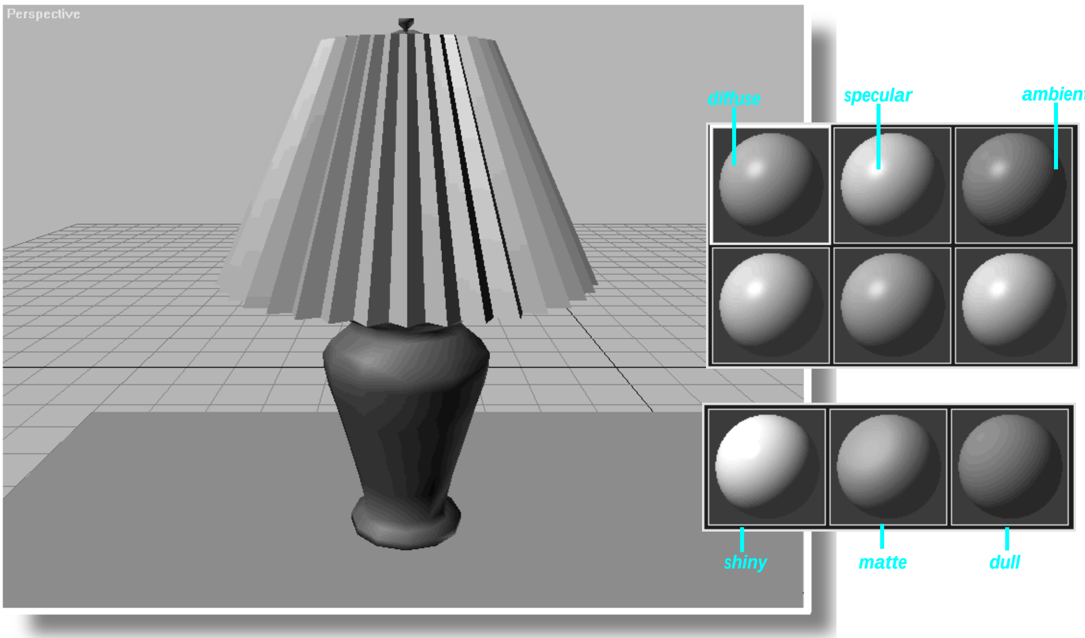
You assign materials to individual objects or selection sets; a single scene can contain many different materials.

You create materials using a dialog called the Material Editor. You can save them in one or more *material library* files, and load saved materials from a library.

Basic materials are for objects of a single uniform color. This chapter describes how to create basic materials. It also describes how to manage materials in a scene and how to use material libraries. Chapter 21, “Designing Mapped and Compound Materials,” describes how to create and use *mapped* materials, *compound materials*, and how to combine maps in a *map tree*.

See the Online Reference for details about all material parameters.

What Are Basic Materials?



Basic materials give an object a single uniform color. In 3D Studio MAX, basic materials are distinguished from *mapped* and *compound* materials, described chapter 21, “Designing Mapped and Compound Materials.”

The way a real-world object’s surfaces reflect light determines how the object appears under different lighting conditions. In 3D modeling, materials simulate surfaces’ reflective properties.

Color Components

A surface of a “single” color usually reflects many colors. Our perception combines these colors and also compensates for lighting conditions to arrive at a description of a single color as “brown” or “green.”

Basic materials model surface colors by using three color components:

Ambient color—The color of the object where it is in shadow. This color is what the object reflects when illuminated by ambient light rather than direct light.

Diffuse color—What the object reflects when illuminated by “good lighting”—that is, by direct daylight or artificial light that makes the object easy to see.

Specular color—The color of shiny highlights.
When we describe an object’s color in conversation, we usually mean its diffuse color. The choice of an ambient color depends on the kind of lighting. For moderate indoor lighting, specular color can be a darker shade of the diffuse color, but for bright indoor lighting and for daylight, it should be the complement of the primary (key) light source. The specular color should be either the same color as the key light source, or a high-value low-saturation version of the diffuse color.

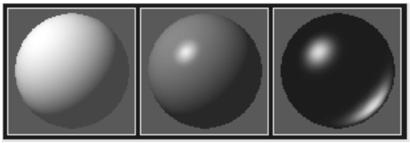
See “Choosing Colors for Realism”for a discussion of choices for ambient, diffuse, and specular colors.

The specular color appears in highlights; the size and shape of highlights are controllable. For a polished surface, a highlight is small and strong. The highlight on a *matte* surface is large and weak; some matte surfaces have no highlight at all.

The three color components and the highlight controls describe all basic reflective materials. (Some additional controls describe transparent and self-illuminating materials.) This method of representing materials is a fairly simple one, but can yield effective renderings.

Transparent and partially transparent objects have a transmissive or *filter* color as well as reflective color. Objects that glow, such as light bulbs, generate their own light and are simulated by *self-illuminating* materials. Basic materials can simulate transparent and self-illuminating objects as well as the more common reflective objects.

Light and Basic Shading



Diffuse, specular, and glancing reflections

Materials work in combination with lights. The intensity of light that falls on a surface determines the intensity of color to display. As described in chapter 17, “Lighting Your Scene,” three factors contribute to the intensity of light where it falls on an object:

Light Intensity—Light’s intensity at its origin.

Angle of Incidence—The more a surface inclines away from the light source, the less light it receives and the darker it appears. The angle between a ray of light and a surface is the angle of incidence for that face.

Distance—Light diminishes over distance. The effect is known as *attenuation*. By default, attenuation is turned off; you can turn it on and set it to affect distances that you choose.

When the angle of incidence is 90 degrees, the face is shaded at full intensity unless the light is attenuated. Full intensity is the light’s Multiplier value times the value of the face’s surface color. The Multiplier value is 1.0 by default; the surface value is the Value component of the surface color’s HSV description. As the angle of incidence diverges from 90 degrees, the intensity of the face decreases.

Lights and the Component Colors of a Material

As the names of a basic material’s three color components imply, the kind of light that strikes a surface with a material determines how the surface appears when it is shaded. Ambient color appears where the surface is lit by ambient light alone—that is, where the surface is in shadow. Diffuse color appears where light falls directly on the surface. It is called “diffuse” because light striking it is reflected in various directions. Highlights, on the other hand, are reflections of light sources.

Specular highlights appear where the *angle of reflection* (typically 90 degrees) is equal to the angle of incidence. Glancing highlights appear where the angle of incidence is low, relative to the observer or camera.

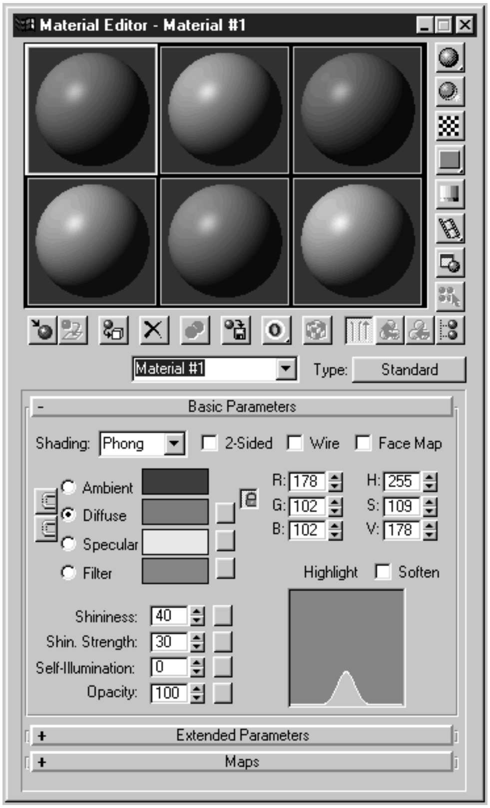
All shiny surfaces have specular highlights; glancing highlights are characteristic of metallic surfaces.

Some surfaces reflect their environment as well as the light sources that illuminate them. You can’t model such surfaces with basic materials. See chapter 21, “Designing Mapped Materials,” for mapping reflections other than highlights.

The three color components blend at the edges of their regions. Between ambient and diffuse, the blending is calculated according to a standard shading model; between diffuse and specular, you can control the amount of blending using controls in the Material Editor.

Opening the Material Editor

You preview, create, and change materials on the Material Editor. Open the Material Editor by clicking Material Editor on the toolbar, or by choosing it from the Tools menu.



Material Editor

Choosing a Sample Slot

The Material Editor has sample slots for viewing previews of materials. When you first view the Material Editor, the material previews have default colors.

To select a sample slot:

- Click the sample slot to make it active. The active sample slot is displayed with a white border around it.

You can right-click a selected sample slot to display a shortcut menu of display properties:

- Choose standard arrangements of 6, 15, and 24 sample material slots.
- Magnify the selected slot, placing it in a separate floating window.

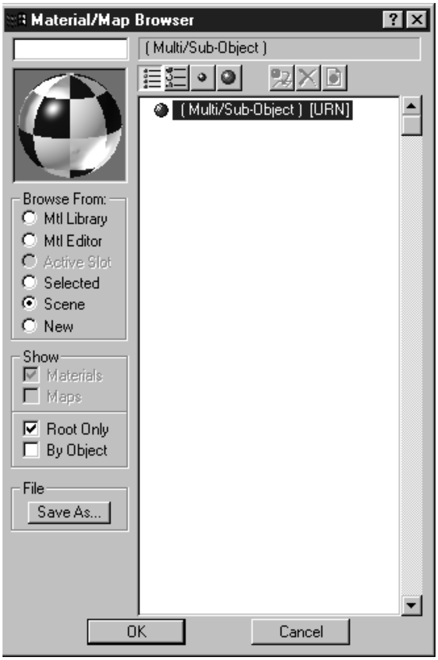
Choosing a Sample Object

You can also choose what type of object you want to see in the sample slot. Choose from three generic objects of sphere, cylinder, and box, or choose a *.max* file to use the first object in the file. See the Online Reference for details about choosing a sample type.

Choosing Sample Display Properties

You can choose many other options to control how the Material Editor displays the lighting, background, colors, and animation of materials in the sample slots. You select these choices using the buttons along the upper right edge of the Material Editor.

Managing Materials in the Material Editor



Material/Map Browser

You create, preview, and change materials in the Material Editor. The scene is where you assign materials to objects. You can also save collections of materials as libraries on disk.

Materials in the Material Editor are saved when you save a *.max* file.

The Material Editor displays from 6 to 24 material samples in its sample slots. These materials don't necessarily appear in a scene or library, but you use the Material Editor to move a material to and from a scene or to and from a library. The Material Editor is the general "clearing-house" interface for managing materials.

When you create a new scene, the materials stored in the Material Editor have generic names such as Material #1 or Material #6, according to their corresponding sample slot. These are basic materials with identical default settings except for the color components, which vary.

Creating a New Material

You create a new material by making a sample slot active, then doing one of the following:

- Edit the material in the sample slot by changing its material parameters.
- Get a new material and set its parameters.

If you change a material, use it in a scene, or save it to a library, give the material a unique name in order to avoid confusion with the default names.

To give a material a different name:

- Edit the name field that appears below the Material Editor toolbar.

The name of the active material appears in the title bar of the Material Editor dialog.

Getting Materials

You get a new material, or existing materials from other sources, by clicking Get Material on the Material Editor toolbar.

Clicking Get Material displays the Material/Map Browser, which is the primary tool for managing materials. You can also open the Material/Map Browser from the Tools menu on the menu bar.

You can get materials in the following ways using the Material/Map Browser:

- Get new materials. Select New on the Browse From area and choose one of the material or map types.
- Get a material from selected objects. Select Selected on the Browse From area and choose from a list of materials assigned to currently selected objects.

- Get a material from the scene. Select New on the Browse From area and choose from a list of materials assigned to any objects in your scene.
- Get a material from a material library. Select Mtl Library on the Browse From area and choose from a list of materials in the open material library.

When you choose a material in the Material/Map Browser, a sample sphere renders a preview of the selected material. You then place the material in a sample slot of the Material Editor by doing one of the following:

- Double-click the selected material to place it in the active slot.
- Drag the selected material to any slot in the Material Editor. You can drag as many materials to as many different Material Editor slots as you wish.

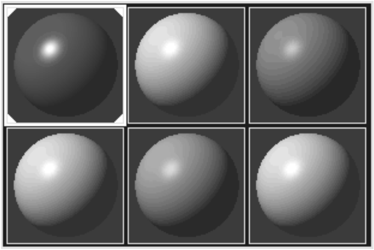
Picking Materials from Objects

You can also pick a material from any object using the eye-dropper tool next to the Material Name field.

To pick a material from an object:

- Click the eye-dropper, and then click an object in the scene to put the material under the eye-dropper into the active sample slot of the Material Editor.

Managing Materials in the Scene



Corner brackets on a hot sample slot

A scene initially has no materials. The colors used for shading, when no materials have been assigned, are for use within 3D Studio MAX; they appear in renderings only when you haven't assigned materials to the scene's objects.

Assigning Materials to Objects

To use materials in a scene, you assign them to objects in the scene. You assign materials to selected objects using the following techniques:

- Click a sample slot on the Material Editor, then click Assign Material To Selection on the Material Editor toolbar. All selected objects are assigned the active material.
- Drag a sample slot from the Material Editor over any object or selection of objects.
- Drag a material from the Material/Map Browser over any object or selection of objects.

Recognizing Hot Materials

When you assign a material to an object or selection, that material becomes a *hot material*. When you change the material's properties, 3DS MAX immediately updates the scene to reflect those changes. Any object with that material will change its appearance—not just the objects in the current selection.

- When a material is hot, its sample slot appears with outlined white corner brackets.
- When a material is hot and it is assigned to the selected object, its sample slot appears with solid white corner brackets.

Copying Materials

You can make a copy of a hot material to experiment with the material parameters without affecting the material in the scene. Copying materials is also a good strategy for trying variations of a material.

To copy materials:

- Drag a material to another sample slot. The original material remains in its sample slot and a copy using the same name is placed in the other sample slot.
- If the active material is hot, you can click Make Material Copy on the Material Editor toolbar. The hot material in the sample slot is replaced with a copy using the same name.

Updating Materials

When you change the parameters of a copied material, you can update the material in the scene using the same name by clicking Put Material to Scene.

The Put Material to Scene button is enabled *only* when:

- The material in the active sample slot has the same name as a material in a scene.
- The material in the active sample slot is not hot.

This command is meant to fit into the following overall sequence of handling materials:

1. You create a hot material either by assigning it to objects in the scene or by getting it from the scene.
2. You make a copy of the material.

3. You make changes to the copy of the material.

4. You update the scene by putting the changed material back into the scene.

These steps are not as immediate as changing a material while it is hot, but they help you avoid changing the scene's materials unintentionally or in unexpected ways.

Removing Materials from Objects

You completely remove materials from objects by assigning the NONE material from the Material/Map Browser.

To remove materials from selected objects:

- On the Material/Map Browser select New on the Browse From area, then drag NONE from the material list to the selected objects.

Selecting Objects by Material

When a material in the Material Editor is assigned to objects in the scene, you can select the objects from the Material Editor.

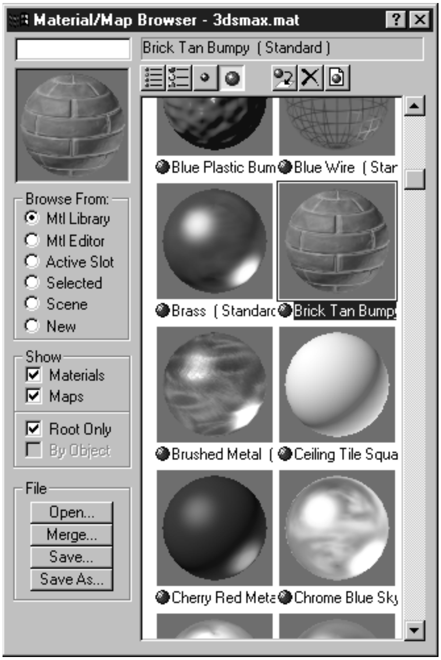
To select objects assigned to the active material:

1. Click a sample slot containing a hot material.

2.  Click the Select by Material button in the Material Editor to display the Select Entities dialog. The names of objects assigned the active material are highlighted for selection.

3. Click Select to select materials assigned the active material.

Managing Materials in Material Libraries



Library files are useful for collecting related materials and for sharing materials between multiple *.max* files. Material library files have a *.mat* file name extension.

You access material library files by selecting Mtl Library on the Material/Map Browser.

Opening and Merging Material Libraries

You can open material libraries and merge libraries by clicking Open or Merge on the File area of the Material/Map Browser.

Material libraries are just a collection of material names and their parameters in the Material Editor. This is the same method used for storing materials used in your scene files. When you open or merge a material library you can select either a standard *.mat* library or a *.max* scene.

Adding Materials to a Library

You can save materials in the Material Editor to a library in the Material/Map Browser. If no library is open, 3D Studio MAX starts creation of a new, unamed library.

To add a material to a library:

-  Click Put To Library on the Material Editor toolbar.
- Drag a material sample slot to the Material/Map Browser.

Deleting Materials from a Library

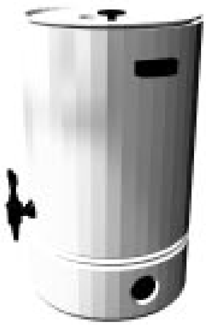
You can delete materials from a library, either individually or all at once. The controls are above the material list.

-  Click Delete from Library to delete a single material or map.
-  Click Clear Material Library to delete *all* the materials and maps in the library.

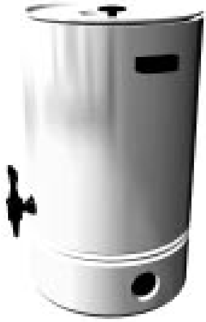
Saving Libraries

Additions you make to a library, by clicking Put to Library or dragging materials into the Material/Map Browser, are automatically saved to the material library. Clearing the library and deleting materials are temporary until you save the library file. Click Save or Save As on the File area of the Material/Map Browser to save the library to a file.

Choosing a Shading Type



constant



phong/blinn

metal

You can choose a shading type for each material by choosing from the Shading list on the Basic Parameters rollout. A *shading type* specifies the shading algorithm used to render a material.

Constant—Renders each face of a surface with a single color intensity based on the face normal. Does not smooth between face edges. Specular highlights are blocky and unrealistic.

Phong—Interpolates intensities across a face based on the averaged face normals of adjacent faces. Calculates the normal for every pixel of the face. Smooths the edges between faces and renders highlights realistically.

Blinn—A variation on Phong shading that produces rounder specular highlights and can handle glancing highlights better than Phong.

Metal—Simulates polished metal. Calculates its own specular color and chooses the location of specular and glancing highlights to model the appearance of metallic materials.

Note: For the Phong, Blinn, and Metal materials, you can turn off smoothing between faces, either for the object as a whole, or selectively by using smoothing groups. See chapter 19, "Adjusting Normals and Smoothing."

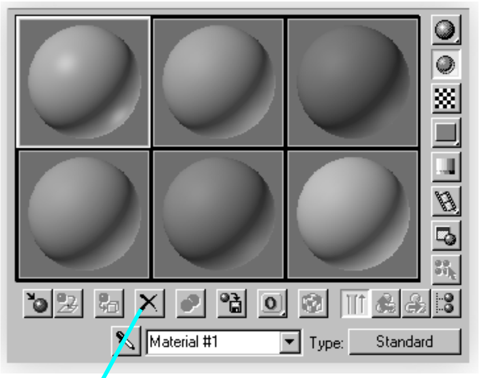
Changing Component Colors

When you adjust basic material attributes, including component colors, you change the preview in the active sample slot. (Changes to other basic material attributes such as shininess or opacity are described in topics that follow.)

The basic material controls are on the Basic Parameters rollout. Make sure this rollout is visible before you proceed.

Before you change colors, you can click Reset Map/Mtl To Default Settings to clear all basic parameters and start with a default gray-shaded material.

Note: You can also make a color component active by clicking the radio button next to its name. This makes the color active but does not display the Color Selector. You can then adjust the color by using the RGB and HLS spinners.



Reset Map/Mtl To Default Settings button

Choosing Colors for Realism

Materials add greater realism to a scene only if you choose their colors and other properties to appear like real-world objects. This topic presents some general guidelines. When possible, you should also observe colors in the objects you are modeling, especially under different lighting conditions.

For objects on which you want the viewer to focus attention, basic materials don't often provide the level of realistic detail you probably want. However, for distant and peripherally visible objects, as well as for some kinds of real-world materials—for example, molded plastic—basic materials do enable you to create realistic renderings.

Indoor and Outdoor Lighting

Whether a scene is indoors or outdoors affects your choice of material colors, just as it affects the way you set up lights. Full sunlight is bright and unidirectional. Most indoor lighting is less intense and more even (that is, multidirectional) than daylight. However, some special indoor lighting (and nighttime outdoor lighting), as for the stage, also features intense, directional light.

Direct sunlight has a yellow tint. Materials for objects to appear under daylight should have a specular color of a light, desaturated yellow; for example, RGB values of 240, 240, 188. The ambient color should be the complement of the specular: a deep, dark purple with a hint of the diffuse color.

Materials for objects to appear under normal interior lighting should have a specular color that is close to white. (Our perception compensates for the yellow or greenish tint that is often present in artificial light.) The ambient color can often have the same hue as the diffuse color, but with a darker value.

Materials for objects to appear under spotlights should follow the general guidelines for daylight materials. The specular color should match the spotlight's color, and the ambient color should be a very dark value of the spotlight color's complementary hue mixed with a bit of the material's diffuse color.

If you want to render an object under changing lighting conditions, you can choose colors that are a compromise between the optimal colors for each kind of lighting, or you can animate the material so that its colors change to suit the changing light.

Designing Natural and Manufactured Materials

Sample of natural and manufactured object

Your color choices for Ambient, Diffuse, and Specular color components determines whether your materials appear natural or manufactured.

Designing Natural Materials

Most natural materials have a matte surface with little or no specular color.

For natural materials such as these, use the following guidelines.

Ambient color—The ambient color depends on whether the scene is indoors or outdoors, as described in "Indoor and Outdoor Lighting."

Diffuse color—Choose a color found in nature. It is best to use the observed color of the object itself, or a similar object.

Specular color —Make the specular color the same hue as the diffuse, but with a higher value and a lower saturation.

Shininess—Set the shininess and shininess strength to low values, as described in "Setting the Shininess of Materials."

Some foliage, bird feathers, fish scales, and so on, are shiny. For materials such as these, set the shininess and shininess strength to higher values. You might also want to change the specular color so it's closer to the lighting color than the surface's diffuse color.

Water is reflective, and best modeled by a color component in combination with a reflection map. See chapter 21, "Designing Mapped Materials."

Designing Manufactured Materials

Manufactured materials often have a synthetic color rather than an "earth tone." Also, many manufactured materials, such as plastics and porcelain glazes, are very shiny.

For manufactured materials, use the following guidelines.

Ambient color—The ambient color depends on whether the scene is indoors or outdoors.

Diffuse color—Although the diffuse color doesn't have to be an "earth tone," as with natural materials you should use the observed color of the object itself, or a similar object.

Specular color—Make the specular color close to white, or to the color of the light source. White is especially characteristic of plastic materials.

Shininess—Set the shininess and shininess strength to high values.

Designing Metallic Materials

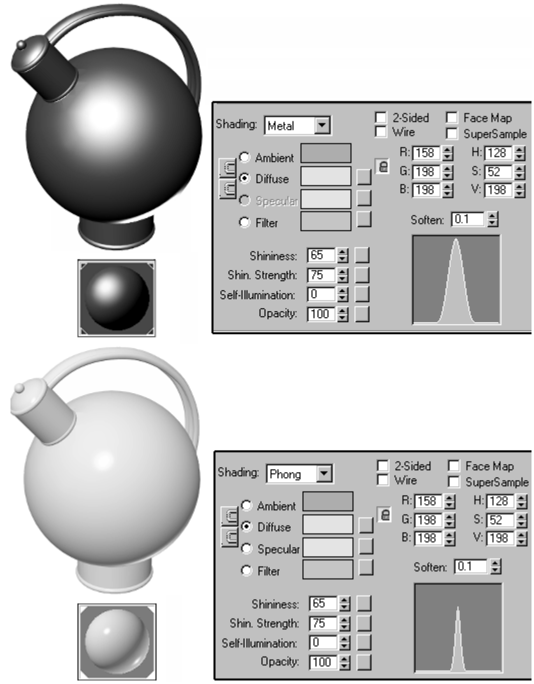
Metallic materials have a characteristic “glancing” highlight that appears where the light is at a low angle of incidence with respect to the observer or camera. To generate this effect, Metal shading uses the Cook/Torrance illumination model.

For metallic materials, use the Metal shading type. When you choose Metal from the Shading list, the Specular color and Soften parameters are disabled.

The Metal shader calculates its own specular color, which can vary between the diffuse color and the color of the light.

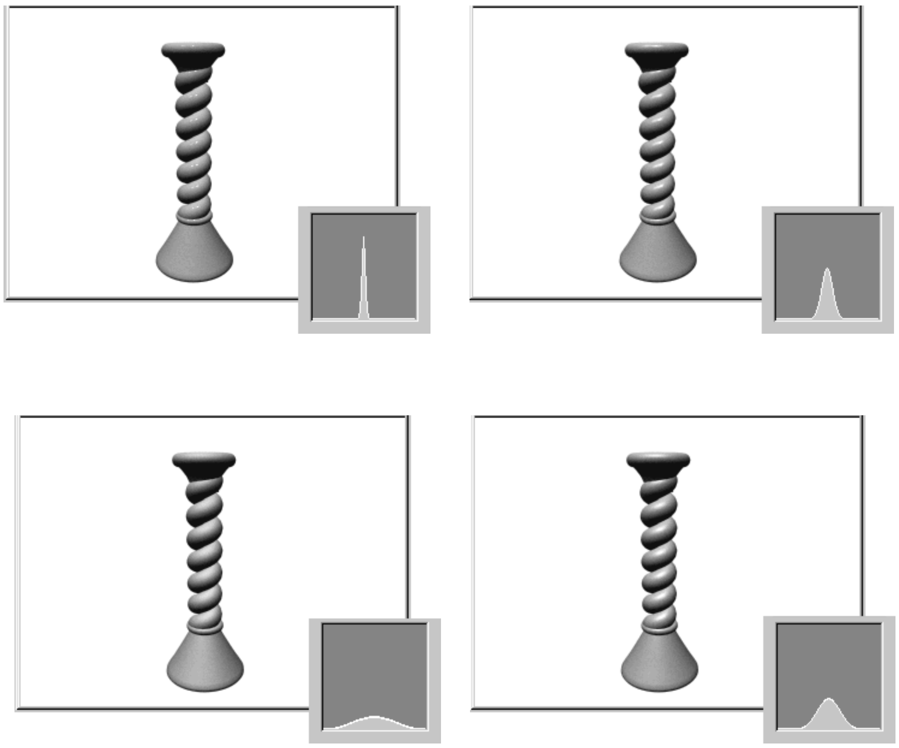
In the diffuse region of a metal material, the ambient component is greater than it is for other kinds of materials.

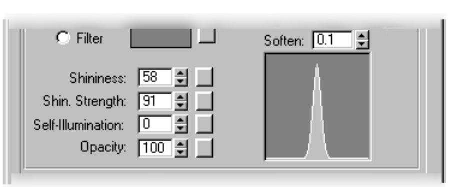
 When you preview metal materials, it is useful to turn on a backlight. This displays metal’s characteristic glancing highlight. To turn it on, click the Backlight button to the right of the sample slots.

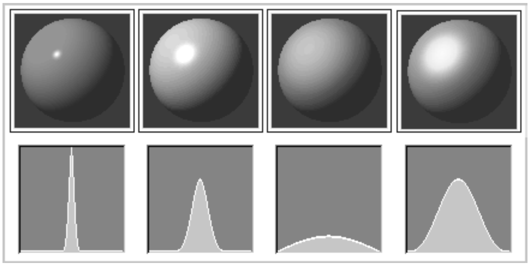


Highlight curves for metallic materials

Setting the Shininess of Materials







You set the following Material Editor parameters to affect highlights and shininess:

- Specular color
- Shininess
- Shininess Strength
- Soften highlight

You adjust the specular color as you do other basic material color components, as described in “Changing Component Colors.” The Shininess and Shininess Strength parameters affect the Highlight curve.

To change the size of a highlight:

- Change the Shininess value. The width of the Highlight curve and the highlight in the preview change. At 0% shininess, the curve is at its maximum width. At 100% shininess, the curve is extremely narrow.

To change the intensity of a highlight:

- Change the Shininess Strength value. The intensity of the Highlight curve and the highlight in the preview change. At 0% strength, there is no highlight. At 100% strength, the curve is at its maximum height.

The shape of the Highlight curve affects the blending between the specular and diffuse color regions of the material. The steeper the curve, the less blending there is and the sharper the edge of the specular highlight.

Setting the Soften Highlight Value

When a material has a high Shininess Strength but low Shininess value, Phong shading can create unrealistically hard edges between the specular and diffuse regions.

To soften these hard edges, increase the value of the Soften parameter.

Designing Shiny Materials

Shiny materials have bright, distinct highlights.

- Specular color is the color of the light it reflects.
- Shininess has a high value. Highlights are small in proportion to the surface area of the material.
- Shininess Strength has a high value. The highlights are bright.

Setting Shininess to medium-high and Shininess Strength to medium-low gives the appearance of a shiny material that shows wear.

Matte Materials

These have the opposite qualities from shiny materials.

- Specular color is the same as or very close to the diffuse color.
- Shininess has a low value.
- Shininess Strength has a low value.

Glossy Materials

Glossy materials, such as finished leather, reflect a lot of light without having strong localized highlights.

- Shininess has a medium low value.
- Shininess Strength is medium high.

Highlight Positioning

The position of lights in a scene affects where the highlights appear. Specular highlights appear where the observer or camera is 90 degrees from the light source. For metal, glancing highlights appear where the object is backlit and the angle of incidence is close to the angle of observation.

If highlights are an important element of your composition, you need to adjust light positions accordingly. Probably you will have to experiment to get a good effect. The Adjust Highlight command, described in chapter 17, “Lighting Your Scene,” can help you position lights to get good highlights.

Reflections

Adding a reflection map, even to just a slight degree, can increase the realism of shiny surfaces. See chapter 21, “Designing Mapped Materials.”

Creating Transparent Materials

Transparent materials such as glass both reflect light and transmit it. The transmitted light might be tinted by the material's *filter color,* which is not necessarily the same as the material's other color components.

Setting the Opacity Value

In the Material Editor, transparency is controlled primarily by the Opacity setting—you create transparent materials by reducing the opacity of a material.

To reduce a material's opacity:

- Change Opacity to a value less than 100%. The material becomes more transparent. An object that is fully transparent (0% Opacity) is nearly invisible except for the light it reflects (the specular highlights).

 To help you preview transparency in the sample slots, it is useful to view the sample object against a background. To do so, click the checkered Background button to the right of the Sample slot.

Setting the 2-Sided Option

With a highly transparent 3D object, you can see the geometry behind the surfaces toward the observer. If you check 2-Sided, 3D Studio MAX renders the faces that would be hidden if the object were opaque, including specular highlights on the rear faces.

2-Sided rendering takes more time than rendering just the forward faces.

Setting the Opacity Type

Opacity Type controls how colors are affected behind the transparent object— how shading affects transmitted light. There are three opacity types on the Extended Parameters rollout:

- *Filter* multiplies the color behind a transparent surface by the filter color.
- *Subtractive* subtracts the filter color from the color behind the transparent surface.
- *Additive* adds the diffuse color to the color behind the transparent surface, ignoring the filter color. Additive opacity tends to make the object appear to glow. You can use additive opacity with self-illumination to enhance the glow effect. See "Using Self-Illuminating Materials."

Using Filter Color

Filter color is the color of light that passes through a transparent object, as in a stained-glass window or a green-tinted soda bottle.

By default, the filter color is 50% gray (Hue 0, Saturation 0, Value 128). At this default value, the filter color is invisible; you must give it a hue or change its value to see the filter effect.

Filter Type and Opacity Percentage

When a material has 100% opacity, the Filter type shows only the diffuse color. When it is at 0% opacity, it shows no color except specular highlights. At 50% opacity, the filter shows 50% of the filter color and 50% of the background. Between these percentages, 3D Studio MAX interpolates the amount of filter and diffuse color to use.

Opacity Falloff

Opacity Falloff is another way to increase the effect of three-dimensionality. When falloff is set greater than zero, the amount of transparency is adjusted according to the angle of a face's normal.

If a face is perpendicular to the viewer, the angle of its normal is 0 degrees; if the face is edge-on to the viewer, its normal is 90 degrees. Opacity falloff uses these two values as extremes.

You adjust Opacity Falloff controls on the Extended Parameters rollout. Falloff is either inward or outward:

In—Transparency increases as the normal angle approaches 0 degrees and decreases where the angle approaches 90 degrees.

Out—Transparency decreases as the normal angle approaches 0 degrees and increases where the angle approaches 90 degrees.

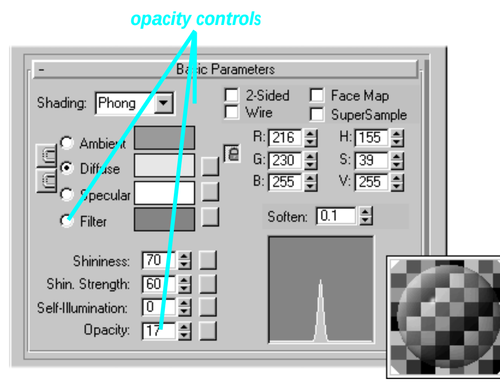
Inward falloff simulates a material that is denser at the edges, such as a hollow glass ball; outward falloff simulates a material that is denser at the center, such as a cloudy marble.

The Amount spinner controls how much falloff to apply. When this value is 0, there is no falloff.

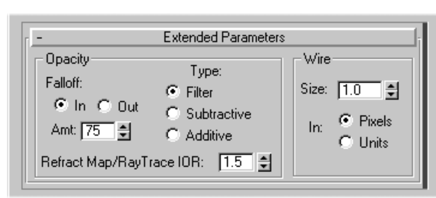
Reflections

Many transparent materials are also shiny. Adding a reflection map, even to just a slight degree, can increase the realism of shiny transparent surfaces. See chapter 21, "Designing Mapped Materials."

Inward and outward falloff

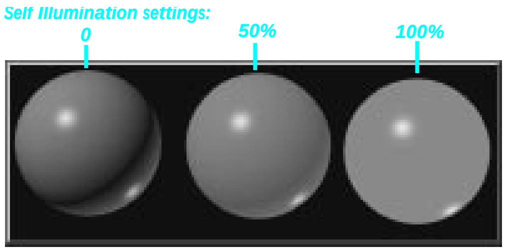


Basic Parameters that affect transparent materials



Extended Parameters that affect transparent materials

Using Self-Illuminated Materials



Self-illumination materials model luminous surfaces

To model objects that emit light—for example, the headlights of a car—use a self-illuminating material.

To make a material self-illuminating:

- Increase the Self-Illumination value in the Basic Parameters rollout.

  As self illumination increases, the sample object appears flatter and more luminous.

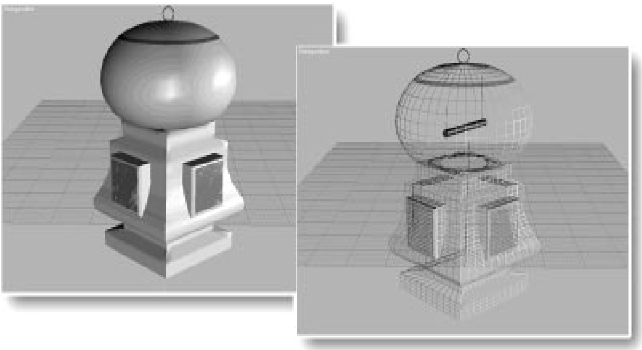
Self illumination replaces a material's ambient color with its diffuse color. At 100% self illumination, no ambient color is visible. Shading shows only the diffuse color and specular highlights (if any).

Self-illuminated materials do not show shadows cast onto them, and they are unaffected by the lights in the scene. The brightness (Value in the HSV color description) remains the same regardless of the scene's lighting.

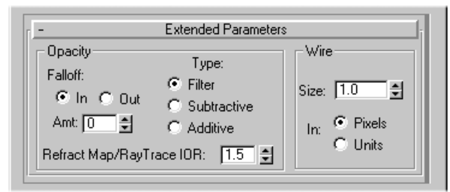
To make a visible light source in a scene, you can combine a geometric object with a light object, and give the geometric object a self-illuminating surface. For example, you could create a lofted light bulb shape, assign it a self-illuminating white or yellowish material, and place an Omni light in the same location.

To make a material self-illuminating and transparent as well, use the Additive Opacity Type with self illumination. See "Setting the Opacity Type."

Wireframe Materials



Object rendered shaded and wireframe



Extended wireframe parameter controls

Create a wireframe material to render objects as meshes. They show the edges of a surface's faces rather than the faces themselves.

To turn on wire mode:

- Check Wire in the Basic Parameters rollout.

  The material is now shaded as a wireframe mesh. The wire portions of the geometry have the same material settings as previously—color components, shininess, and so on.

Tip: Check the 2-Sided option for a wireframe material.

You have two choices for how wireframe materials are rendered. The controls for tuning wireframe shading are on the Extended Parameters rollout.

- If you choose Pixels, the thickness of the wires maintains the same apparent thickness regardless of the scale of the geometry or how near or far the object is positioned. In other words, pixel wires have a constant display size as if the wires were traced over an image.
- If you choose Units, the wires behave as if they were modeled in the geometry. They appear thinner at a distance and thicker at close range. Scaling a wireframe object does scale wire width.

For both Pixels and Units, the Size spinner controls width. If Pixels is active, Size specifies how many pixels wide the wires are. If Units is active, Size is a measurement in the current world units. If the unit size is large relative to the size of the object, the wireframe appears solid or nearly so.

Note: Wireframe objects can cast shadows, but only ray-traced shadows can cast unit thicknesses accurately. Shadow-mapped shadows cast pixel-width wires, even if Units is active. Shadow-casting methods are described in chapter 17, "Lighting Your Scene."

Chapter

21

Designing Mapped and Compound Materials

Maps are patterns that a material applies to a surface. With maps you can simulate textures, applied designs, reflections, refractions, and other effects.

Mapped materials create greater subtlety and realism than basic materials. They add detail to objects without adding complexity to the object geometry.

The simplest kind of map is a bitmap. There are other kinds of maps than bitmaps. Marble, for example, is a procedural texture generated by a program. One way to use maps in combination is to assign multiple maps at the same level in one material. Another way is to use *compound map* types, that combine materials in a hierarchy, or *map tree*.

You can also create a *stand-alone* map tree, independent of any material, to use as a component of other materials or as an environment map. Environments are described in chapter 29, "Creating Atmospheres and Environments."

Compound materials combine two or more materials in a hierarchy. You use compound materials to assign multiple materials to an object, possibly based on the object's geometry.

See the Online Reference for details about the various maps available in 3D Studio MAX.

Uses of Maps

3DS MAX supports several different kinds of surface mapping. Some kinds of mapping use the colors of the bitmap as diffuse maps do. Others, such as opacity maps, use only pixel intensity—the value of the pixel color's HSV description.

Material Maps

The following kinds of maps directly correspond to a material's basic parameters:

- *Ambient maps* apply to the ambient color component of a material; usually the ambient map is locked to use the diffuse map. Use ambient maps to get more realistic surface shading. For example, use a darker, or tinted, version of the diffuse map.
- *Diffuse maps* apply to the diffuse color component of a material. A diffuse map specifies surface color, as if the map image were painted onto the surface.
- *Specular maps* apply to the specular color component, altering the color of the surface's highlight.
- *Shininess maps* apply to the Shininess parameter, altering the pattern of the surface's highlight based on the intensity (the value) of pixels in the map.
- *Shininess Strength maps* apply to the Shininess Strength parameter, altering the intensity of the surface's highlight based on the intensity (the value) of pixels in the map.
- *Self-illumination maps* apply to the Self-Illumination parameter. Based on the Self-Illumination value, lighter (higher-value) areas of the map become self-illuminating, while darker areas don't change the surface.
- *Opacity maps* supplant the Opacity parameter. Lighter (higher-value) areas of the map are opaque, while darker areas become transparent.
- *Filter color* maps apply to the filter color component, altering the filter color of a transparent material based on the map's color.

Three kinds of maps don't directly correspond to a material's basic parameters:

- *Bump maps* create an embossed effect that makes a smooth surface appear to have three-dimensional variations in height. Lighter (whiter) areas of the map appear to be raised, while darker (blacker) areas of the map appear closer to the original surface geometry.
- *Reflection maps* apply the map so it appears to be reflected off the surface.
- *Refraction maps* apply the map so it appears to be refracted through the surface or object.

You can apply any number of maps, in any combination, to a single material.

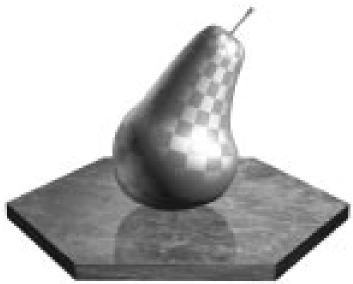
Stand-Alone Maps

You can also set up maps that are not part of any material. These maps are called stand-alone maps and they are used for various environment and modification effects. Some examples of the uses for stand-alone maps include:

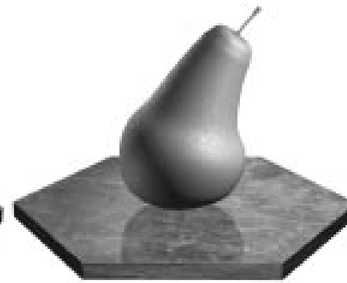
- *Displacement maps* used with the Displace modifier.
- *Projection maps* used with lights.
- *Environment* maps used for background rendering.

diffuse map

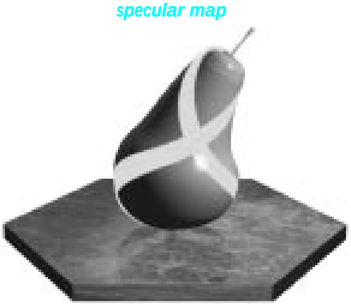
specular map



shininess map

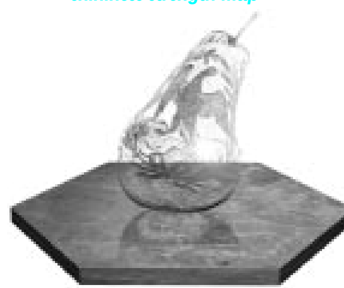


shininess strength map



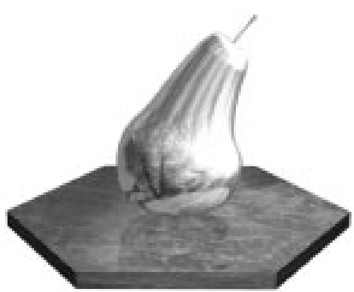
self-illumination map

opacity map



filter color map

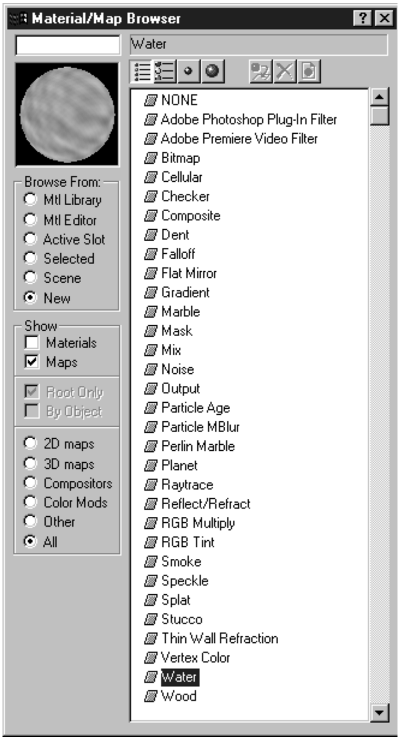
bump map



reflection map

refraction map

Map Types



Material/Map Browser dialog showing all map types

Chapter 20 showed how to use the Material/Map Browser to select materials and manage libraries. The Material/Map Browser is also where you select map types.

Different map types are grouped into categories. You can choose whether the Browser lists maps, materials, or both, and which map type categories are displayed.

Different types of maps create different effects, behave in particular ways, or are provided as ways to combine multiple materials.

- *2D maps* are two-dimensional images that are typically mapped onto the surface of geometric objects, or used as environment maps to create a background for the scene. The simplest 2D maps are bitmaps; other kinds of 2D maps are generated procedurally.

  2D maps require an object to have mapping coordinates applied.

- *3D maps* are patterns generated procedurally in three dimensions. Marble, for example, has a grain that goes through the assigned geometry. If you cut away part of an object with marble assigned as its texture, the grain in the cutaway portion matches the grain on the object's exterior. It is all generated by the same program.

  3D maps do not require mapping coordinates.

- *Compositors* are meant specifically for compositing other colors or maps.

- *Color Modifiers* alter the color of pixels in a material.

- *Other* map types are used for special effects such as reflection and refraction.

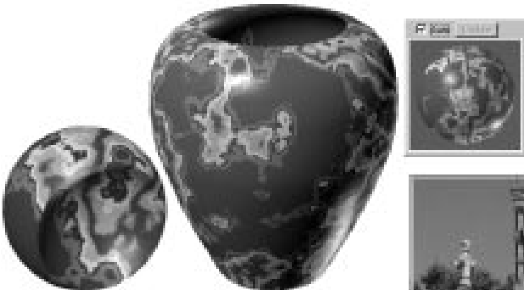
Bitmaps

The simplest kind of map is a bitmap. A bitmap is any of the image file types supported by 3D Studio MAX. These include still-image file formats, such as *.tga, .jpg, .bmp,* and so on, or an animation file such as *.avi, .flc,* or *.ifl*.

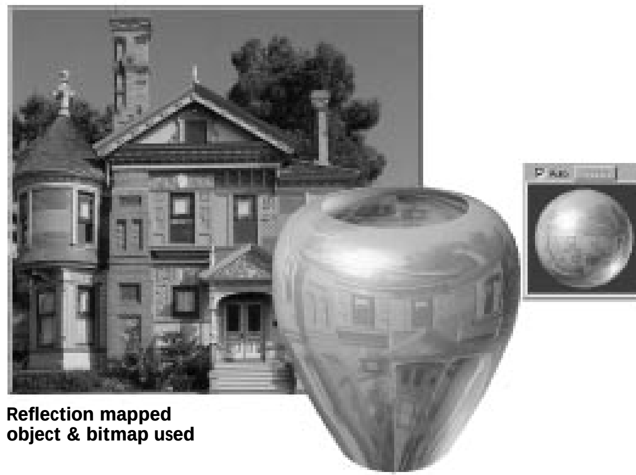
Procedural Maps

All maps that are not bitmaps are a form of procedural map. These maps generate an image based on parameters that you set.

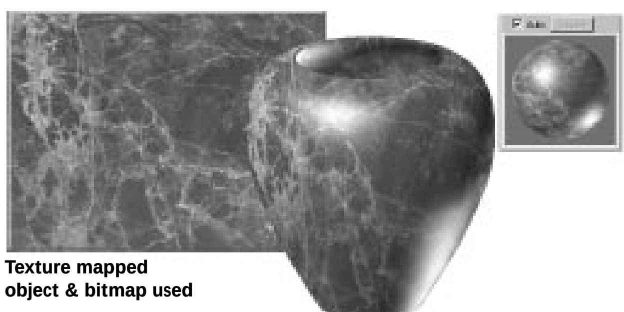
Most procedural maps can accept one or more maps as parameters. When you combine maps in this way you create a hierarchy of maps called a map tree. See, "Creating a Map Tree."



3D Perlin Marble



Reflection mapped object & bitmap used



Texture mapped object & bitmap used

Mask mapped Object & component maps used

Assigning Maps

You assign maps on the Material Editor one of two ways:

- As a material map used by one of the map buttons on the Maps rollout.
- As a stand-alone map in a sample slot.

Whichever method you use, your first step is to select a map from the Material/Map Browser. You use the Material/Map Browser to inspect maps in the Material Editor, in the scene, or in a material library (*.mat)* file.

Browsing for a Stand-Alone Map

Browsing for a stand-alone map is exactly like browsing for a material except that you choose a map from the browser.

To assign a stand-alone map to a Material Editor sample slot:

- Click Get Material on the Material Editor toolbar then drag a map type from the browser to a sample slot on the Material Editor. The chosen map is displayed in the active sample slot of the Material Editor.

If you select Scene in the Browse From area on the browser you can choose maps assigned as projection lights, environment maps, modifier maps, or any other map used in the scene.

Browsing for a Material Map

You choose between two versions of the Material/Map Browser depending on whether you want to assign a single map or multiple maps. After choosing a material map from the Browser, 3DS MAX places the map on one of the map buttons on the Maps rollout.

To assign a material map:

1. On the Maps rollout, click a material map button to display the modal Material/Map Browser.
2. On the browser, choose a map and click OK.

The modal Material/Map Browser shows only maps, and it closes when you click OK.

To assign multiple material maps:

- Choose Tools > Material/Map Browser, then drag maps from the browser to map buttons on the Maps rollout on the Material Editor.

The modeless Material/Map Browser stays open until you close it and can be used to drag maps to any number of map buttons for any number of materials.

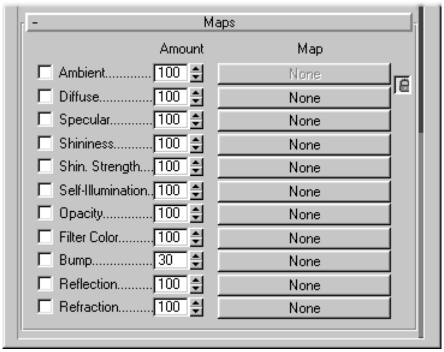
Browsing Bitmaps with the Asset Manager

If you select New on the Browse From areas on the Material/Map Browser and then select Bitmap, you get an empty bitmap that has no image file assigned. You must then select an image file on the Bitmap Parameters rollout.

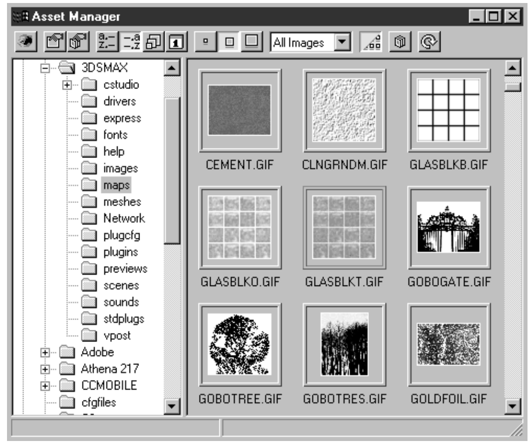
You can use the Asset Manager as a shortcut for assigning new bitmaps. By dragging image files from the Asset Manager to any map button, you automatically assign a bitmap as the map type and assign the selected image file a to the bitmap. See the Online Reference for details on using the Asset Manager.

Removing Maps

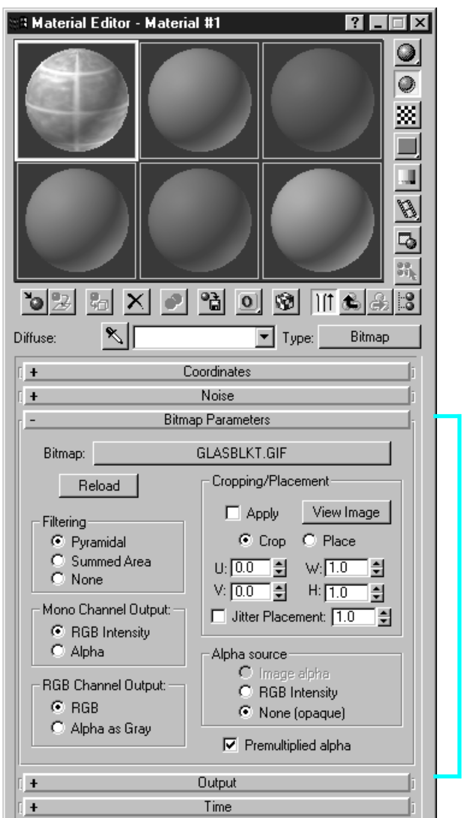
If you want to delete an assigned map, choose New in the browser, and drag the NONE item over the button containing the map you want to remove.



Maps rollout

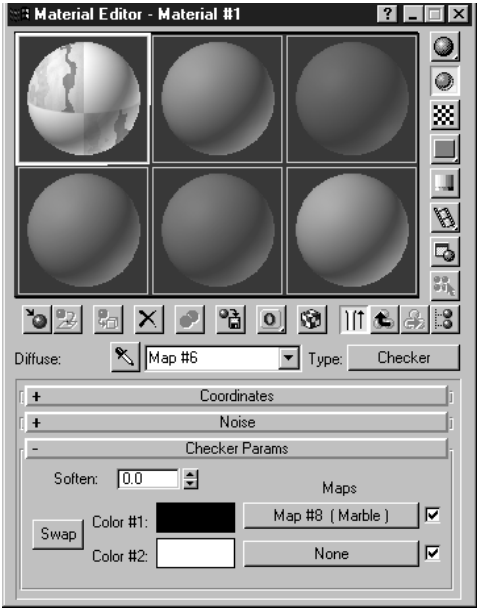


Asset Manager

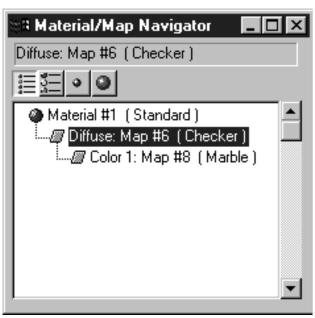


Parameters rollout for a diffuse bitmap

Creating a Map Tree



Checker Params rollout



Three level hierarchy in material map navigator

You create a map tree by assigning a map to a component of another map.

For example, the checker map is a checker pattern of two colors. You can assign a map to either of the two colors.

Create a Checker diffuse map:

1. On the Material Editor, click a sample slot.

2. Click the diffuse map button to assign a map.

3. On the Material/Map browser, choose Checker, and click OK.

The Checker Params rollout shows the default black and white colors of the checker map.

Create the map tree:

1. On the Checker Params rollout, click a map button to assign a map to a color.

2. On the Material/Map Browser, choose Marble, then click OK.

You now have a three-level *map tree* as follows:

- The material
- The Checker map used as a diffuse map
- The Marble map used as a component of Checker

The Material/Map Navigator displays this hierarchy.

You can now assign a map to one of the Marble color components, adding another level to the tree and creating a more complex material. In general, this is how you create a map tree: starting from a basic material, you assign maps and then in turn assign maps to components of the child maps.

Navigating a Map Tree

Once you begin adding maps to a material, you need a convenient way to navigate through the multiple levels you create. The Material Editor contains multiple tools for navigating material and map trees.

Using the Material/Map Navigator

Click Material/Map Navigator on the Material Editor toolbar to quickly move through a complex material or map tree.

The Material/Map Navigator is a miniature version of the Material/Map Browser that displays only the tree of the material or map in the active sample slot of the Material Editor. Clicking any level in the Material/Map Navigator takes you to the parameters for that level.

You can make the Material/Map Browser act like the Material/Map Navigator by selecting Active Slot in the Browse From area.

Navigating with Other Methods

The Material/Map Navigator is the most versatile method for navigating complex material and map trees. For simple materials you might want to use some of the other navigation tools in the Material Editor.

Material/Map Buttons—Materials and maps used as components of other materials and maps appear as buttons in the parent material's Parameters rollout. Click the materials or map button to move to that level of the tree.

Parent/Sibling Buttons—The buttons Go To Parent and Go To Sibling move you up the tree and horizontally across one level respectively.

Ancestor List—The name field shows the name of the material or map whose parameters are currently displayed. This field is also a list that shows the current map at the bottom with the names of ancestor levels in the map tree above.

Rendering a Map Tree

You can output the current mapping level of a material to an image file, such as a *.tga* or *.avi* if the map is animated. This feature is particularly useful for creating bitmaps from parametric maps, such as checker and marble, or for creating single maps from complex map trees.

To render the current mapping level:

1. Move to the level of the map hierarchy that you want to render.
2. Right-click in the sample slot, and choose Render Map from the shortcut menu.
3. On the Render Map dialog, set the rendering parameters, specify a file name for the map, then click Render.

A virtual frame buffer displays the rendered map. The rendering includes only the maps up to the current map level.

Creating Map Copies and Map Instances

As with geometric objects, a map copy is independent. Changes you make to it affect only the parameter to which the map is applied. A map instance, however, is linked to all other instances of the map. Changes you make to tiling or output level, for example, affect all other instances of the map.

- When you want two maps always to coincide, it is best to make the maps instances. Then if you adjust tiling, *UV* coordinates, and so on, both maps update to appear the same.
- When you don't want the settings for one map to affect other parameters, make the second map a copy.

Instancing Maps within the Same Material Level

You create map instances within the same level of a material by dragging one map button over another map button. When you release the drag you choose whether you want to Copy, Instance, or Swap the maps.

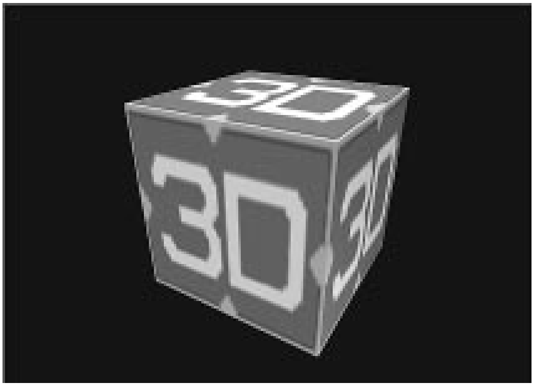
This technique only works for map within the same material and at the same level of the map tree.

Instancing Maps between Different Materials or Levels

You create map instance between multiple materials and levels of the map tree by dragging maps from the Material/Map Browser over a map button. If the source of the map comes from a material or map in the Material Editor or scene, you can choose to Instance or Copy the map.

Another option is to copy and paste maps in Track View. See chapter 25, "Basic Track View Use."

Applying 2D Mapping Coordinates to an Object



Default texture mapping for a box

Objects to which you assign a mapped material must have *mapping coordinates* if any of the maps are 2D maps. These coordinates specify how the 2D map is projected onto the material. They are known as *UV* or *UVW* coordinates to distinguish them from the *XYZ* coordinates that describe the scene's geometry.

Using Default Mapping Coordinates

Renderable objects have an option, Generate Mapping Coordinates, as a creation parameter. You select this for the object to render mapped materials. Each kind of object has its own default mapping coordinate settings.

Using UVW Map Modifiers

You apply the UVW Map modifier as follows:

- Use mapping coordinates other than the object default coordinates.
- Fine-tune mapping or apply mapping at the sub-object level.
- Animate mapping coordinate placement.
- Apply coordinates to objects without default coordinates, such as an imported mesh.

You change the parameters and transforms of the UVW Map modifier on the Modify panel.

Normally you apply the UVW Map modifier to a single object. You can apply it to a selection of multiple objects, but the position and alignment of the UVW Map gizmo is defined by *one* object in the selection set. The mapping that results is then applied to all the objects.

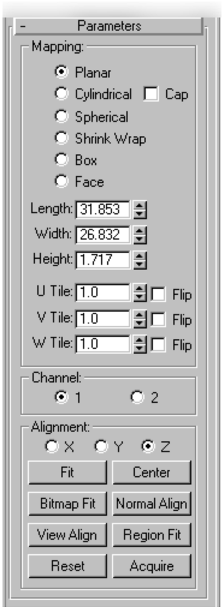
You apply other map modifiers to modify map tiling and scale, unwrap map coordinates, and match mapping to a camera view. See the online reference about map modifiers.

Double Coordinate Channels

You can apply two UVW mapping coordinate channels to an object. The coordinates themselves are assigned using a map modifier. At each map level of a material, you can choose which of the two UVW channels to use.

Use the two channels to combine two mapping projections on the same object. For example, you can use spherical projection for a primary map and planar projection for a decal map. See "Choosing 2D Mapping Projection," and "Creating Decals."

Choosing 2D Mapping Projection



Parameters of UVW Map modifier

Different kinds of mapping are distinguished by how the map is geometrically projected onto the object and how the projection interacts with the object's surfaces.

You choose the mapping projection on the Parameters rollout of the UVW Map modifier.

Planar

In planar mapping, the map is projected from a single plane—somewhat like projecting a slide. Planar projection is useful when only one side of an object needs to be mapped. It is also useful for obliquely mapping multiple sides, and for mapping two sides of a symmetrical object.

Cylindrical

In cylindrical mapping, the map is projected from a cylinder. The map wraps around the cylinder. You can usually see a seam where the edges of the bitmap meet. Cylindrical projection is useful for objects that are roughly cylindrical in shape.

By default, cylindrical mapping doesn't treat top or bottom faces separately. Top and bottom faces are faces whose normal is nearly parallel to the axis of the cylinder. Without capping, on top and bottom faces the map streaks radially toward the center of the cylinder. You can change this by setting Cap to on. Setting Cap on creates an additional mapping onto top and bottom faces of the object.

Note: If the ends of the object geometry are not at right angles to the sides, the Cap projection bleeds onto the sides of the object.

Spherical

Spherical mapping surrounds the object by projecting the map from a sphere. You see a seam and mapping singularities at the top and bottom of the sphere, where the bitmap edges meet at the sphere's poles. Spherical mapping is useful for objects that are roughly spherical in shape.

Shrink-Wrap

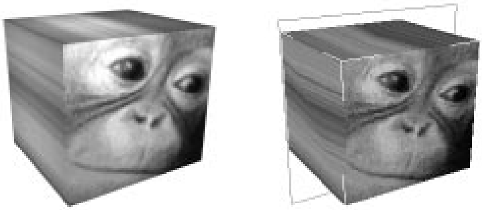
Shrink-wrap mapping is also spherical, but truncates the corners of the map and joins them all at a single pole, creating only one singularity. Shrink-wrap mapping is useful when you want to hide the mapping singularity.

Box

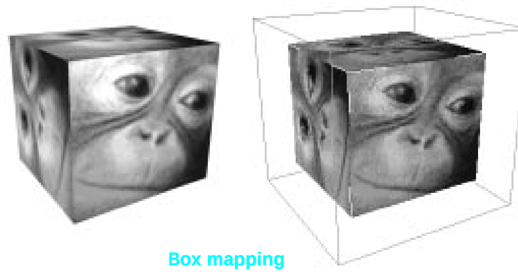
Box mapping projects from the six sides of a box; each side projects as a planar map, and the effect on the surface depends on the surface normal. Each face is mapped from the closest box surface whose normal most closely parallels its own normal.

Face

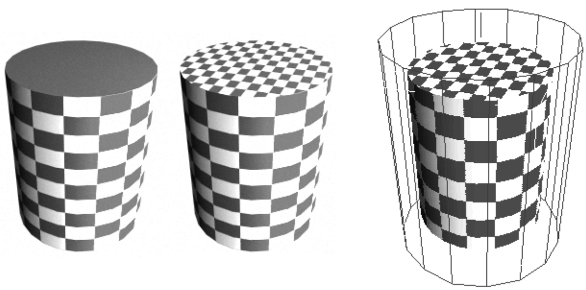
Face mapping applies a copy of the map to every face of an object. Pairs of faces, sharing a hidden edge are mapped with the full rectangular map. Single faces with no hidden edge are mapped with a triangular portion of the map.



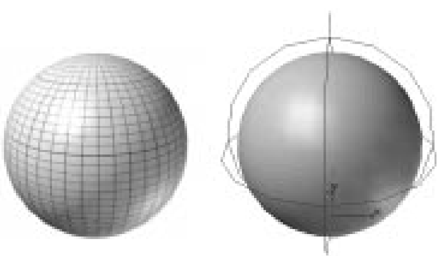
Planar mapping



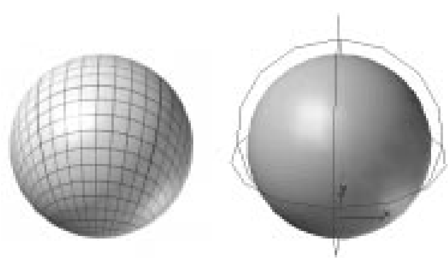
Box mapping



Cylindrical mapping with & without Cap.

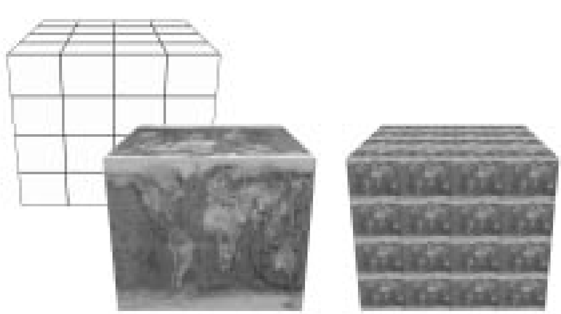


Spherical mapping



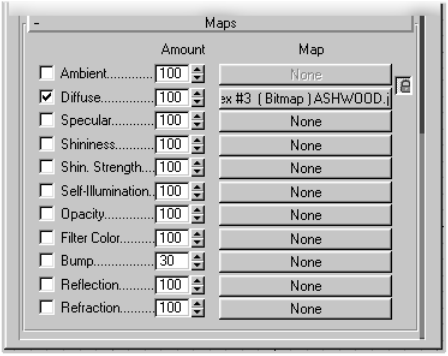
Shrink wrap mapping

Wireframe



With & without face mapping

Setting Map Strength and Amount



Material Editor Maps rollout

You can selectively turn maps on or off, and adjust their strength.

Turning a Map On and Off

After you assign a map, you can turn it off or back on. The controls for doing so appear in the Maps rollout of the parent material.

- The on/off check box for each kind of map is to the left of the name of the map type.
- You can also effectively turn a map off by setting its strength to 0. Use this technique to animate the on/off state of the map.

Adjusting Map Amount

The strength of a map is set by the Map Amount parameter on the Maps rollout.

Most maps define Map Amount as a percentage. A zero percent amount is equivalent to turning the map off.

Bump maps are an exception. Map Amount for Bump maps is an absolute value that can be positive or negative.

At intermediate values, the effect of the map mixes with the basic parameters of the material.

The following examples describe the effect of changing Map Amount:

Diffuse Example—Imagine you have applied a black and white Checker map to the diffuse channel of a red material. The Map Amount combines with the red diffuse color.

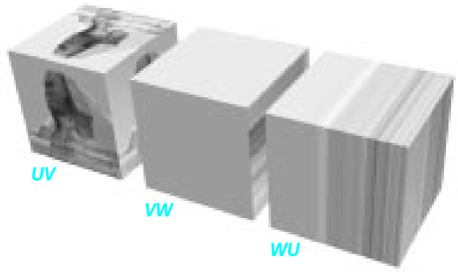
- A Map Amount of 100 colors the surface with black and white checkers.
- Reducing the Map Amount tints the surface with increasing amounts of the red diffuse color.

Opacity Example—Imagine you have applied a black and white Checker map to the Opacity channel of a material. The Map Amount combines with the Opacity parameter.

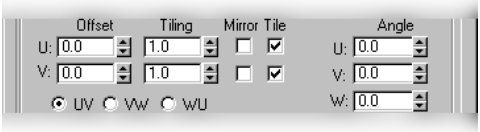
- A Map Amount of 100 creates a pattern of solid and transparent checkers over the surface regardless of the Opacity setting.
- Reducing the Map Amount causes both the transparent and opaque checkers to approach the Opacity value.

Self-Illumination Map Amount works the same as Opacity Map Amount.

Setting Map Position



Three possible 2D mapping planes



Coordinates rollout

Mapping coordinates define the base position of maps on an object. You change UVW parameters on the Coordinates rollout of the Material Editor to further adjust the position of maps.

Setting 2D Map UV Coordinates

For bitmaps, U and V values are based on the bitmap size. A value of 1.0 equals the width of the map in the U dimension and the height of the map in the V dimension. Because of this, U and V values that are multiples of 1 have no effect—the map is offset all the way around to its original location. Only the fractional portion of UV offset values affects the map's position. For example, setting U to 0.5 moves the map halfway across each surface of the box.

UV offsets are useful when you turn tiling off so that the bitmap appears in a single location. See "Creating Decals."

UV, VW, and WU radio buttons—Select which 2D plane is used for mapping. UV is selected by default. The other options change the map's orientation according to the other planes defined by the object's mapping coordinates. The *W* coordinate is at right angles to the *UV* plane.

Setting 3D Map UVW Coordinates

For 3D maps, the coordinate values are labeled XYZ or UVW based on which mapping coordinate option you choose.

XYZ—Uses world coordinates. The XYZ values offset the map in world units. This is the default setting for most 3D maps.

UVW1 and UVW2—Use mapping coordinates applied to the object. The UVW values offset the map by a percentage, like 2D UV values.

Choosing Mapping Channels

The method of choosing the mapping channel depends on whether you're working with a 2D map or a 3D map.

2D Maps—When Texture is chosen on the Coordinates rollout, you can choose Explicit UVW1 or Explicit UVW2 from the Mapping list. These choices match the Channel 1 and Channel 2 options of the UVW Map modifier.

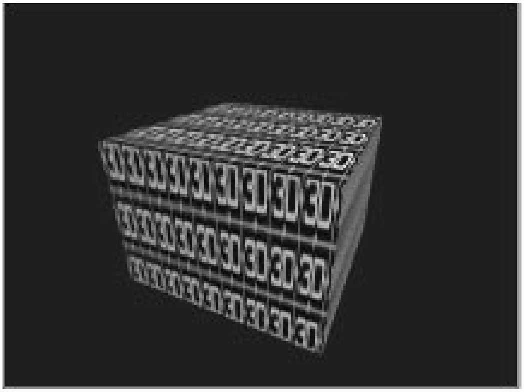
A third choice on the Mapping List is Planar XYZ From Object. This choice ignores explicit mapping coordinates and assumes planar mapping projection based on an object's local XYZ coordinates.

3D Maps—On the Coordinates rollout, choose either the UVW1 option or the UVW2 option.

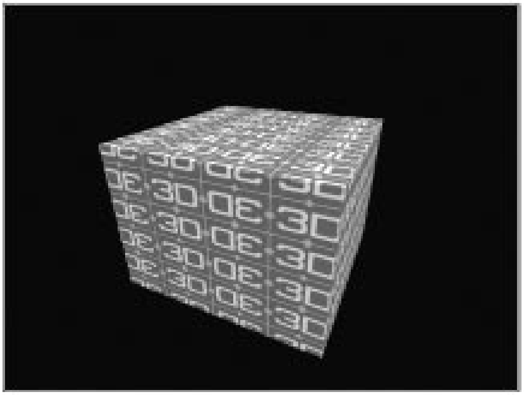
See the Online Reference for details on mapping coordinate channels.

Setting Map Tiling and Scale

Often when you apply a bitmap, especially as a texture pattern, you want the pattern to repeat. This effect is known as *tiling*, as in a tiled floor or fountain.The tiling parameters have different effects and options based on whether you are editing a 2D or 3D map.



Tiled mapping



Tiled & mirrored mapping

Decal mapping

Setting Map Tile Values

The Tiling value sets the scale of a map as the number of times the map is applied along a specified dimension. A value of 1.0, the default, applies the map once; a value of 2.0 applies the map twice. Fractional values apply a portion of the map. For example, a value of 2.5 applies the map two and a half times.

Another way to look at the Tiling value is to think of it as a scale value where the size of the map is divided by the Tiling value. For example, a Tiling value of 2.0 applies the map at one-half its original scale.

Choosing 2D Mirror or Tile Options

You can select two options that modify the effect of the Tiling value for 2D maps (such as bitmaps and Checker). These options are Tile and Mirror.

- Select the Tile option to cover the surface of an object with multiple copies of a map when the Tiling value is greater than 1.0. If you clear Tile, the map appears only once and is not repeated.
- Select the Mirror option to create two reflected images of the map for each original image. When Mirror is selected the meaning of the Tiling value changes. In a single mapping dimension (U or V), a value of 1.0, the default, shows two copies of the map; a value of 2.0 shows four copies; a value of 1.5 shows three copies; and so on. Mirroring in both dimensions multiplies the effect.

Choosing Filtering for Bitmaps

When the mapping scale for a bitmap is smaller than the bitmap's actual size, 3D Studio MAX must filter the bitmap to reduce its scale. Filtering is also used to antialias maps when they are rotated, or recede in perspective. You can choose the filtering method used.

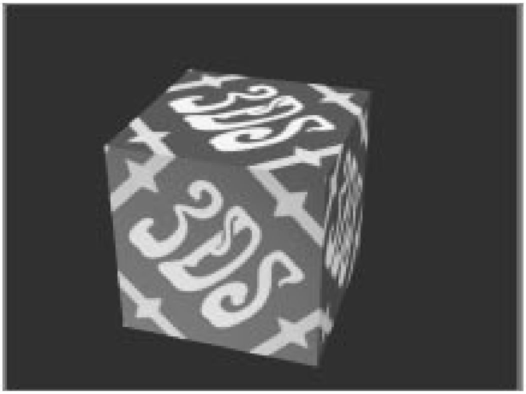
You select Filtering options of Pyramidal, Summed Area, or None on the Filtering area of the Bitmap Parameters rollout. See the online reference for details about the filtering options.

Creating Decals

Decals are useful for mapping single designs, small elements such as stickers or light switches, and so on. You can create decals by setting Tiling values and Tile options.

- Set both Tiling values greater than 1.0. This scales the map so it does not cover the entire surface.
- Clear the Tile and Mirror options. This prevents the map from repeating to fill the surface.
- Use the Offset values to position the decal on the surface.
- You may also want to use a second UVW channel or Planar From Object XYZ to assign a mapping projection for the decal that is different from the projection for the rest of the material.

Setting Map Rotation Angle



Rotated map

You can control the angle of a map by either rotating the gizmo of a map modifier or by changing the Angle parameters on the Coordinates rollout.

Rotating a UVW Map Modifier Gizmo

If you can rotate the mapping coordinates for all maps assigned to an object by rotating the gizmo of a UVW Map modifier. See chapter 6, “Basics of Creating and Modifying Objects,” for information about transforming modifier gizmos.

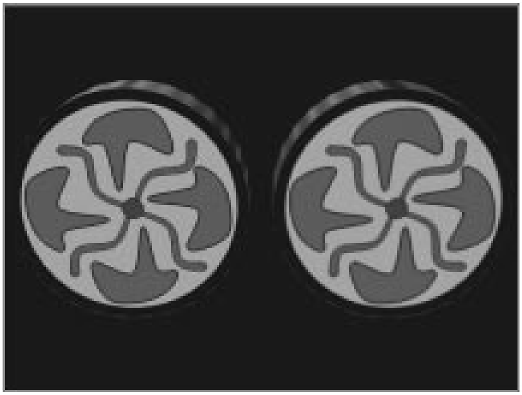
Setting the Angle Parameter for a Map

You set the Angle parameters on the Coordinates rollout of the Material Editor to control the rotation angle for a map.

2D and 3D maps both have Angle parameters for each UVW or XYZ coordinate axis. You change these parameters to change the map rotation angle about each axis.

For 2D maps, you can also click Rotate on the Coordinates rollout to display an “arcball” rotation icon. Drag on or around the arch ball to interactively rotate the map. You can see map rotation in the active sample slot.

Adding Blur



Map rendered without & with antialiasing blur

You can blur a mapped image using parameters on the Coordinates rollout.

Blur is meant primarily to antialias maps that appear too jagged when you render them. It bases blurring on distance in the scene, and is essentially a rendering effect.

The related parameter, Blur Offset, bases blurring on the map itself; it is more of an image-processing effect to enhance realism. For example, you can use Blur Offset on a bitmap used as a reflection map to simulate reflections from a surface that isn't completely shiny. Blur Offset can also sharpen an image.

- For texture maps, and maps besides bump maps, the Blur value is most effective in the 0.5–2.0 range. Lower values decrease antialiasing; higher values increase it.
- Blur Offset adjusts the image before antialiasing Blur is applied. If all you need is antialiasing, leave Blur Offset at its default of 0.0.
- Blur and Blur offset do not work if Filtering is set to None on the Bitmap Parameters rollout.
- Blur and Blur Offset do not work with non-antialiased 3D maps. (They do work with the Noise 3D map.)

To make a map image fuzzier:

- In the map's Coordinate's rollout, increase the Blur Offset value.

  Blur Offset is a very strong parameter. The Blur Offset spinner has increments of 0.001. Values greater than 0.1 are likely to be too high.

To make a map image sharper:

- In the map's Coordinates rollout, decrease the Blur Offset value to a value below 0.0.

  The negative Blur Offset value sharpens the image.

Blur Offset and Bump Maps

The Blur Offset parameter is especially useful with bump maps. The most effective Blur values to use with a bump map are in the range 0.3 through 0.6. However, this causes distant and obliquely viewed surfaces to blur more than nearby surfaces, which in turn causes bumpy surfaces to become smoother when they are farther away. Increasing the Blur Offset value for the bump map can correct this effect.

Setting Bitmap Transparency

Alpha source controls

Some bitmaps have an *alpha channel* that specifies transparency. A 32-bit bitmap file contains four channels of data: red, green, blue, and alpha. The first three provide color information to the pixels, while the alpha channel provides transparency information.

When you use a bitmap, you can choose to use or ignore any alpha channel information in the map. You can also generate pseudo alpha information from the color intensity of the map.

- Select Image Alpha to use the embedded alpha channel of a bitmap.
- Choose None (opaque) to ignore the embedded alpha channel.
- Choose RGB Intensity to create a new alpha channel based on the RGB intensity of each pixel. This option can be useful if the bitmap contains no alpha channel to begin with.

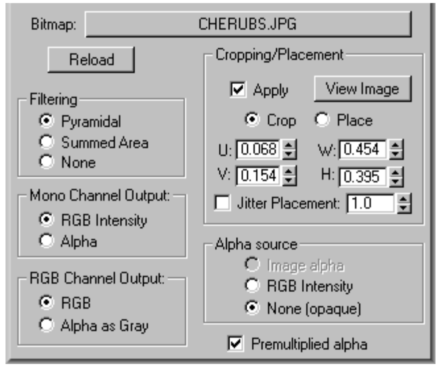
Choosing Premultiplied Alpha

The two methods of storing alpha in a bitmap are premultiplied and non-premultiplied. To composite an image that is in non-premultiplied format, the alpha must be multiplied by each of the R, G, and B channels before adding it in to the color of the background image. This provides the correct transparency effect, but must be done each time you composite. With pre-multiplied alpha, you store the R,G, and B components with the alpha already multiplied, so compositing is more efficient.

Select Premultiplied Alpha to specify that premultiplied alpha is expected in the bitmap. When turned off, alpha is treated as non-premultiplied, and any RGB values are ignored.

If you apply a bitmap with an alpha channel, and it doesn't decal correctly, it's probably because the bitmap file contains non-premultiplied alpha, and the RGB values are maintained separately from the alpha values. To correct this, clear Premultiplied Alpha.

Setting Bitmap Cropping and Placement



You use the controls in the Cropping/Placement area to crop a bitmap, or reduce its size. These controls change the size and position of a bitmap within the area set by the Tiling parameters.

Select Apply to see the results of cropping and placement in the sample slot and in the shaded viewports.

Cropping a Bitmap

Cropping a bitmap means to discard outer portions of the map to reduce it to a smaller rectangular area.

To crop a bitmap:

1. Select Crop, then click View Image to display the bitmap in the Specify Cropping/Placement window.

2. Specify a cropping region by dragging the corners, sides, or center of the region outline.

Placing a Bitmap

The placement function scales the size of the map and places it anywhere within its tile.

To place an image:

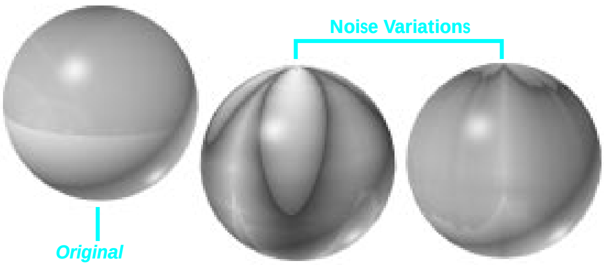
1. Select Place, then click View Image to display the bitmap in the Specify Cropping/Placement window.

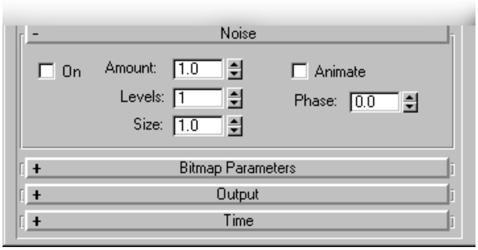
2. Drag the corners or sides of the region outline to scale the image.

3. Drag in the center of the region to move the image.

4. If you select Jitter Placement, you can set an amount of random offset for tiled images.

Adding Noise to a 2D Map





Noise rollout

You can add random noise to the appearance of your material. Noise perturbs the *UV* mapping of pixels by applying a fractal noise function.

Noise patterns can be very complex, and are a versatile way to create apparently random patterns. They are also good for simulating surfaces found in nature, as is characteristic of fractal images.

The noise controls are in a map's Noise rollout.

To add noise to a material:

1. On the Noise rollout, select On.

2. Change the noise parameters to get an effect you like.

Setting Noise Parameters

The noise parameters interact closely with each other. Slight variations in each can create noticeably different effects.

Amount—The strength of the noise function.

Levels—Or iterations—the number of times the function is applied.

Size—The scale of the noise function relative to geometry. At very small values, the noise effect becomes white noise. At large values, the scale can exceed the scale of the geometry, in which case it has little or no effect.

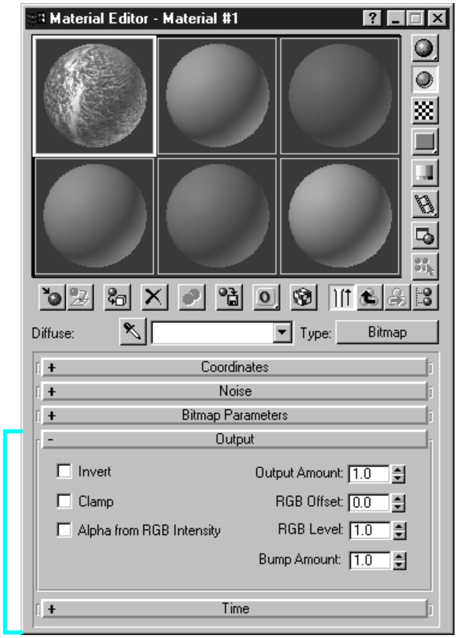
Animating Noise

You can animate noise using these methods:

- Animate the noise parameters of Amount, Levels, and Size directly. This method animates the size and scale of the noise pattern.

- Select the Animate option and animate the Phase value. This method creates a more subtle, organic animation by changing the boundaries of the noise pattern.

Setting Map Output



Output rollout

Map with inverted hues

The Output rollout for a map contains a few controls to adjust the map color globally and to change the map's effect on single-value parameters (as opposed to color components).

Output in this context refers to the output from the map to its parent material.

Invert—Inverts the hue of each pixel, changing it to its additive complement. Black becomes white, red becomes cyan, and so on.

Clamp—Limits the values of the colors to 1.0. Select Clamp when you increase RGB Level, but don't want the map to appear self-illuminated. If you set RGB Offset greater than 1.0 with Clamp selected, all colors become white.

Alpha from RGB Intensity—Generates an alpha channel based on the intensity of the map RGB channels. Black becomes transparent, and white opaque.

Output Amount—Changes the saturation and alpha value. At 1.0, the map is completely opaque. At 0.0 the map is fully transparent.

RGB Offset—A value added to the intensity of the map. Increasing the value increases overall brightness of the map until the map becomes white. Decreasing the value decreases brightness until the map becomes pure black.

RGB Level—Changes the saturation of the map without affecting the alpha value. Values greater than 1.0 increase saturation. Values less than 1.0 decrease saturation. At values below 0, the hue inverts and saturation increases.

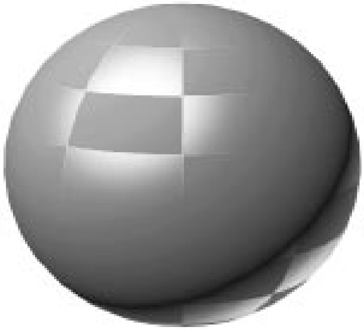
Bump Amount—Changes only the Bump channel, providing a finer level of control over bump maps. This parameters is used when you have a bump map made of a complex map tree. Use Bump Amount to change the bumpiness of a single component map.

Examples of Applying Maps

Applying Textures with Diffuse Maps

You use a diffuse map to assign a pattern or a texture to a material. The colors of the diffuse map replace the material's diffuse color component.

Usually you lock the Ambient map to use whatever is assigned to the Diffuse map. This isn't strictly necessary. By using a different Ambient map, you can obtain interesting blend effects. A good use for Ambient maps is to use a darker version of the Diffuse map to enhance shading.



Shininess mapping

Controlling Highlights with Maps

You can affect the Shininess, Shininess Strength, and Specular color of a material using maps.

- Assigning a bitmap to a material's specular color is similar to diffuse mapping, except that the pattern of the map appears as only in highlights.
- To affect *where* highlights appear, assign a map to the material's Shininess parameter.
- You can assign a map to the material's Shininess Strength parameter as well.

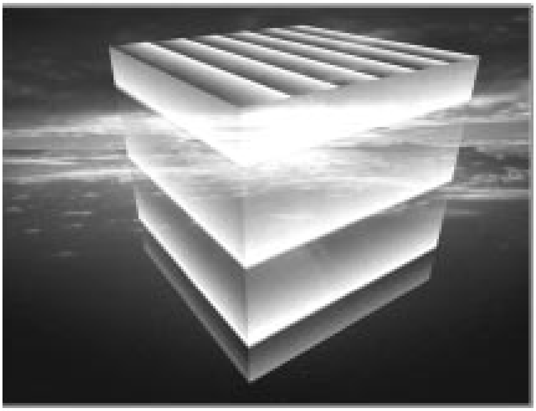
Tip: Shininess mapping usually works best when you assign the same map to both Shininess and Shininess Strength. (In the Maps rollout, you can do this by dragging and dropping.)

Transparency mapping

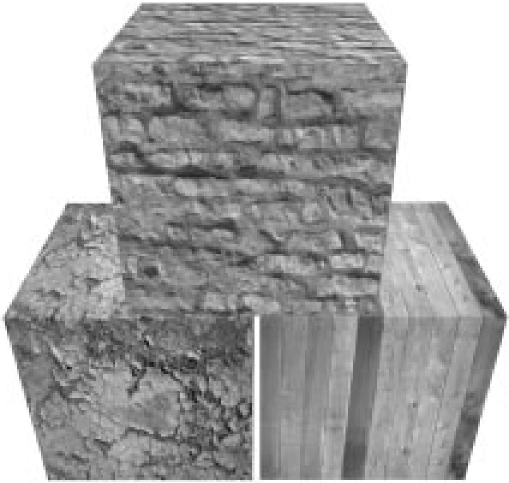
Making Objects Partially Transparent

You use an opacity map to make an object partially transparent. Lighter (higher-value) areas of the map render as opaque, while darker areas render as transparent.

Tip: Assign the same map as an opacity map and a shininess strength map. This prevents objects from being rendered with highlights in the "empty" portions the opacity map specifies.



Same bitmap used for transparency & self illumination mapping



Objects with the same images
used for diffuse & bump maps

Making a Partially Glowing Object

You use a self-illumination map to make portions of an object appear to glow. Lighter (higher-value) areas of the map render as self illuminating; darker areas have no effect on the surface—other material properties determine how it renders.

Self illumination means that the glowing area is not affected by lights in the scene (its ambient color component goes away), and does not cast or receive shadows.

Making Objects Bumpy

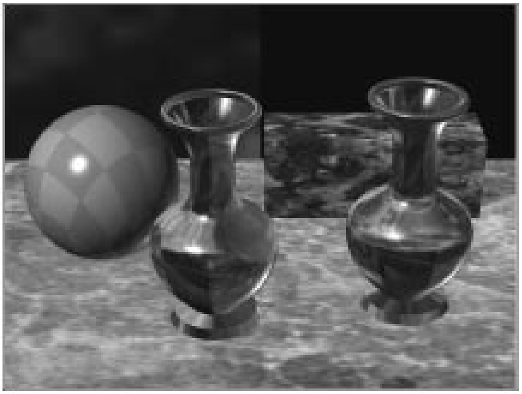
You use a bump map to make an object appear to have a bumpy or irregular surface. When you render an object with a bump-mapped material, lighter (whiter) areas of the map appear to be raised and darker (blacker) areas appear to be low.

Grayscale images can make effective bump maps. Maps that shade between white and black generally work better than maps with hard edges between the white and black areas.

Note: The bumps are a simulation created by perturbing face normals before the object is rendered. Because of this, bumps don't appear on the silhouette of bump-mapped objects.

Creating Reflections and Refractions

Reflection mapped objects



Refraction mapped objects

You use a reflection map to make an object appear as if it's reflecting another image. 3D Studio MAX provides multiple techniques for doing this:

- Use a standard 2D or 3D map to simulate a reflection. The map is an arbitrary image that does not depend on scene background or geometry.
- Use automatic maps, such as the Reflect/Refract map, to generate a reflection map automatically. The reflection is based on the scene's geometry and background.
- Use the Raytrace material or map to calculate highly accurate reflections and refractions using selective ray tracing.

Refractions are similar to reflections. Bitmaps simulate reflections, while Reflect/Refract maps generate them based on the scene's background and geometry.

Setting Reflection Amount

The default Reflection Map Amount is 100 percent. For most surfaces, however, reducing the amount gives more realistic results. For a glossy wooden bowl, you could use Wood as a diffuse map and a low-strength reflection map.

Setting Reflection Dimming

You set Reflection Dimming parameters to dim reflection maps that are in shadow.

By default, dimming is not applied and reflection-mapped materials do not respond to changes in light and shadow. Logically, a perfectly reflective surface always reflects its surroundings and is not affected by direct light. This can lead to some unrealistic situations in which a reflection map takes on the attributes of a self-illuminated material.

You enable reflection dimming by selecting Apply on the Extended Parameters rollout, and then setting the following:

Dimming Level—controls the amount of dimming that takes place in shadow. When set to 0.0, the reflection is completely dark in shadow. When set to 0.5, the reflection map is half dimmed in shadow.

Reflection Level—controls intensity of the reflection that is not in shadow. When the Dim Level is less than 1, the reflection is not only affected in the shadows, but in the lit areas as well. To compensate for this, the Reflection Level value multiplies the illumination level of the lit area of the reflection.

Tinting and Intensity of Reflection Maps

Reflection maps are affected by the Ambient, Diffuse, and Specular colors even when the Reflection Map Amount is set to 100 percent.

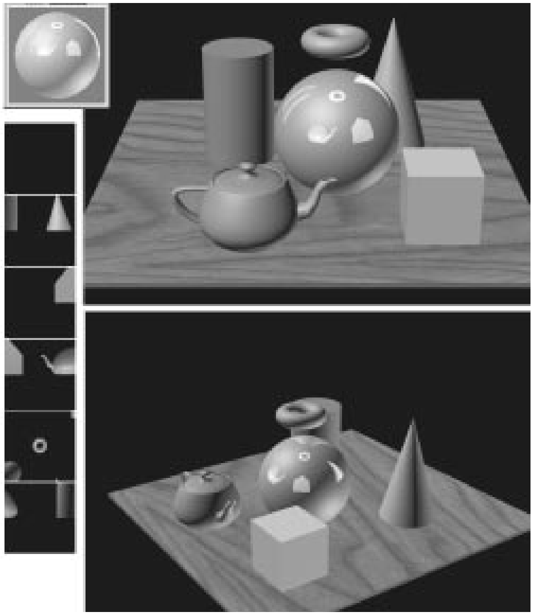
- Ambient, Diffuse, and Specular colors tint reflection maps.
- The Specular Value parameter affects reflection intensity of non-Metal materials. A Specular Value of 255 shows the reflection map at full intensity, and a Specular Value of 0 turns the map off.
- The Diffuse Value parameter affects reflection intensity of Metal materials. A Diffuse Value of 255 shows the reflection map at full intensity, and a Diffuse Value of 0 turns the map off.

Setting the Index of Refraction

The physical properties of refractive objects often distort the image. 3DS MAX has a special parameter to adjust this distortion. It is in the parent material's Extended Parameters rollout.

Refract Map/Raytrace IOR—IOR stands for *index of refraction*. An index of 1.0 is comparable to air, and does not refract. At values greater than 1.0, the refraction tends to become more convex; at values less than 1.0, the refraction tends to become more concave.

Using Bitmaps for Reflection and Refraction



You assign standard 2D and 3D maps as reflection or refraction maps when you want the best rendering speed and are not too worried about accurate effects.

Assigning Bitmaps or Procedural Maps

You assign a bitmap or procedural map as a reflection or refraction map to simulate generic reflections.

Procedural maps like Noise and general bitmaps like clouds, sunsets, and landscapes work well as reflection maps.

Assigning Cubic Maps

Cubic maps are a collection of six standard maps that 3D Studio MAX stitches together to create a more realistic reflection.

You assign cubic maps by first assigning a Reflect/Refract map and selecting From File as the source on the Reflect/Refract Parameters rollout. You then click one of the six buttons in the From File area to display a file selector from which you select and assign the maps.

Each of the six assigned maps in the cubic map must be square, and must be the same size.

Rendering Cubic Map Files

You can render cubic maps from the Reflect/Refract Parameters rollout. This creates a realistic cubic reflection map centered on a selected object. The generated cubic map is a good compromise between the speed of using a generic map and the accuracy of using automatic reflections.

To render a cubic map:

1. On the Reflect/Refract Parameters rollout, choose the From File option.
2. On the Render Cubic Map Files area, click To File to display a file dialog, then enter the name you want for one of the six cubic bitmap files.
3. Click Pick Object And Render Maps, then pick an object in your scene from which you want the six views rendered. This is usually the same object on which you will apply the material, but it doesn't have to be.

The six views are rendered and saved to disk. Their filenames then appear in the six buttons in the From File area.

Creating Automatic Reflection and Refraction

Refractions show the scene or background behind the refractive object

You generate automatic reflection and refraction maps by assigning a Reflect/Refract map and selecting Automatic as the source on the Reflect/Refract Parameters rollout. As a rule, automatic reflections take longer to render than bitmapped reflections and refractions.

Automatic reflection and refraction maps create temporary cubic maps of the scene for every object using the material.

Setting Reflect/Refract Parameters

You control the automatic map size, blurriness, and frequency of generation by setting parameters on the Reflect/Refract Parameters rollout. See the online reference for full details about these parameters.

- A reflective object can reflect another reflective object. You can set the number of interreflections from 1 to 10. You set this Rendering Iterations parameter in the Render Scene dialog.
- Automatic reflection does not support self-reflection. Generated cubic maps show other objects in the scene, but do not show any part of the object assigned the Reflect/Refract material. For example, the handle of a chrome teapot will not appear in the reflection on the body of the teapot unless you detach the handle as a separate object.
- Automatic refraction works effectively with geometry behind the object and with environment-mapped backgrounds. It does not effectively simulate an object *in* a refractive medium, such as a pencil in a glass of water, for example.

Assigning Thin Wall Refraction

Assign the Thin Wall Refraction map for a superior method of refraction mapping that provides the type of jog, or offset effect, you'd find when viewing part of an image through a plate of glass. Compared to Reflect/Refract mapping, this Thin Wall mapping is faster, uses less memory, and provides a much better visual effect.

Thin Wall refraction through flat object

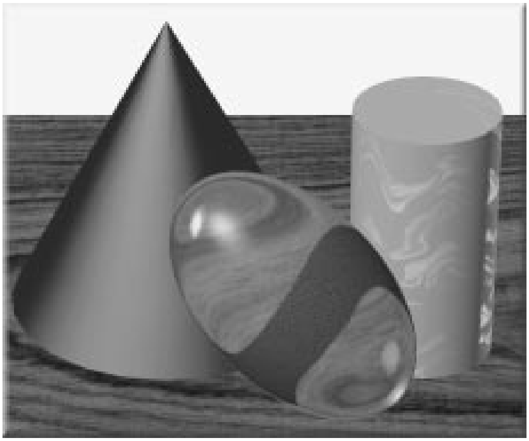
Setting Thin Wall Parameters

Thin Wall mapping uses controls similar to those of Reflect/Refract mapping for blurring and frequency of map generation. Additional controls are:

Thickness Offset—Affects the size of the refractive offset, or jog effect. When set to 0, there's no offset effect.

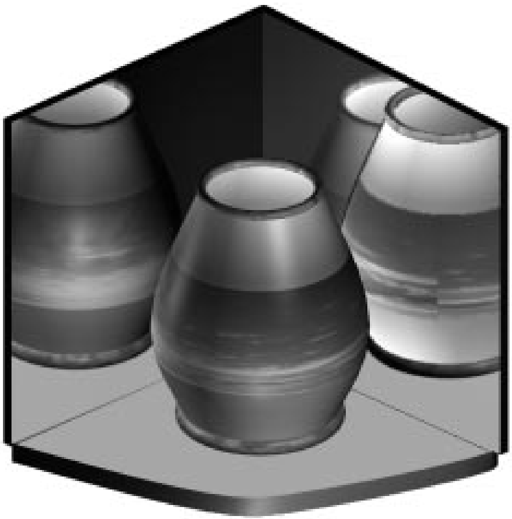
Bump Map Effect—Affects the size of the refraction effect due to the current bump map.

To see the effect of the Thin Wall Refraction map, create a scene that includes a thin box as a wall, and set up the wall so it is partially in front of other geometry. Assign a material to the box with the Thin Wall Refraction map, and render the scene. You will see a jog between the geometry viewed through the wall, and the geometry not viewed through it.



Thin Wall refraction through rounded object

Creating Flat Mirrors



Reflective faces on an object created with Flat Mirror

You use a Flat Mirror map to generate reflections on a flat surface. Reflect/Refract maps don't work well for flat surfaces because each face reflects a small part of the environment based on where its face normal points. Using this technique, a large flat face can reflect only a small part of the environment. Flat Mirror generates a reflection using a larger part of the environment.

Setting Flat Mirror Parameters

Flat Mirror controls on the Flat Mirror Parameters rollout are similar to those for automatic reflection and refraction.

You also can set controls to distort the flat-mirror reflections. Your choices are:

- No distortion
- Distortion based on the assigned Bump map
- Distortion based on noise parameters on the Flat Mirror Parameters rollout

Rules for Using Flat Mirror

Flat Mirror cannot generate reflections correctly unless you observe these rules:

- If you assign the Flat Mirror material to multiple faces, the faces must lie in a plane.
- Non-coplanar faces in the same object cannot have the same flat mirror material.
- Because of the preceding restriction, make sure that the material ID used by the Flat Mirror sub-material is unique to the coplanar faces in the object.

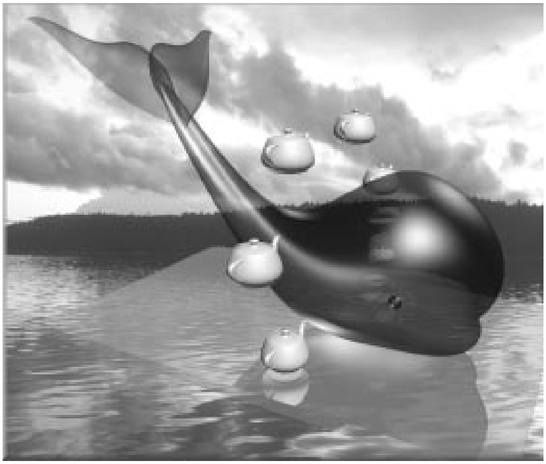
Flat Mirror ID Assignment

On the Flat Mirror Parameters rollout you specify the material ID number to which you want the mirror assigned. The other faces on the object use the properties of the material except for the Flat Mirror reflection map. If your object needs completely different material characteristics, or multiple flat mirrors, you need to use a Multi/Sub-Object material.

See, "Multiple Materials on a Single Object."

For example, if you have an object such as a box, which is created with unique material IDs for each side, you can specify the material ID for the side of the box containing the mirror reflection. The remaining sides of the box will have the same material characteristics, but without the reflection.

Ray Tracing Reflection and Refraction



Raytrace material applied to glass whale model and underlying rectangle

Ray tracing is a rendering method that traces the path rays of light take through the scene until they reach your eye. As the rays of light bounce from one object to the next, they pick up information about color, intensity, and reflection from each object they strike.

This method of rendering produces extremely accurate reflection and refraction but takes longer to calculate than other methods.

Choosing the Raytrace Material or Map

The Raytrace material and Raytrace map add selective ray tracing to 3D Studio MAX. Raytrace calculations are only performed for objects that are assigned a Raytrace material or Raytrace map. This technique improves the performance of rendering with ray tracing.

Assign the Raytrace material or Raytrace map only to prominent objects in your scene that can take advantage of raytracing.

- Choose Raytrace material when you want to define a standard ray-traced reflective or refractive material and change the minimum parameters necessary.
- Choose Raytrace map when you want full control to define a custom ray-traced material. You can also apply a Raytrace map type to other maps such as Ambient or Diffuse for custom effects.

Setting Raytrace Parameters

Raytrace material and Raytrace map contain multiple rollouts with parameters to control properties of the raytracing process. You can control the following properties:

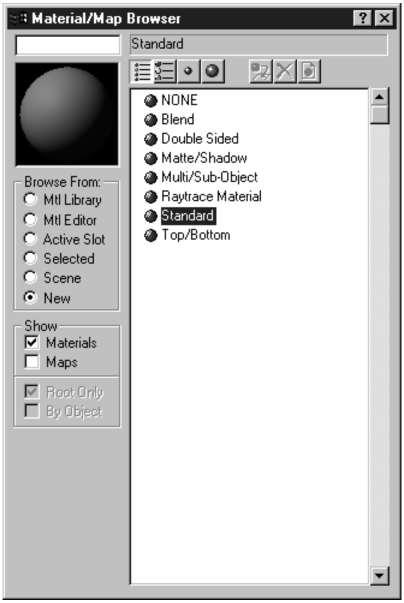
- Self-reflection and environment handling
- Anti-aliasing, blur, and defocusing
- Refractive color and density
- Performance of the ray trace solution

See the Online Reference for details about Raytrace material and map parameters.

Including and Excluding Objects

Another way to improve the performance of raytraced rendering is to include or exclude objects from the solution. You use the Raytrace Include/Exclude dialog to identify objects that can be ignored by the ray trace solution. Excluded objects do not appear in reflections or refractions.

Choosing Different Kinds of Materials



Material/Map Browser dialog showing all material types

Different types of materials create different effects, behave in particular ways, or are provided as ways to combine multiple materials.

Compound materials are similar to compound maps. The difference is that compound materials exist at the material level—applying a compound material to an object inherently creates a compound effect that often uses mapping. You load or create compound materials using the Material/Map Browser.

Standard

Standard materials are based on the component-color controls described in chapter 20, "Designing Basic Materials." However, you can add complexity to them by assigning maps to their color components or parameters.

Multi/Sub-Object

Multi/Sub-Object materials are a way to assign more than one material to the same object. The Multi/Sub-Object material stores two or more sub-materials, which you assign at the sub-object level by using the Mesh Select modifier. You can also assign the sub-materials to whole objects by using the Material modifier.

Raytrace

Raytrace materials are used for default ray-traced reflective or refractive materials. See, "Ray Tracing Reflection and Refraction."

Top/Bottom

Top/Bottom materials store two materials. One material is rendered on an object's top faces and the other on the object's bottom faces, depending on whether a face normal points up or down.

Double Sided

Double Sided materials also store two materials. One material is rendered on the object's outer faces (the usual side for one-sided materials, as determined by face normals), and the other is rendered on the object's inner faces.

Blend

Blend materials, like Mix maps, combine two materials by mixing their pixel colors.

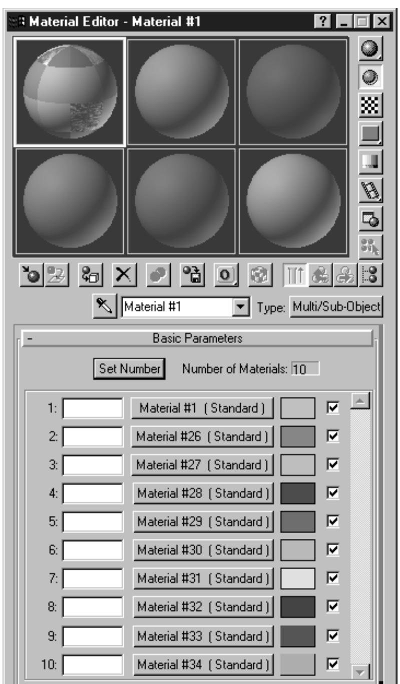
Matte/Shadow

Matte/Shadow materials are used for creating matte objects to use with environment maps and Video Post compositing.

See the Online Reference for details on all material types.

Multiple Materials on a Single Object

Object with multiple materials assigned via Multi/Sub-Object



Basic Parameters rollout for Multi/Sub-Object materials

A Multi/Sub-Object material is a collection of materials you can use to assign different materials to different faces of an object at the sub-object level.

Creating a Multi/Sub-Object Material

The number of a sub-material in a Multi/Sub-Object material corresponds to a material ID. For example, the material in slot 1 is assigned to object faces with material ID 1, the material in slot 2 is assigned to object faces with ID 2, and so on.

To create a Multi/Sub-Object material:

- On the Material/Map Browser, choose Multi/Sub-Object and click OK.

The controls for a Multi/Sub-Object material are essentially a list of the materials it contains.

To assign a sub-material:

- On the Basic Parameters rollout, click a material button.

Assigning Multi/Sub-Object Materials

Sub-object material assignment is based on *material ID* numbers. Geometric objects have default material IDs, just as they have default smoothing groups. The material ID assignment depends on the type of geometry. For example, most curved objects such as spheres have a single material ID by default; boxes have six IDs—one for each face; and Hedra have three—one for each of their *P*, *Q*, and *R* axes.

You control Multi/Sub-Object material assignment using the following method:

- Examine the Material ID assignment on an object using the Mesh Select modifier.
- Optionally change Material IDs, using the Mesh Select and Material modifiers, to define which faces receive different materials.

- Create a Multi/Sub-Object material with a sub-material for each of the Material IDs on the object.
- Assign the Multi/Sub-Object material to the object.

If your object has been collapsed to an Editable Mesh, you can assign Material IDs to faces in the object's base parameters.

Attaching Objects with Multi/Sub-Object Materials

When you attach objects with assigned materials, their materials are combined to create a new Multi/Sub-Object material.

You control how the new Multi/Sub-Object material is defined by choosing options on the Attach Options dialog.

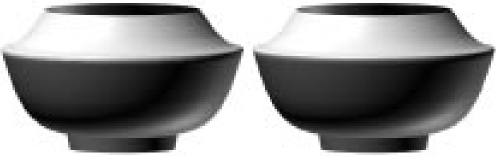
Match Material IDs to Material—Changes material ID on the attached objects to match the submaterials of the new Multi/Sub-Object material. This is your recommended choice.

Match Material to Material IDs—Organizes submaterials in the new Multi/Sub-Object material to match the original material IDs of the attached objects.

Do Not Modify Mat IDs or Material—Combines materials and material IDs with no change. Not recommended except to maintain compatibility with earlier versions of 3DS MAX.

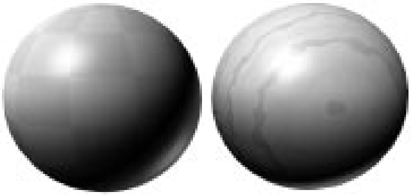
See chapter 12, "Editing Meshes," and the Online Reference, for more about attaching objects and assigning Material IDs.

Combining Materials



Top/Bottom materials

Double Sided material



Blend material

Creating a Top/Bottom Material

Top/Bottom materials contain two materials. One is rendered on the object's top faces (those with normals pointing upward). The other material is rendered on the object's bottom faces (those with normals pointing downward).

You can choose whether "up" and "down" are defined by world or local coordinates.

Creating a Double Sided Material

Double Sided materials contain two materials. One is rendered on the object's outer faces (the usual side for one-sided materials, as determined by face normals). The other material is rendered on the object's inner faces.

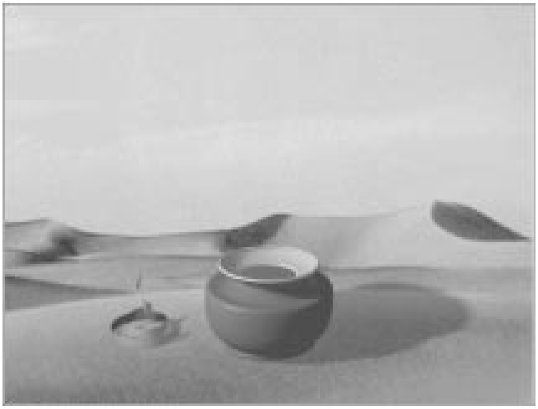
You can set the Translucency parameter to affect the blending of the two materials.

Creating a Blend Material

Blend materials blend two other materials. You control the blending by choosing a mask map and by setting parameters for a Mix Curve.

Tip: Using a Noise map as the blend mask can produce good effects with a natural appearance.

Travelling Mattes and Catching Shadows



Scene with matte-created shadows

In filmmaking, a *matte* is a compositing tool.It masks part of a scene so that the masked area can later be filled with a different background. For example, a singer filmed in a studio is shown singing on a mountain in the finished scene.

The Matte/Shadow material performs a similar function. Faces with this material reveal the environment map when you render the scene. The faces are effectively invisible against the background, but they block geometry behind them. They can receive shadows from other objects and can be affected by the current atmosphere. You use mattes to make the environment appear to be part of the scene.

Objects with Matte/Shadow material appear gray in rendered viewports. You only see the Matte/Shadow effect in the final rendering.

To create a matte:

1. Create the object to use as a matte. If you want the matte to receive shadows, make the object shape similar to the shadow-receiving portion of the background.

For example, the ground and standing objects toward the front of the background usually receive shadows. The sky, and objects in the distant background, usually do not.

Displaying the background in the viewport can help you shape the object.

2. Use the Material Editor to assign Matte/Shadow material to the object.

You can also assign Matte/Shadow material to a sub-object selection.

Controlling Shadows

You can set parameters to control the handling of shadows on a Matte/Shadow material.

Affect Alpha—Sets whether shadows cast on a matte material are applied to the alpha channel. Use this option to render bitmaps with alpha shadows for compositing.

Shadow Brightness—Set this value to alter the apparent brightness of the shadows. The higher the value, the more transparent the shadow, allowing the background to show through.

Color—Sets the color of the shadow. In most cases, it’s best to use a very dark color, close to black. This feature is particularly useful when you're using a Matte/Shadow material to composite your shadows against a background image such as video. You can tint your shadows to match shadows in the image.

Animating Materials

This topic gives some suggestions about how to animate materials.

Animating Basic Materials

In general, you animate a basic material by changing its parameters in different keyframes while the Animate button is active. 3D Studio MAX interpolates values between keyframes, as it does when you animate transforms and modifiers.

Be aware that the tracks for a material assigned to an object are distinct from the material tracks that belong to the Material Editor. Animating a material in the Material Editor affects the scene only if the material is hot.

Animating Mapped Materials

As with basic materials, you can create animation keys for map parameters.

The noise parameters and the Noise map itself provide the Phase parameter specifically for animating the noise function.

Changing One Material into Another

Probably the easiest way to animate the change of one material into another is to create a Blend material, make the two other materials its sub-materials, and then animate its Mix Amount parameter.

Preview and Playback

When you animate a material, or use an animated bitmap in a material, you can create and view a preview of this material before you decide to use it in a fully rendered animation.

You can create a preview movie of an animated material using the Material Editor. These controls are on the Make Preview flyout.

Preview flyout

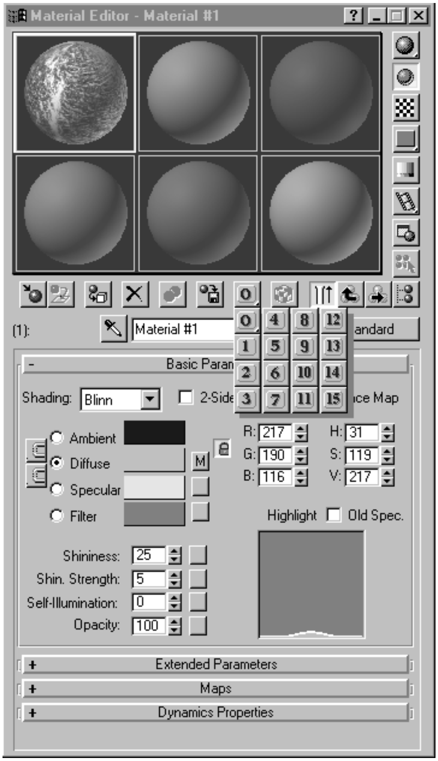
The active preview movie is saved in a file called *_medit.avi*. Each time you create a new preview, the Material Editor overwrites this file. To keep a preview movie on hand, save it under a different name.

Synchronizing an Animated Bitmap with the Scene

Maps have a Time rollout with parameters that you set to synchronize an animated map with scene animation.

Time controls for bitmaps

Using Effects Channels



Effects Channel flyout in the Material Editor

You assign an Effects Channel to a material in order to designate it for use in video post-processing. The Video Post *events* Image Filter and Image Layer can use designated materials for masking. Some events also post-process materials with a designated effects channel, to generate special effects.

To assign an effects channel to a material:

- Choose a channel number from the Effects Channel flyout.

The channel number can be from 1 to 15. Zero (0) is a special case that means the material has no effects channel.

If you assign the same effects channel number to more than one material, all materials with that number are used in Filter or Layer masking, or receive special post-processing.

See chapter 31, “Using Video Post,” for about Image Filter and Image Layer events.

Note: Assigning a material an effects channel number greater than zero tells the Renderer to generate an RLA channel containing such numbers. This information is saved only if you save the rendered scene to a *.rla* file. However, the effects channel data is available to Video Post when it post-processes your rendered scene.

Chapter

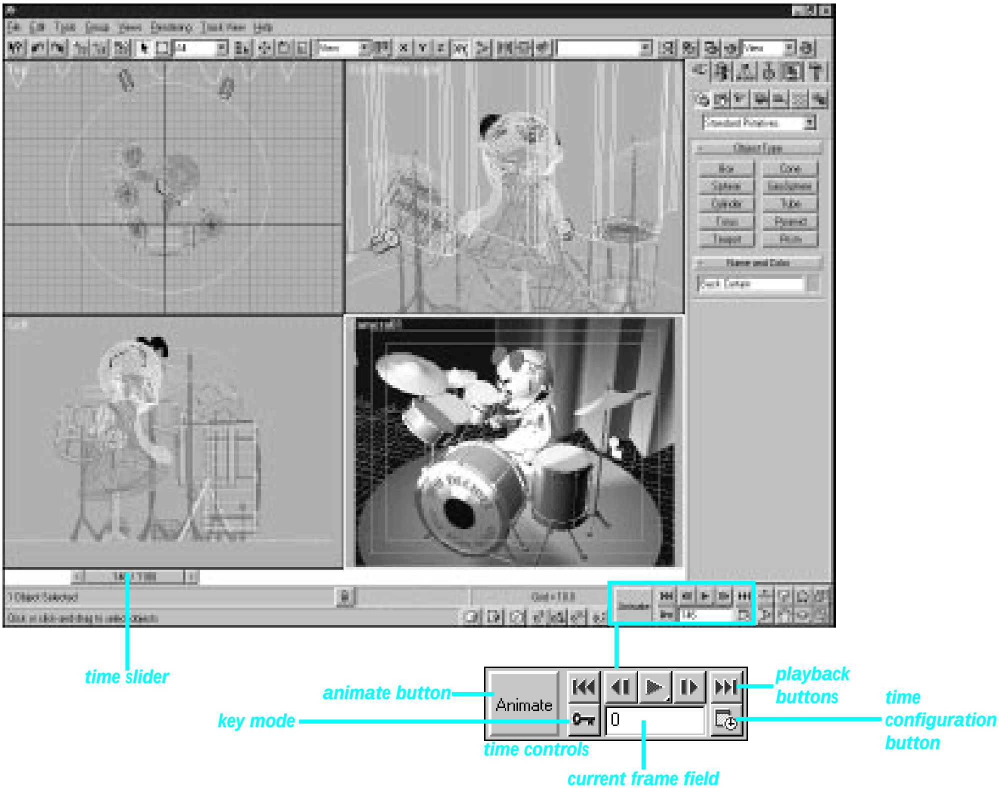
22

Animation Concepts and Methods

Animation is used throughout 3D Studio MAX. You can animate the position, rotation, and scale of an object, and almost any parameter setting that affects an object's shape and surface. You can link objects for hierarchical animation, using both forward and inverse kinematics, and you can edit your animation in Track View.

This chapter discusses the basics of creating animation with 3D Studio MAX. It looks briefly at a comparison between computer animation and classic hand-drawn animation, and then describes the creation of *keyframed* animation, using the Animate button.

Animation Concepts



Animation is based on a principle of human vision. If you view a series of related still images in quick succession, you perceive them as continuous motion. Each individual image is referred to as a *frame*.

Traditional Animation Method

Historically, the main difficulty in creating animation has been the effort required of the animator to produce a large number of frames. One minute of animation might require between 720 and 1800 separate images, depending on the quality of the animation. Creating images by hand is a big job. That's where the technique of *keyframing* comes in.

Most of the frames in an animation are routine, directed toward some goal. Traditional animation studios realized they could increase the productivity of their master artists by having them draw only the important frames, called *keyframes*. Assistants could then figure out what belonged on the frames in between the keyframes. The in-between frames were called tweens.

Once all of the keyframes and tweens were drawn, the images had to be inked or rendered to produce the final images. Even today, production of a traditional animation usually requires hundreds of artists to generate the thousands of images needed.

The 3D Studio MAX Method

3D Studio MAX is your animation assistant. As the master animator, you create the keyframes that record the beginning and end of each animated sequence. The values at these keyframes are called *keys*. 3DS MAX calculates the interpolated values between each key to produce the completed animation.

3DS MAX can animate just about any parameter in your scene. You can animate modifier parameters, such as a Bend angle or a Taper amount, material parameters, such as the color or transparency of an object, and much more.

Once you have specified your animation parameters, the 3D Studio MAX renderer takes over the job of shading and rendering each frame. The result is a high-quality animation.

Comparing Frames and Time

Traditional animation methods, and early computer animation programs, are rigidly locked to the concept of producing animation frame by frame. This is OK if you always work in a single format or do not need to specify an animated effect at a precise time.

Unfortunately, animation comes in many formats. Two of the more common formats are film at 24 frames per second (FPS) and video at 30 FPS. Also, the need for accurate time-based animation versus frame-based animation is critical as animation becomes more common for scientific and legal presentation.

3D Studio MAX is a time-based animation program. 3DS MAX measures time, and stores your animation values, at 1/4800 of a second. You can configure 3DS MAX to display time in a format best suited for your work, including traditional frames format.

Many of the examples in the following chapters describe time using the frames method for the sake of tradition and familiarity. Keep in mind that you are really animating using a very precise time-based method, and frames aren't created until you instruct 3DS MAX to render your animation.

Identifying 3DS MAX Animation Tools

You can locate the basic animation tools in the following areas of the 3DS MAX interface.

Track View—Provides detailed animation editing capabilities in a floating window. See chapter 25, "Basic Track View Use."

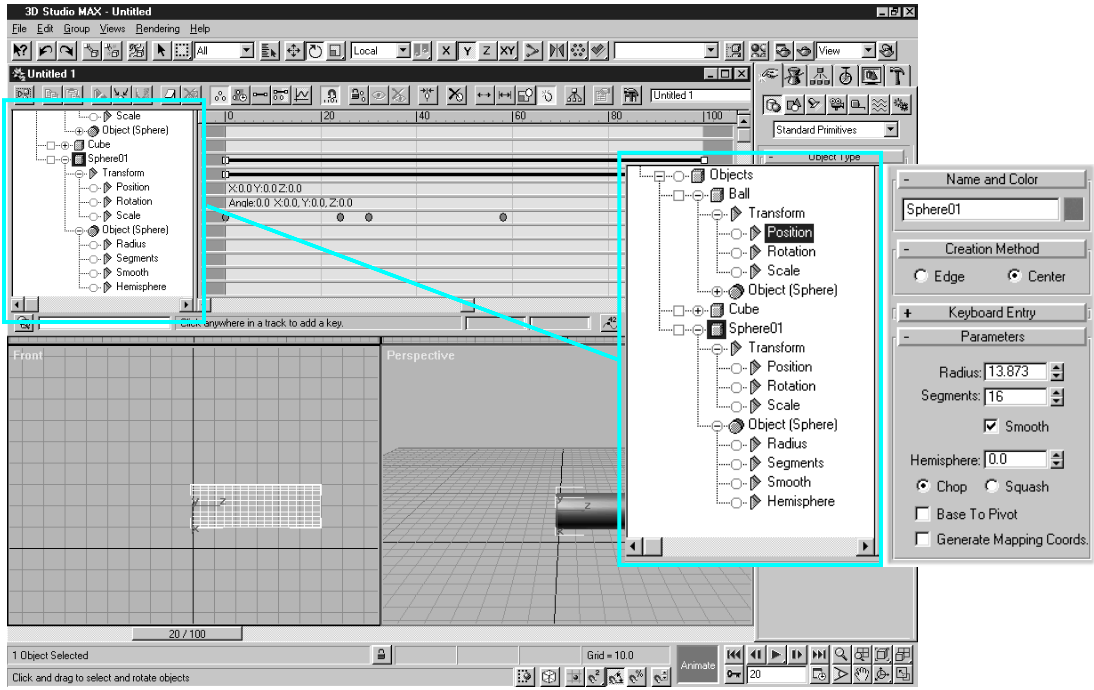
Motion Panel—Use this panel to adjust transform controllers that affect all position, rotation, and scale animation. See chapter 27, "Working with Controllers."

Hierarchy Panel—Use this panel to adjust all parameters governing the linkage of two or more objects. These include inverse kinematics parameters and pivot point adjustments.

Time Controls—Use these controls to move through time in the viewport displays. You can move to any point in time, and play animations in the viewports. The time controls include:

- Time Slider
- Animate button
- Animation playback buttons
- Current time field
- Key Mode toggle
- Time Configuration button

Using the Animate Button



You produce animation in 3D Studio MAX by turning on the Animate button, setting a current time, and then changing something in your scene. You can change the position, rotation, or scale of an object, or change almost any setting or parameter.

When you make a change, a key storing the new value for the changed parameter is created at the current time. If that key was the first animation key created for the parameter, a second animation key is also created at time 0 to hold the parameter's original value.

Keys are not created at time 0 until you create at least one key at another time. After that, you can move, delete, and re-create keys at time 0.

Turning Animate on has the following effect:

- The Animate button and the border of the active viewport turn red to indicate you are in animation mode.
- Keys are created whenever you transform an object or change an animatable parameter.
- The time slider sets the time where keys are created.

To begin animating an object:

1. Click Animate to turn it on.

2. Drag the time slider to a time other than 0.

3. Do one of the following:

- Transform an object.
- Change an animatable parameter.

For example, if you have a cylinder that has not been animated yet, it has no keys. If you turn Animate on, and at frame 20 you rotate the cylinder 90 degrees about its *Y* axis, Rotate keys are created at frames 0 and 20. The key at frame 0 stores the original orientation of the cylinder, while the key at frame 20 stores the animated rotation of 90 degrees. When you play the animation, the cylinder rotates 90 degrees about its *Y* axis over 20 frames.

Modeling Without Animating

Just as you can animate at any time by turning Animate on, you can also model at any time in your animation without creating animation keys.

The results of changing an object or any other parameters with Animate off varies according to whether or not the object or parameters have been animated yet.

- If you create a new object, or change an object parameter that has not been animated yet, you can work at any time with Animate off. The changes you make are constant through the entire animation.

  For example, you might animate an object bouncing around your scene and then decide to create pads for the object to land on. To do that, you drag the time slider to a time when the bouncing object hits the ground, and make sure Animate is off before you proceed. You can then create a pad under the bouncing object and repeat the process at the next time where it hits the ground. Because Animate was off, it does not matter at what time the pad objects were created. They remain inanimate through the entire animation.

- If you change an object or parameter that is already animated, while Animate is off, the amount of change is applied equally across all the animation keys.

  For example, you might animate a sphere's radius to be 15 at frame 0, 30 at frame 10, and 50 at frame 20. If you drag the time slider to frame 10, turn Animate off, and increase the sphere's radius from 30 to 40, the change in the radius is applied to the other two keys as well. Because you increased the radius by 10 units with Animate off, all radius keys are increased by 10 units. The sphere's radius is now 25 at frame 0, 40 at frame 10, and 60 at frame 20.

  If Animate had been on when you changed the radius, it would have been an animated change applied only to the key at frame 10.

Identifying What Can Be Animated

Because most parameters in 3D Studio MAX can be animated, the easiest way to find out if something can be animated is to just do it. Usually, if you want to animate a parameter, it can be animated.

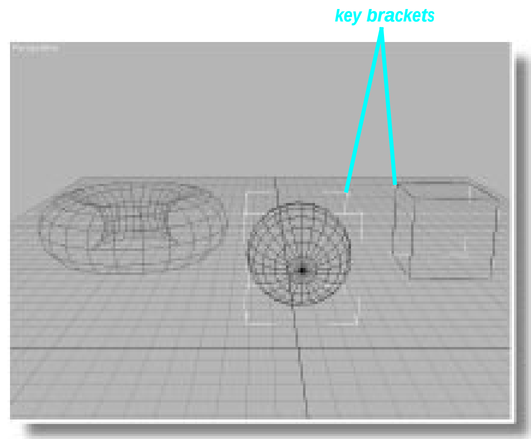
Sometimes you need to know in advance if you can animate a parameter. If so, you can use Track View. The Track View Hierarchy list displays every parameter that can be animated. Any item in the list displayed with a green triangle icon can be animated. See chapter 25, "Basic Track View Use," and chapter 27, "Working with Controllers."

Viewing and Copying Transform Keys

3D Studio MAX displays white brackets around objects that have transform keys at the current time. These *key brackets* only appear in viewports using the wireframe rendering method.

Key brackets are only displayed for transform keys. Use the Track View to view all key types.

For example, suppose you animated a sphere by moving it at frame 20, and scaling and rotating it at frame 50. When you drag the time slider, white brackets appear around the sphere at frames 50, 20, and 0.



Identifying objects with transform animation keys

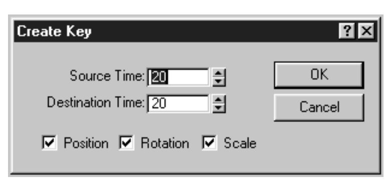
Controlling Key Bracket Display

You can control the display of key brackets using options in the Animation panel of the Preference Settings dialog.

Animation panel of the Preference Settings dialog

Creating Transform Keys with the Time Slider

You can use the time slider to create transform keys by copying transform values from one time to another. To specify the type of key to create and the source and destination time for the key values, right-click the time slider to display the Create Key dialog.



Create Key dialog

You set parameters in the Create Key dialog:

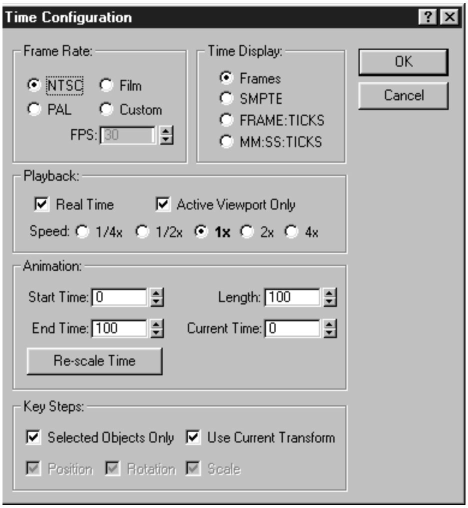
Source Time—Specifies the time from which transform values will be copied.

Destination Time—Specifies the time where the key will be created.

Position, Rotation, Scale—Determine which transform key values will be copied to the destination time.

When you click OK, new keys for the specified transforms are created at the destination time, using values from the source time. Keys do not have to exist at the source frame, because the interpolated values at the frame are used.

Controlling Time



Animation is created by changing your scene over time. You can exercise great control over time in 3D Studio MAX as follows:

- How time is measured and displayed.
- How much time is in your active work segment.
- How much time is covered by each rendered frame of your animation.

Other issues in the following topics describe how to move through time and how to view animation in the viewports.

Choosing the Time Display Format

When you start 3D Studio MAX, the default time display is in frames, but you can use alternative time-display formats. For example, you might want to see time in seconds and minutes.

You can specify four different time-display formats on the Time Configuration dialog under Time Display. When you change the time display format, you not only change the way that time is shown in all parts of 3DS MAX, but you also change the method with which you access time.

You can use these time display formats:

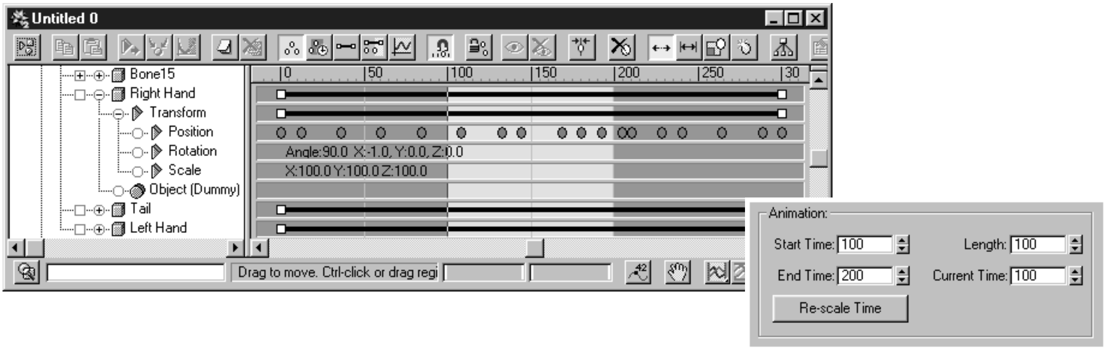
Frames—Displays time in whole frames. This is the default display mode for 3DS MAX. The amount of time covered by a single frame depends on your choice for the current frame rate. For example, in NTSC video each frame represents 1/30th of a second.

SMPTE—Displays time using the *Society of Motion Picture and Television Engineers* format. It is the standard time-display format for most professional animation work. From left to right, the SMPTE format displays minutes, seconds, and frames, delineated by colons. For example, 2:16:14 represents 2 minutes, 16 seconds, and 14 frames.

FRAME:TICKS—Displays time using frames and the 3DS MAX internal time increment called *ticks*. There are 4800 ticks per second, so you can actually access time down to 1/4800 of a second.

MM:SS:TICKS—Displays time in minutes (MM), seconds (SS), and ticks, delineated by colons. For example 2:16:2240 represents 2 minutes, 16 seconds, and 2240 ticks.

Setting Time Segments



The active time segment specifies a block of working time. You might think of it as a window in time that you use to focus on a specific part of your animation.

Specifying an Active Time Segment

You specify the active time segment by setting the Start Time and End Time for the segment on the Time Configuration dialog.

You can change the active time segment whenever you want without affecting the keys you've already created. For example, if you have keys scattered over a range of 1000 frames, you can narrow your active time segment to work on only frames 150–300.

Changing the active time segment has the following effect:

- Restricts the range of time you can access using the time slider.
- Restricts the range of time displayed when using the animation playback buttons.

The default setting for the active time segment runs from frames 0 to 100, but you can set it to any range.

Rescaling the Active Time Segment

You use the Rescale Time button on the Time Configuration dialog to change your entire animation based on the active time segment.

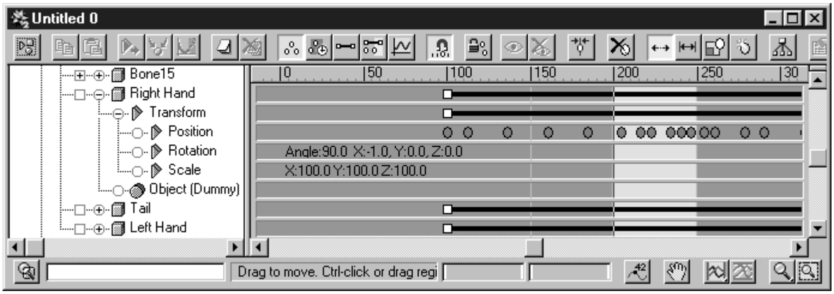
Use Rescale Time as follows:

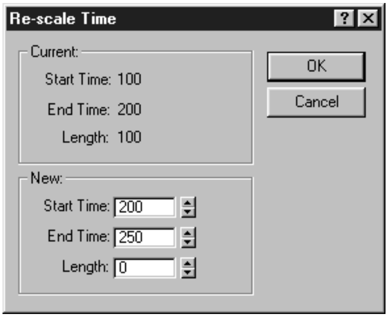
- Scale all animation in the active time segment to fit within a new time range.
- Move the entire animation to a new time.

You click Rescale Time and then enter new Start Time and End Time values on the Rescale Time dialog. When you click OK all the animation in the active time segment is moved and scaled to fit the new Start Time and End Time settings. Any animation outside the active time segment is moved to match the new active time segment boundaries.

For example, imagine you have an animation from frame 0 to frame 300 and an active time segment starting at frame 100 and ending at frame 200. Click Rescale Time and set the new Start Time to frame 200 and the new End Time to frame 250. Clicking OK gives you the following result:

- Animation in the original active time segment moves forward 100 frames and shrinks to a length of 50 frames. The new active time segment is from frames 200 to 250.

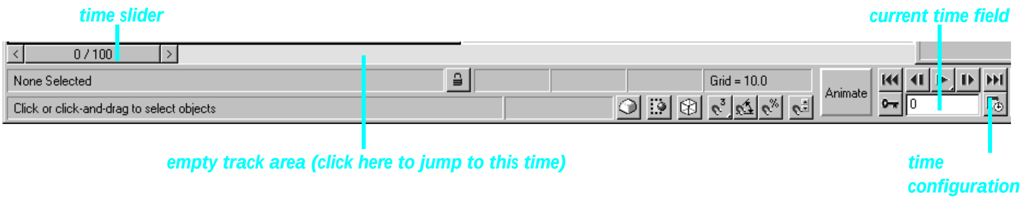




- Animation in the frames preceding the original active time segment moves forward 100 frames to connect to the start of the new active time segment.
- Animation in the 100 frames after the original active time segment moves forward 50 frames to connect to the end of the new active time segment.

The figures show the active segment and animation keys before and after rescaling.

Moving Through Time



You can move to any time in your active time segment by using either the time slider, or the Current Frame field in the time controls area. You can also move through time using the playback control buttons.

Using the Time Slider

The time slider shows you the current time, and lets you move to any time in your active time segment.

To change the current time using the time slider, do one of the following:

- Drag the time slider.
- Click in the empty track to either side of the time slider.
- Click the increment arrows at either end of the Time Slider.

When you click in the slider track, the time slider jumps to the time where you clicked. This is a faster method of moving through time than dragging the time slider.

The time slider displays the current time, followed by a slash (/), followed by the total time in the active time segment. For example **25/100** means frame 25 of 100 frames. The current time also appears in the current time field. If your scene has been animated, it's played back in all viewports as you drag the time slider.

Moving to an Exact Time

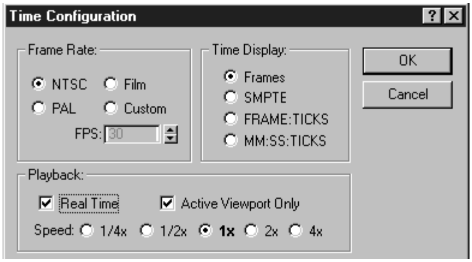
The Current Time field always displays the current time. You can also enter time value and press ENTER to move to that time.

Using the Time Control Buttons

You use the Time Control buttons to move forward and backward in time and to play your animation in one or more viewports. The Time Control buttons include four buttons for moving through time and a center flyout for controlling animation playback.

They work like VCR controls that you use to move through frames and to start and stop animation playback.

Choosing a Frame Rate and Playback Speed



The *frame rate* of an animation is expressed in frames per second (FPS). This is the number of frames 3D Studio MAX displays and renders for every second of real time.

Because 3D Studio MAX stores your animation keys using real time, measured at 1/4800 of a second, you can change the frame rate for your animation at any time without affecting your animation timing.

For example, if you create three seconds of animation using the NTSC video frame rate of 30 FPS, you will have a 90-frame animation. If you later discover you need to output to PAL video, at 25 frames per second, you can switch to the PAL frame rate, and your animation is now set to 75 frames of output. No change in the timing of your animation has occurred. Only the number of frames that 3DS MAX will display and render has changed.

Setting the Frame Rate

You use the settings in the Frame Rate area of the Time Configuration dialog to switch back and forth between frame rates at any time.

NTSC—U.S. and Japanese video standard of 30 frames per second.

PAL—European video standard of 25 frames per second.

Film—Movie standard of 24 frames per second.

Custom—Frame rate set in the FPS parameter.

Configuring Animation Playback

You use settings in the Playback area of the Time Configuration dialog to specify the playback speed, and the number of viewports that play the animation.

Real Time—Animation plays at the selected playback speed—skipping frames, if necessary, to maintain the correct speed. The different playback speeds are useful when using the Motion Capture utility.

Active Viewport Only—Animation plays only in the active viewport.

Speed—Select one of these options to multiply the frame rate by the selected speed.

Viewport Playback Speed

The ability of 3DS MAX to play your animation at a specified rate depends on many things, including the complexity of the scene, the number of objects moving in the scene, the geometry display mode, and so on. The worst case is a camera move in shaded mode, in which the viewport is filled with detailed geometry. In such cases, it's best to simplify the viewport display, using either wireframe display or, in extreme cases, box display mode.

Naturally, it takes more computing power to display your animation in four viewports, and playback smoothness is reduced. When Active Viewport Only is selected, you can switch active viewports during playback either by clicking the label of an inactive viewport, or by right-clicking in an inactive viewport.

Chapter

23

Building Hierarchies

One of the most useful tools in producing computer animation is the ability to link objects together to form a hierarchy. By linking one object to another you create a parent-child relationship. Transformations applied to the parent are also transmitted to the child. By linking more objects to both parent and child objects you can create complex hierarchies.

Linking objects together and building complex hierarchies have the following common uses:

- Creating complex motions
- Simulating jointed structures
- Providing the structure for inverse kinematics (IK)

This chapter explains how to plan and build linked hierarchies. It also demonstrates how the standard hierarchical linking commands are used to set up complex motion and to create jointed structures.

Chapter 24, "Using Inverse Kinematics," explains how linked structures can be animated with IK.

See the Online Reference for details about linking and hierarchies.

Understanding Hierarchies

Object linking and building hierarchical structures are common techniques in computer modeling and animation. You find the commands to build and manipulate hierarchies in the following places of the 3D Studio MAX interface:

-  The Link and Unlink buttons on the left side of the toolbar are used to make and break links between objects in your scene.
- The Bones system on the Create panel is used to create a hierarchy of bones, or to automatically bone an existing hierarchy.
- The Hierarchy panel contains commands to control how links behave.

Defining Terms to Describe Hierarchies

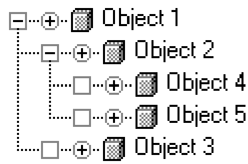
You can link objects together so that one object is superior and controls the transforms of one or more subordinate objects. The linkage is unidirectional in that superior objects control subordinates but subordinates have no effect on their superiors. This structure is called a *hierarchy*.

3D Studio MAX uses a family tree analogy to describe the relationship between objects linked together in a hierarchy.

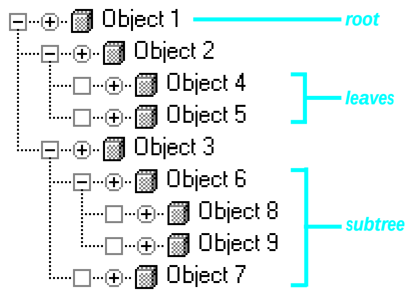
Parent—Object that controls one or more children. A parent object is often controlled by another superior parent object. In the figure, objects 1 and 2 are parent objects.

Child—Object controlled by its parent. A child object can also be a parent to other children. In the figure, objects 2 and 3 are children of object 1. Objects 4 and 5 are children of object 2.

Ancestors—Parent and all of the parent's parents of a child object. In the figure, objects 1 and 2 are ancestors of object 4.



Parent-child relationship in a hierarchy



Example of a hierarchical structure.

Descendants—Children and all of the children's children of a parent object. In the figure, all of the objects are descendants of object 1.

Hierarchy—Collection of all parents and children linked together in a single structure.

Root—Single parent object that is superior to all other objects in the hierarchy. All other objects are descendants of the root object. In the figure, Object 1 is the root.

Subtree—All the descendants of a selected parent. In the figure, objects 6, 8, 9, and 7 represent the subtree under object 3.

Branch—Path through the hierarchy from a parent to a single descendant. In the figure, objects 1, 3, 6, and 8 represent a branch from the root to a leaf object.

Leaf—Child object that has no children. The lowest object in a branch. In the figure, objects 4, 5, 7, 8, and 9 are leaf objects.

Link—Connection between a parent and its child. Links transmit position, rotation, and scale information from parent to child.

Pivot—Defines the local center and coordinate system for each object. You can think of links as connecting the pivot of a child object to the pivot of its parent.

Common Uses for Linking

- Link a large collection of objects to a single parent so they can be easily animated and transformed by moving, rotating, or scaling the parent.
- Link the target of a camera or light to another object so it tracks the object through the scene.
- Link objects to dummy objects to create complex motions by combining multiple simple motions.
- Link objects to simulate jointed structures to use with inverse kinematics.

Hierarchy of Unlinked Objects

All objects in 3D Studio MAX are part of a hierarchy. 3DS MAX uses an imaginary object named *World* to act as the root of everything in your scene. Therefore, if an object is not linked to some other object in the scene, it is considered a *child of the World*.

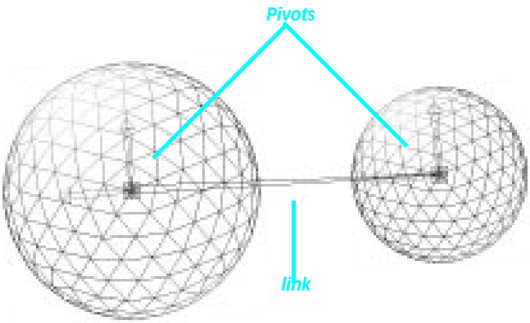
The World object and children of the World have special functions when you work with inverse kinematics and Track View. See chapter 24, "Using Inverse Kinematics," and chapter 25, "Basic Track View Use."

Manipulating Hierarchies with Forward Kinematics

The default method of manipulating a hierarchy in 3D Studio MAX uses a technique called *forward kinematics*. The basic principles employed by this technique are:

- Hierarchical linking from parent to child
- Placement of pivot points to define the connecting joint between linked objects
- Inheritance of position, rotation, and scale transforms from parent to child

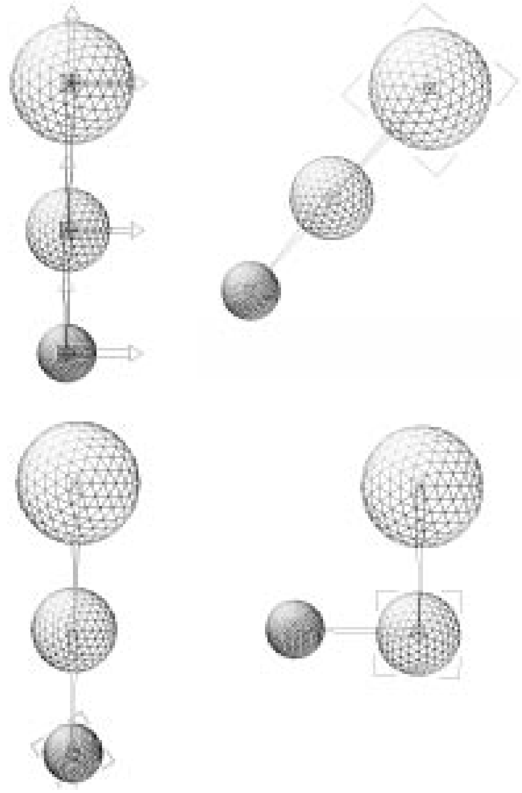
This topic examines links, pivots and inheritance as they apply to forward kinematics.



How Links and Pivots Work

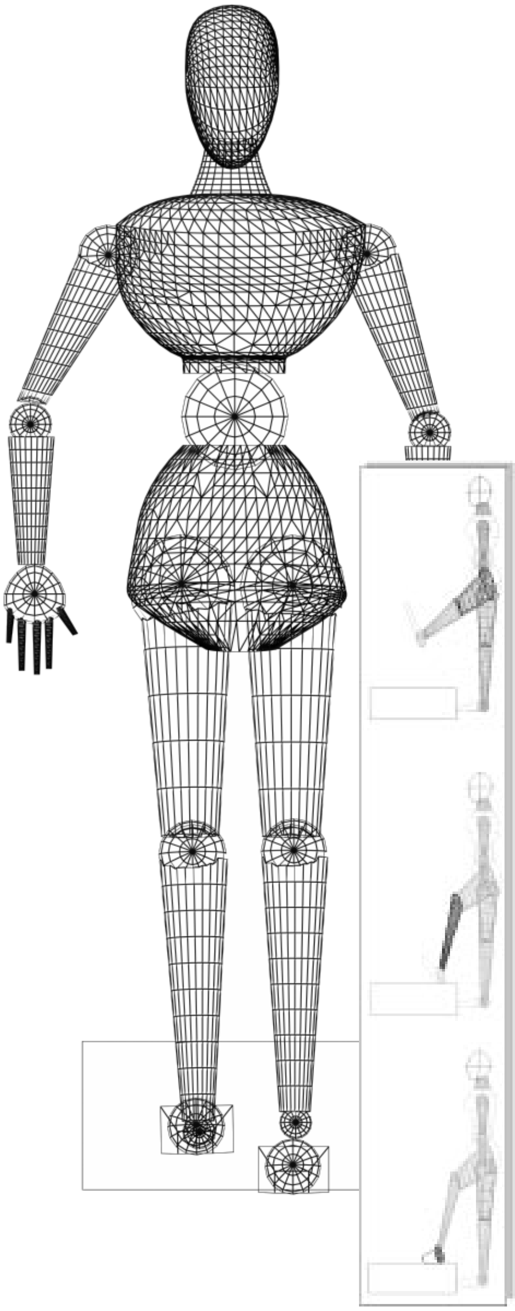
Once two objects are linked together, the child object maintains its position, rotation, and scale transforms relative to its parent object. These transforms are measured from the pivot of the parent to the pivot of the child.

For example, consider the two spheres in the following figure. The larger sphere is the parent of the smaller. The pivots and link between the spheres are indicated to show how the link works. The link extends from the pivot of the parent and connects to the pivot of the child. You can think of the child's pivot as being the joint between the parent and child.



Links act as a one-way conduit to transmit the transforms of a parent object to its child object. If you move, rotate, or scale the parent, the child is moved, rotated, or scaled by the same amount. Because hierarchical links are one-way, moving, rotating, or scaling the child has no effect on its parent.

The end result is that transforms applied to a child object are applied in addition to any transforms inherited from the child's parent.



Manipulating the Hierarchy

A child object inherits the transforms of its parent, and the parent inherits the transforms of its ancestors all the way up the hierarchy to the root object. Because forward kinematics employs this method of inheritance, you must position and animate your hierarchies using a top-down method.

Consider the linked mannequin in the following figure. If you want to position the mannequin's left foot to rest on top of the box beside it, you perform the following steps:

To position the mannequin:

1. Move the mannequin closer to the box by selecting and moving the abdomen.
2. Rotate the left thigh so the entire leg is above the box.
3. Rotate the left shin so the foot is near the top of the box.
4. Rotate the foot so it is parallel with the top.
5. Repeat steps 1 through 4 until the foot is properly placed.

You always start transforming objects at the highest-level parent affected by the motion and work your way down the hierarchy to the last child.

You have considerable control over the exact placement of every object in the hierarchy using forward kinematics. However, the process can become tedious with large and complex hierarchies.

Linking Strategy

Before you begin linking any but the simplest hierarchy you should take a few minutes to plan your linking strategy. Your choices for the root of the hierarchy and how the branches grow out to the leaf objects will have important effects on the usability of your model.

The strategy behind linking objects into a hierarchy can be reduced to two main principles:

- The hierarchy follows a logical progression from parent to child.
- Parent objects move less than their descendants.

Within these two principles you have almost unlimited flexibility as to how you link your objects. If you think about how you intend to use the hierarchy, and link it with that use in mind, you will rarely have a problem.

Progression from Parent to Child

Progression from parent to child means the links do not erratically jump from object to object. If two objects touch each other they should probably be linked as parent and child. There is nothing to prevent you from linking a body in the order of: Thigh->Foot->Shin->Waist You would probably regret such a linking strategy later. The effort to figure out how to transform objects linked in such a strange way would be quite difficult.

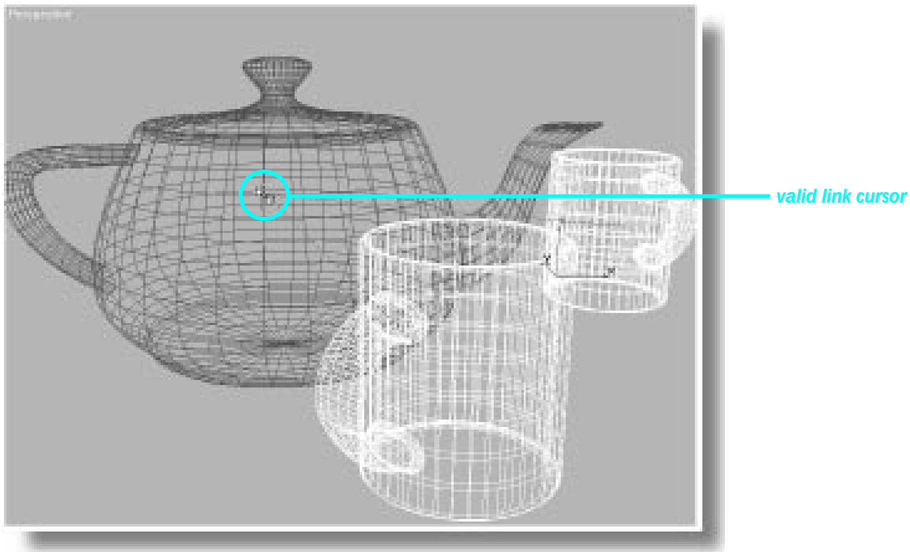
Parents Move Less Than Descendants

Because of the way transforms are inherited from parent to child, small adjustments to a parent object might require you to adjust all of its descendants. The typical approach to linking is to choose as your root object the object that moves the least. Objects close to the root should move very little, and leaf objects should move the most.

This is especially true when you are linking jointed structures like bodies or machinery, or intend to use the hierarchy with inverse kinematics. See chapter 24, “Using Inverse Kinematics.”

An exception to this rule occurs when you are using the root object as a handle. All of the descendants of the root are just along for the ride. Consider a tray full of objects traveling on a conveyor belt. All the objects should be children of the tray even though the tray moves much more than any of the other objects.

Linking and Unlinking Objects



Use Link and Unlink on the toolbar to make and remove links between objects.

Linking Objects

The general process of creating links is to build the hierarchy from child to parent. You click Link on the toolbar, select one or more objects as children, and then drag the link cursor from the selection to a single parent object. The selected objects become children of the parent object.

Once objects are linked, any transformations applied to the parent are also applied to its children. For example, if you scale the parent to 150%, the size of its children and the distance between the children and the parent are also scaled by 150%.

Unlinking Objects

Click Unlink to remove the link from selected objects to their parents. Any children of the selected object are unaffected.

Tip: You can quickly unlink an entire hierarchy by double-clicking the root object to select the object and all of its children. Then click Unlink.

Linking Animated Objects

You should establish links before you begin animating objects. The linking and unlinking of objects cannot be animated, and the links exist throughout an entire animation. See “Animating Hierarchies.”

Building Hierarchies with Bones

You can quickly create a hierarchical system of bones by clicking Bones on the Systems category of the Create panel. Bones are nonrendering objects that define a linked hierarchy without any associated geometry.

Some advantages of using bones include:

- Bones define a hierarchy that you can use as a guide for building a linked model.
- Auto Boning adds a bone hierarchy to a linked model automatically.
- Bones can define a structure used by skin modifiers for animating one-piece models. The Character Studio modifier Physique is an example of this type of modifier.
- Bones are the only objects that can use the IK controller for interactive IK solutions. See chapter 24, “Using Inverse Kinematics.”

Creating Bones

You create bones by clicking the Systems category on the Create panel. Click Bones and do the following:

- Your first click in a viewport defines the root bone of the hierarchy.
- Each subsequent click defines a new bone as a child of the previous bone. The result of multiple clicks is a single chain of bones.
- Right-click to exit bone creation.

To create a branching hierarchy, such as legs branching from a pelvis, do the following:

- Create a chain of bones and right-click to exit bone creation.
- Click Bones again, then click the bone where you want to begin branching. The new chain of bones branches from the bone you click.

Automatically Boning Objects

You use Auto Boning to automatically assign bones to objects in an existing hierarchy. Auto Boning is handy when you want to convert a regular linked hierarchy to a bone structure using the IK controller.

You Auto Bone a hierarchy by clicking Pick Root on the Bone Parameters rollout, and then selecting an object as the root object of your bone structure. Bones are created for that object, plus all its descendants.

You can set options on the Auto Boning area to control the following properties:

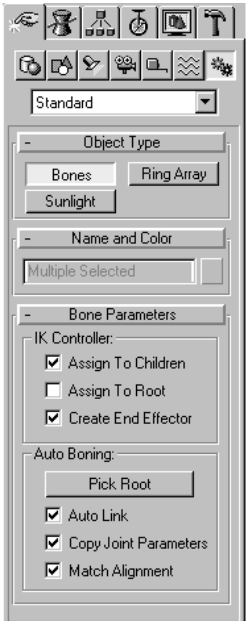
- Automatic linking to link the selected hierarchy to the newly created bone structure. In most cases, you will want this option enabled.
- Copying of any IK joint parameters from the selected hierarchy to the bone structure.
- Matching the local coordinate systems of the bone structure to the selected hierarchy.

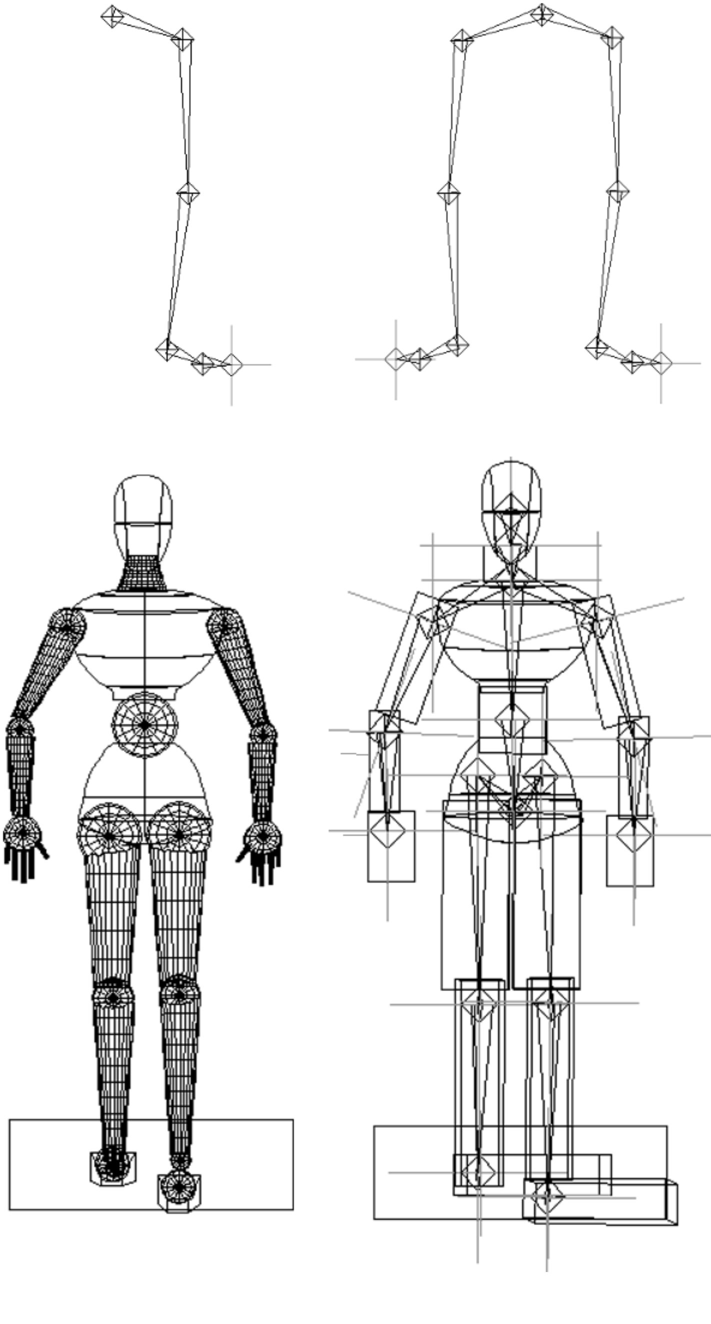
Assigning IK Controllers to Bones

You set options on the IK Controller area to control the assignment of controllers to bones. as follows:

- Clear Assign to Children if you want to create a bone structure with standard controllers for use with forward kinematics or Applied IK.
- Select Assign to Children if you want to create a bone structure using the IK controller.

See the Online Reference and chapter 24, “Using Inverse Kinematics,” for information about IK.


Standard
Object Type
Bones
Ring Array
Sunlight
Name and Color
Multiple Selected
Bone Parameters
IK Controller:
Assign To Children
Assign To Root
Create End Effector
Auto Boning:
Pick Root
Auto Link
Copy Joint Parameters
Match Alignment



Selecting and Deselecting Hierarchies

You can quickly select or deselect an object plus all of its descendants.

Double-clicking

You can double-click an object to select and deselect that object and all of its descendants.

- Double-click an Object in a viewport.
- Double-click an Object icon in the Track View Hierarchy list.

Using the Shortcut Menu

Right-click any selected object, and then choose either Select Children or Deselect Children to add or remove all descendants of the selected objects from the selection.

Using the Select Objects Dialog

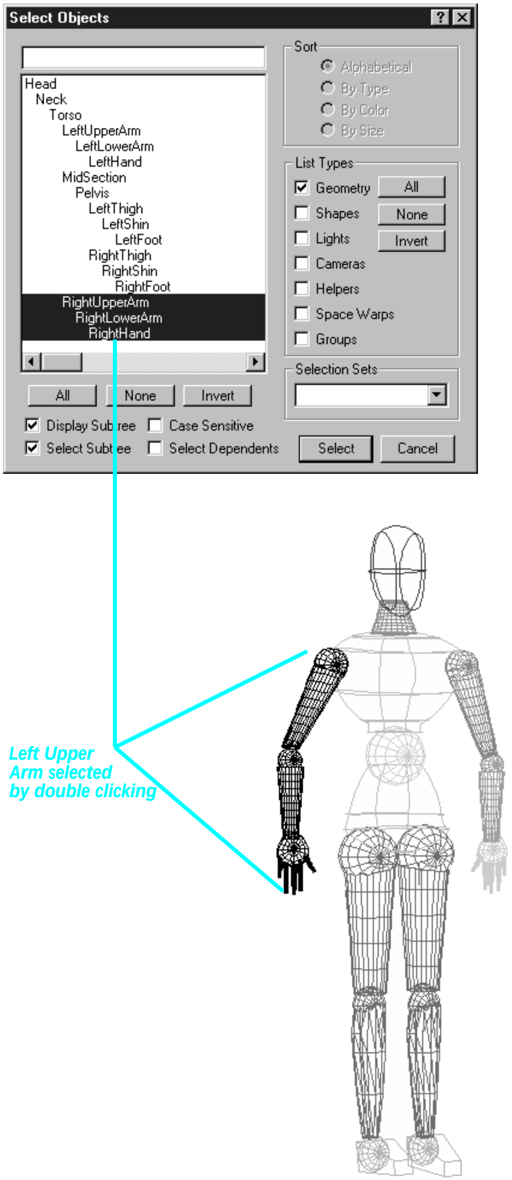
Enable Select Subtree to select hierarchies in the Select Objects dialog. When Select Subtree is enabled, click the name of an object to select the object and all its descendants.

Navigating a Hierarchy

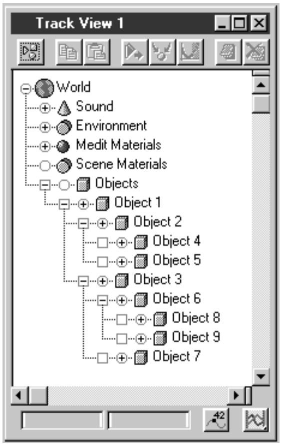
Once you have selected one or more objects in a hierarchy, you can navigate up and down the hierarchical tree using the PAGE UP and PAGE DOWN keys.

- PAGE UP deselects the object and selects the object's parent.
- PAGE DOWN deselects the object and selects all of its children.

These navigation commands are particularly useful when setting joint parameters for inverse kinematics.



Viewing Hierarchy Structure



Displaying the hierarchy in the Track View list window

You can use either of two methods to view the tree structure of your hierarchies. One method is the Display Subtree option in the Select Objects dialog. The other is the Hierarchy list in Track View. Both of these methods display your hierarchies as indented lists expressing the parent-child relationship from the root object to the leaf objects.

Using the Select Objects Dialog

The Select Object dialog appears whenever you use a by-name selection method, such as choosing Edit > Select By > Name from the menu or by clicking Select by Name on the toolbar.

Selecting Display Subtree on the dialog lists objects by indenting children to the right of their parent.

Using Track View

Track View displays all objects using indentation to express hierarchy. Child objects are displayed indented and below their parent. An added advantage of Track View is that you can control the view by collapsing and expanding branches of the hierarchy.

You expand and collapse branches of the hierarchy by clicking the square icon to the left of the object icon.

- A square icon with a plus indicates a collapsed branch under that object. Clicking it expands the branch.
- A square icon with a minus indicates the branch under that object is expanded. Clicking it collapses the branch.
- An empty square icon means no branch under that object. Clicking it does nothing.

See chapter 25, “Basic Track View Use,” for information about the Track View Hierarchy list.

Using Dummies

The primary use of dummy helper objects is to assist in creating complex motions and building complex hierarchies. Because dummies are invisible when rendered, they are an excellent choice for offset joints, connectors between objects, and handles for complex hierarchies.

Using a Dummy to Control Motion

Breaking complex motions into simple components often makes it easier to go back and edit your animations.

Consider a bouncing ball that moves along a path. You could animate the ball by positioning it on many frames, but it would be very difficult for you to go back and adjust the height of the bounce or the path of the ball. You have to edit the motion of the ball on many frames to make even a simple change.

Using a dummy object solves this problem by breaking the motion into simple components. One component is the up and down bounce of the ball. The other is movement on the path.

To create a complex bounce motion using a dummy object:

1. Start with a sphere, then create a dummy object below the sphere, and link the sphere as a child of the dummy.
2. Animate the sphere bouncing up and down above the dummy.
3. Animate the dummy moving.

The sphere bounces on top of the dummy object as the dummy moves around the scene. You can easily change the height and speed of the bounce by changing the sphere animation. You can change the path through the scene by changing the dummy animation.

Using a Dummy as a Handle

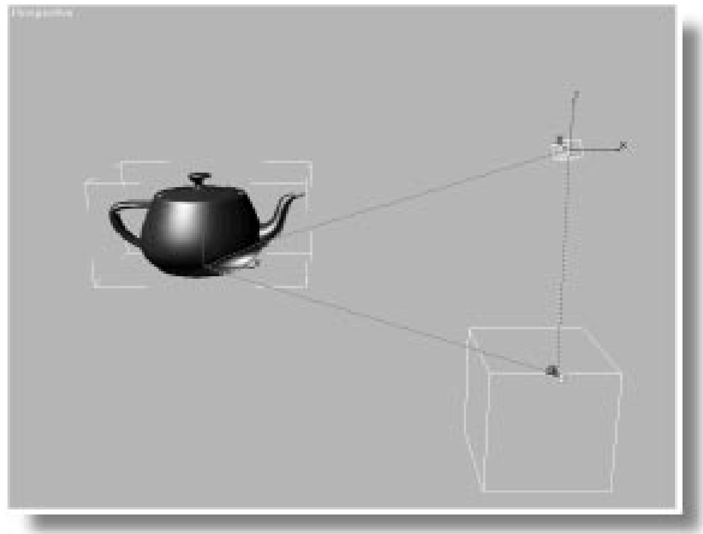
You might want to move and animate a selection of objects individually but also have the ability to transform them as a single object.

A good example of this is a camera on a tripod. You want to adjust both the camera and its target individually but also want to move them together as a single unit.

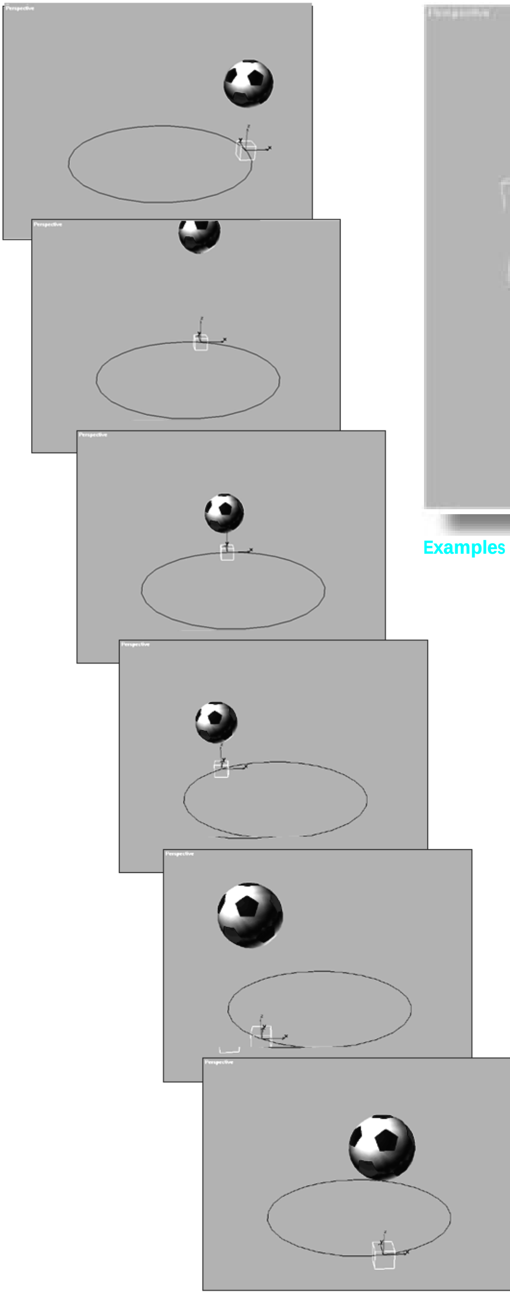
To create a camera tripod:

- Create a dummy object below a target camera and link the camera and target as children of the dummy object.

The camera and the target follow the dummy object. You can quickly position the camera by placing the dummy object and compose your view by adjusting the camera and its target.



Examples of a camera and tripod linkage



Sample frames of bouncing ball

Adjusting Pivots

You can think of an object's pivot point as representing its local center and local coordinate system. The pivot point of an object is used for a number of purposes:

- As the center for rotation and scaling when the Pivot Point transform center is selected.
- As the default location of a modifier center.
- As the transform offset for linked children.
- As the joint location for IK.

You can adjust pivot points by clicking Pivot on the Hierarchy panel, and then using features on the Adjust Pivot rollout.

The functions under the Adjust Pivot rollout cannot be animated. Adjusting an object's pivot on any frame changes it for the entire animation.

Affecting Pivot Only

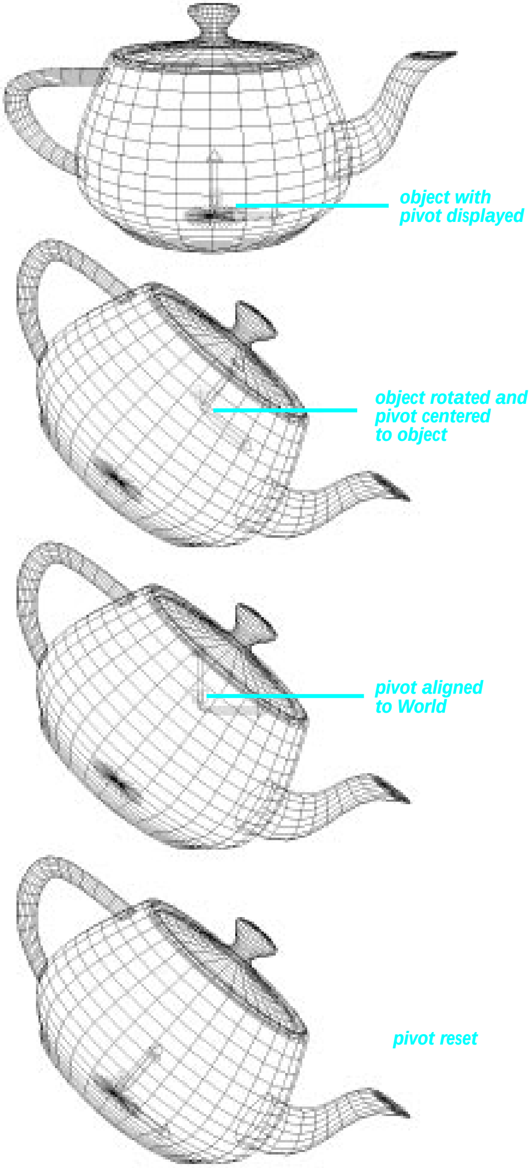
When Affect Pivot Only is on, move and rotate transforms are only applied to the pivot of selected objects.

- Moving or rotating the pivot does not affect the object or its children.
- Scaling the pivot scales the object from the pivot center, but its children are unaffected.

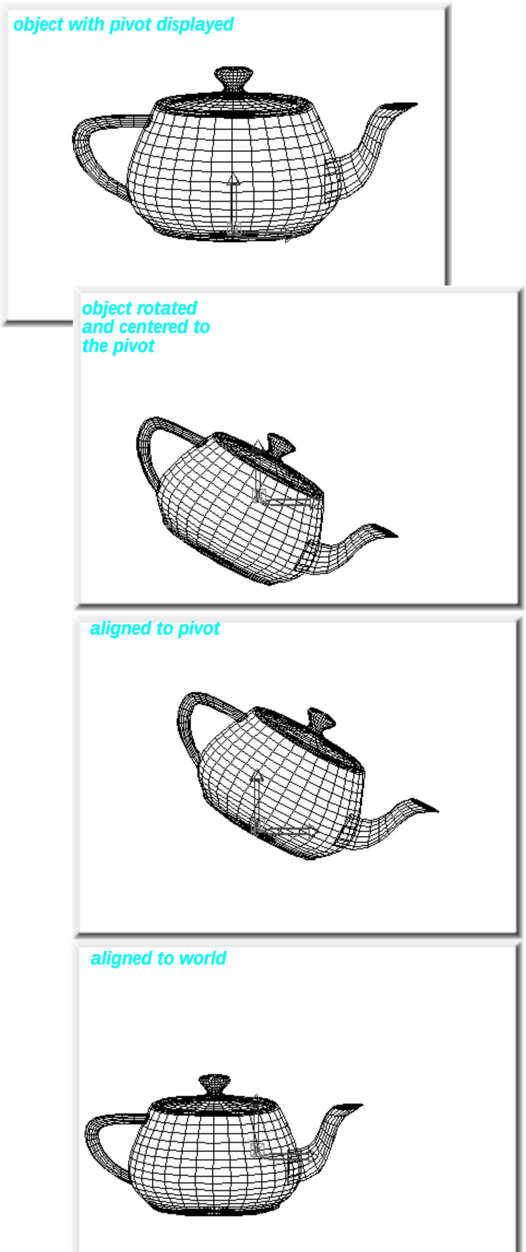
Affecting Object Only

When Affect Object Only is on, transforms are only applied to selected objects. Pivots are not affected.

Moving, rotating, or scaling the object does not affect the pivot or its children.



Examples of Affect Pivot only



Affecting Object only

Affecting Hierarchy Only

When Affect Hierarchy Only is on, Rotate and Scale transforms are only applied to the links between objects and their children.

- Scaling or rotating an object affects the link offsets of all its descendents without affecting the geometry of the object or its descendents. The descendents shift position because of the scaled or rotated links.

Use this technique to adjust the offset relationship between linked objects and for adjusting bones to match geometry.

Aligning Pivots

Buttons on the Alignment area of the Adjust Pivot rollout change names based on the state of Affect Object Only and Affect Pivot Only. Alignment is disabled when Affect Hierarchy Only is active.

Center to Object/Pivot—Moves the object, or pivot, so the pivot is at the center of the object.

Align to Object/Pivot—Rotates the object, or pivot, to align the pivot with the object's original local coordinate system.

Align to World—Rotates the object, or pivot, to align with the world coordinate system.

Resetting the Pivot

Click Reset Pivot to return the pivot point of a selected object to the position and orientation it held when the object was first created.

Reset Pivot has no effect on the object or its children. The state of the Affect Pivot Only and Affect Object Only is ignored.

Adjusting Object Transforms

You use the features on the Adjust Transform rollout to transform objects after they have been linked without transforming descendents, and to reset an object's transform.

Transforming Parent Objects

You might discover, after linking a number of objects, that you need to move, rotate, or scale a parent object but you do not want to affect the object's descendents. You can transform a parent object without affecting its descendents by clicking Don't Affect Children on the Adjust Transform rollout of the Hierarchy panel.

Resetting an Object's Orientation

Click the Transform button in the Reset area to rotate an object's pivot to match it's parent's local coordinate system. Descendents of the object are not affected.

Resetting an Object's Scale

Click the Scale button in the Reset area to set the current scale value as the selected object's base scale value. All following scale transforms are then applied using the base scale value as an absolute local scale of 100%.

Consider a sphere with a radius of 20 units and a linked child object:

1. Uniform Scale the sphere to 200%. The sphere and its child become twice as big. Scale Transform Type-In reports an Absolute Local Scale of 200% and the object's creation parameters report a radius of 20 units. The true radius of the sphere is 200% of 20 units, or 40 units.

2. Select the sphere and click Reset Scale. The sphere and its child remain the same size. Here's what has happened:

- The 200% scale has been absorbed by the sphere as its original state. The sphere has a true radius of 40 units, Creation Parameters report a radius of 20 units, and absolute local scale is 100%.
- The sphere's child object accepts a local scale of 200% so it does not change in size.

Resetting the scale of an object can lead to confusion because the object's true size, absolute local scale, and creation parameters no longer match up.

Using the Reset XForm Utility

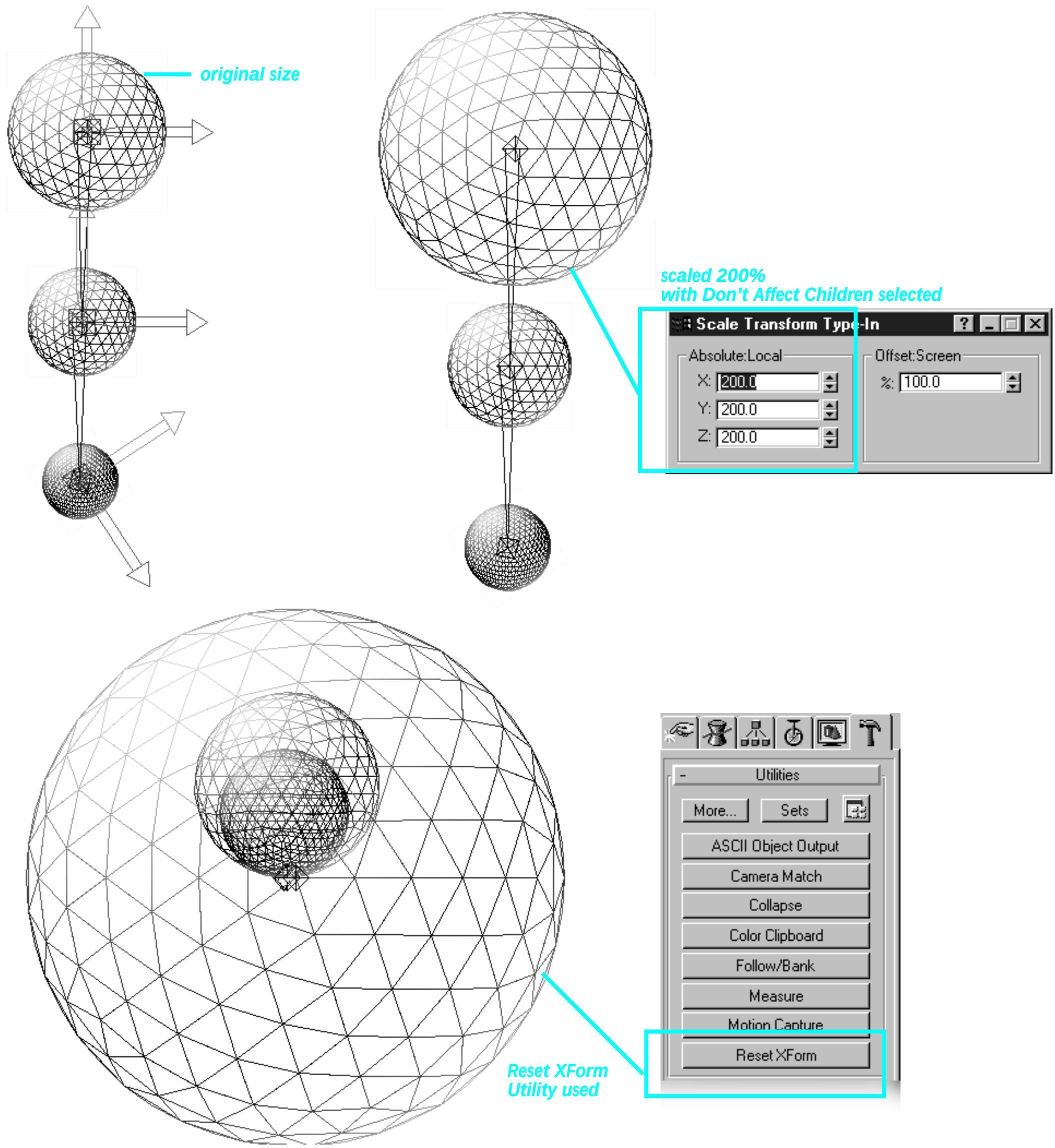
You can also reset the orientation and scale of an object by clicking Reset XForm on the Utility panel. Reset XForm takes the rotation and scale transforms of an object and places them in an XForm modifier on the modifier stack.

Consider the same sphere as before with a radius of 20 units and a linked child object:

1. Uniform Scale the sphere to 200%. The sphere and its child become twice as big. Scale Transform Type-In reports an Absolute Local Scale of 200% and Creation Parameters report a radius of 20 units. The true radius of the sphere is 200% of 20 units, or 40 units.

2. Select the sphere and click Reset XForm. The sphere remains the same size but its child reverts to its original size and position. Here's what has happened:

- The 200% scale has been placed in an XForm modifier on the sphere's modifier stack. The sphere has a true radius of 40 units, Creation Parameters report a radius of 20 units, and absolute local scale is 100%.
  - The sphere's child object now sees only the 100% local scale so it reverts to its original size and position.


original size
scaled 200%
with Don't Affect Children selected
Scale Transform Type-In
Absolute:Local
X: 200.0
Y: 200.0
Z: 200.0
Offset:Screen
%: 100.0
Utilities
More...
Sets
ASCII Object Output
Camera Match
Collapse
Color Clipboard
Follow/Bank
Measure
Motion Capture
Reset XForm
Reset XForm
Utility used

Locking Object Transforms

You can lock an object's ability of Move, Rotate, or Scale about any of its Local axes by selecting objects and then setting options on the Locks rollout of the Hierarchy panel.

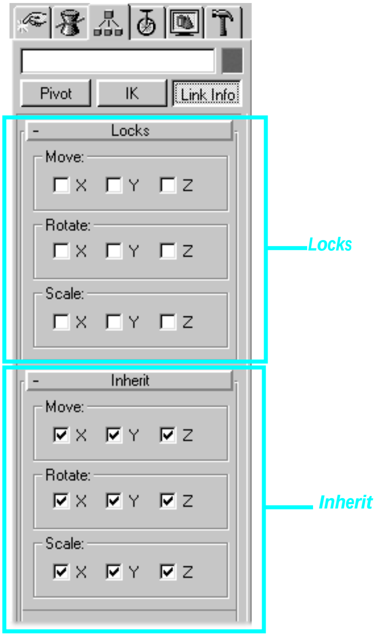
Enabling and disabling Local transform axes is also referred to as setting Degrees of Freedom (DoF) for an object. If an axis is enabled, an object is free to transform about that Local axis.

The Locks rollout contains three areas, one each for **Move**, **Rotate**, and **Scale**. Each area contains three options, one each for the *X*, *Y*, and *Z* Local axes of the selected objects.

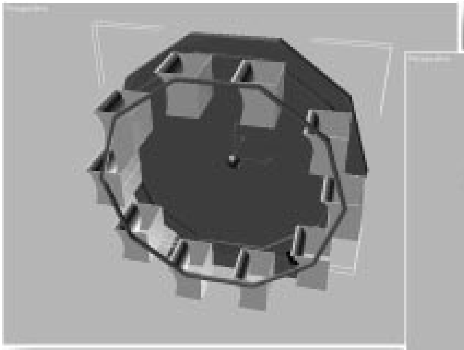
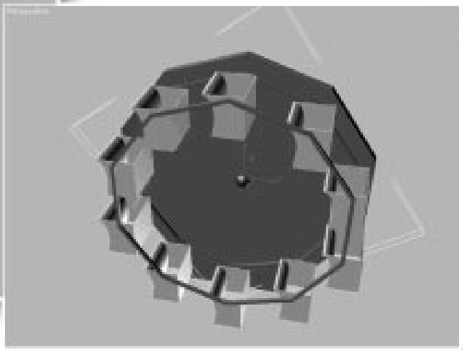
- When selected, the objects cannot be transformed about the selected Local axes when you directly use one of the transform tools.

  Objects can still be transformed by other means such as being a child of a transformed parent object or being part of an inverse kinematics chain.

- When clear, objects can be freely transformed about the cleared Local axes.



Changing Link Inheritance

To set an object's ability to inherit the Move, Rotate, and Scale transforms of its parent, you use the Inherit rollout of the Hierarchy panel. Links transmit transform information from a parent to a child. By default, a child inherits all of the transforms of its parent. You change the link settings to limit what transforms a child inherits.

The Inherit rollout is only available when you have selected a single object. When multiple objects are selected, the rollout is disabled.

Setting Link Inheritance Options

Set the Inherit options to release the link components between a selected object and its parent for any world axis of Move, Rotate, or Scale.

The options that appear when you expand the Inheritance rollout are determined by the transform controllers assigned to the selected object.

The Inherit rollout for objects using standard position, rotation, and scale controllers contains three areas, one area each for Move, Rotate, and Scale. Each area contains three options, one each for the X, Y, and Z world axes.

- When selected, transform information from the parent is passed on to the child for the selected World axes.
- When clear, transform information for the cleared axes is ignored by the child.

Imagine you are animating a ferris wheel that rotates on the world Y axis. The wheel is the parent and the cars are its children. You want the cars to ignore the Y-axis rotation of the wheel. Otherwise, the passengers would all fall out. You use Link Info to clear the Y-axis option on the Rotation area of the Inherit rollout for each car.

Using the Link Inheritance Utility

This utility works exactly the same as the Inherit rollout in the Link Info area on the Hierarchy panel, except that you can set link inheritance for multiple objects in a selection set.

For example, on a ferris wheel, you could select all of the cars and use the Link Inheritance utility to clear Y-axis rotation for all cars at once.

Animating Hierarchies

You animate hierarchies in much the same way you animate anything else in 3D Studio MAX. You click Animate, to turn it on, and apply transformations and modifiers. However, you need to be aware of a few special issues for animating hierarchies.

Animating a Child Object

A child is not constrained by its link to a parent. You can move, rotate, and scale children independent of their parents.

Animating a Parent Object

Only transforms are passed from parent to child. Animating a parent object using move, rotate, or scale animates the parent and the subtree attached to the parent.

Animating a parent's modifiers or creation parameters has no effect on its descendants.

Inverse Kinematics Animation

Another method for animating hierarchical structures is through the use of inverse kinematics. Using this method you animate child objects and pass the effects up the hierarchy to the child's ancestors. See chapter 25, "Using Inverse Kinematics."

Linking Objects After Animation

When you link an object to another, the link relationship between the child and its parent is determined on the frame when the link is made.

Imagine linking a sphere to an animated box. The figure shows a stationary sphere near a moving box.

- At frame 0 the box is beside the sphere.
- At frame 50 the box is 20 units away.

Linking the sphere to the box causes the sphere to move with the box. The distance between the sphere and the box depends on the frame when the link is made. Linking the sphere on different frames has the following effects:

- Link on frame 0, and the sphere stays next to the box as it moves.
- Link on frame 50, and the sphere stays a distance 20 units away from the box as it moves.

Unlinking Objects After Animation

When you unlink a child, its frame 0 transforms are taken from the transforms of its parent at the frame when the link is removed.

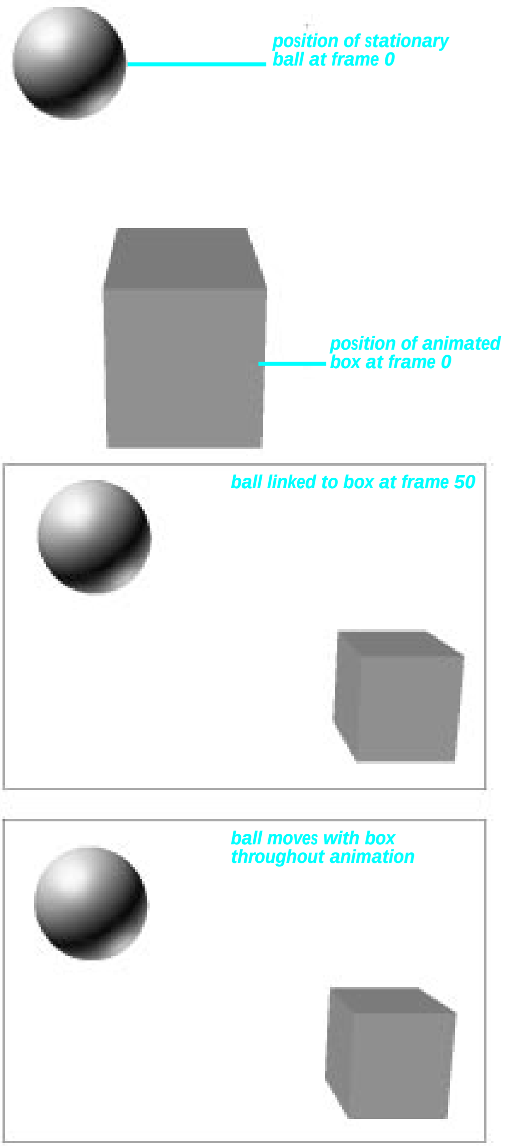
Imagine a sphere linked to a box moving around the face of a clock. The box starts at 12 o'clock and travels all the way around the face over 100 frames.The figure shows a box moving in a circle with a sphere linked above it.

If you unlink the sphere, it stops following the box. The position of the sphere depends on the frame when the link is removed. Unlinking the sphere on different frames has the following effects:

- Unlink on frame 0, and the sphere stops at 12 o'clock.
- Unlink on frame 25, and the sphere stops at 3 o'clock.
- Unlink on frame 75, and the sphere stops at 9 o'clock.

Animating Links

Links are permanent, regardless of the state of the Animate button. The next topic discusses how to simulate animated links using a link controller.



Animated box with unlinked sphere

Animating Links

You assign a Link controller to an object to animate links from one parent to another. You use a link controller instead of using the regular link and unlink buttons on the tool bar.

See chapter 27, “Working with Controllers,” to read about assigning controllers to objects.

An example of using a link controller is to pass a ball from one hand to another. Assume that at frame 0 the ball is in the first hand. The hands are animated to meet at frame 50 and then spread apart until frame 100.

To animate the links for the ball:

1. On the Motion command panel, assign a Link controller as the ball's Transform controller.
2. Set the Time Slider to frame 0, then on the Motion panel click Add Link, and click the hand holding the ball. The ball will now move along with the hand, as if it were linked to it.
3. Set the Time Slider to frame 50, where you want the second hand to pick up the ball, click Add Link, then click the second hand. From this frame on, it's as if the ball were linked to the second hand.

When you play back the animation, the ball travels with the first hand until frame 50, where you added the second link, then it travels with the second hand.

Adding and Deleting Links

You add and delete links on the Motion panel. Expand the Link Parameters rollout and click Add Link or Delete Link.

- Click Add Link then click the object that you want to link to as a parent. The frame at which you add the link is the frame at which control is passed. You can change the link frame with the Start Time parameter.
- Select the name of a parent object in the list and click Delete Link to remove the link.

Properties of the Link controller include:

- The Link controller respects the link inheritance settings applied to the child object.
- The object using a Link controller is not a true child object. It does not appear in the subtree of any linked parent objects.
- Objects with Link controllers do not participate in IK solutions.

Side Effects of the Link Controller

The Link controller works to keep a child object from jumping position at the time when the link changes from one parent to another parent.

Considering the previous example, the following should hold true:

- During frames 0-50 the ball remains constant relative to the first hand.
- During frames 50-100 the ball remains constant relative to the second hand.
- At frame 50—the time when link control changes—the ball does not jump.

If you change the animation of the second hand at frame 75, it affects the position of the hand relative to the ball at the time of the link (frame 50). This change in relative position affects the ball over all frames where it is linked to the second hand. Therefore, as you change the position of the hand at frame 75, the child's position will also change, possibly in a counter-intuitive way. However, when playing back the animation the above three rules will hold true.

Animating Attachment

You assign an Attachment controller to cause an object to hold a position on the surface of another object. The Attachment controller is not a hierarchical link, but it has the effect of "linking" an object to the surface of another object as follows:

- Attach a "source" object to the face of a "target" object so that the source object acts as if it's glued to the target object, no matter how the surface of the target object is deformed.
- Animate the Attachment parameters so that the source object moves over the surface of the target object.

Unlike hierarchical linking, which considers only object transforms, an object using an Attachment controller follows the deformations of another object based on that object's modifiers and space warp bindings.

See chapter 27, "Working with Controllers," to read about assigning controllers to objects.

Setting Attachment Parameters

You use features on the Attachment Parameters rollout on the Motion panel, to pick a target object and position the source object.

- Click Pick Object, then click the target object to perform the attachment.
- Click Set Position and click or drag on the surface of the target object to place the source object onto the surface.

  If you want to move the source object along the normal of the face—move it below or above the face— click Affect Object Only on the Hierarchy panel and move the object, using Local transform coordinates.
- Select Align to Surface if you want to align the world Z axis of the source object with the surface normal of the target object. Even with Align to Surface selected, you can always rotate the source object to orient it the way you want in relation to the target object.

  If you were to align trees on an uneven terrain, you would clear Align to Surface so that all of the trees grew upright, regardless of the angle of the terrain surface.

Animating Attachment Position

You can move to any frame and click Set Position to animate the source object moving across the surface of the target object. It is not necessary to turn on the Animate button, because you are working with an animation controller.

When you set positions for the source target on multiple frames, its attachment to the target object is only fixed at each keyframe. Frames between keys are interpolated and may not match the target surface.

Test your animation, and either adjust the values of the keys, or add intermediate keys to better match the target surface. Too many keys can result in jittery movement of the source object, while too few keys may result in the source object missing the surface of the target object over some frames.

Chapter

24

Using Inverse Kinematics

Inverse kinematics (IK) is a method of manipulating linked structures where the animator positions objects at the end of the linkage and the program calculates the positions and orientations of all other objects in the linkage.

With inverse kinematics you can quickly set up and animate complex motions. The basic procedure involves these tasks:

- Building a model of a jointed structure.
- Linking the model together and defining pivot points, as described in chapter 23, “Building Hierarchies.”
- Defining joint behavior at the pivot points.
- Animating the model either by manipulating objects interactively or by applying an IK calculation to follow another object in the scene.

See the Online Reference for details on all IK parameters.

Understanding Inverse Kinematics

Inverse Kinematics is a positioning and animation method built on the concepts of hierarchical linking. To understand how IK works you must first understand the principles of hierarchical linking and forward kinematics. If you have not yet read chapter 23, "Building Hierarchies," you should do so now.

Differences Between Forward Kinematics and Inverse Kinematics

Forward kinematics uses a top-down method where you begin by positioning parent objects and work down the hierarchy positioning each child object.

Basic principles of forward kinematics include:

- Hierarchical linking from parent to child.
- Pivot points defining joints between objects.
- Children inheriting the transforms of their parents.

These principles are fairly forgiving. As long as everything is linked together and the pivots are located at joint locations, you can successfully animate the structure.

Inverse kinematics uses a goal-directed method, where you position a child object and the program calculates the position and orientation of the parent objects. The final position of the hierarchy, after all of the calculations have been solved, is called the *IK solution*.

Inverse kinematics starts with linking and pivot placement as its foundation and then adds the following principles:

- Joints are constrained with specific positional and rotational properties.
- Position and orientation of parent objects are determined by the position and orientation of child objects.

Because of these added constraints, IK requires greater thought about how you link your objects and place pivots. Where many different solutions for linking objects may be suitable for forward kinematics, there are usually just a few good solutions for any given IK approach.

Inverse kinematics is often easier to use than forward kinematics and you can quickly create complex motions. The drawback is that you give up some of your control to the IK functions.

Choosing an IK Method

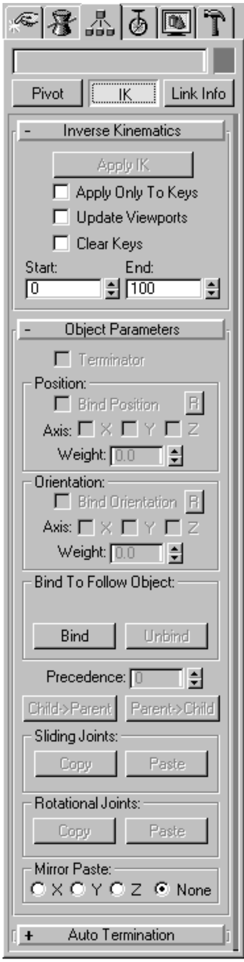
You can choose between methods of animating with IK.

Standard IK—Applies separate IK solutions as standard transform animation keys.

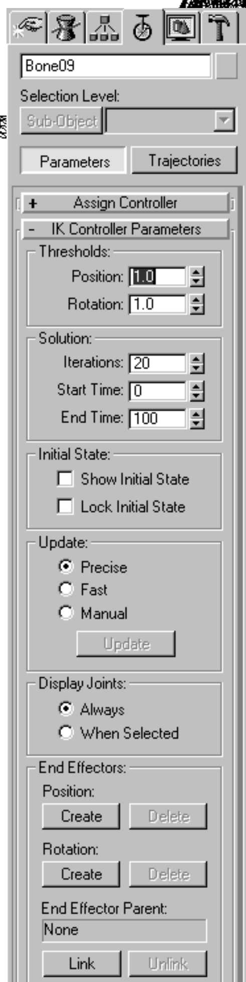
- Works with any linked hierarchy of objects except bones using the IK controller.
- Can combine forward kinematics with inverse kinematics on the same objects.
- Can be applied automatically to a range of frames or interactively to single frames.
- Must be reapplied after you make changes.

New IK—Uses an IK controller to apply IK solutions procedurally across a range of frames.

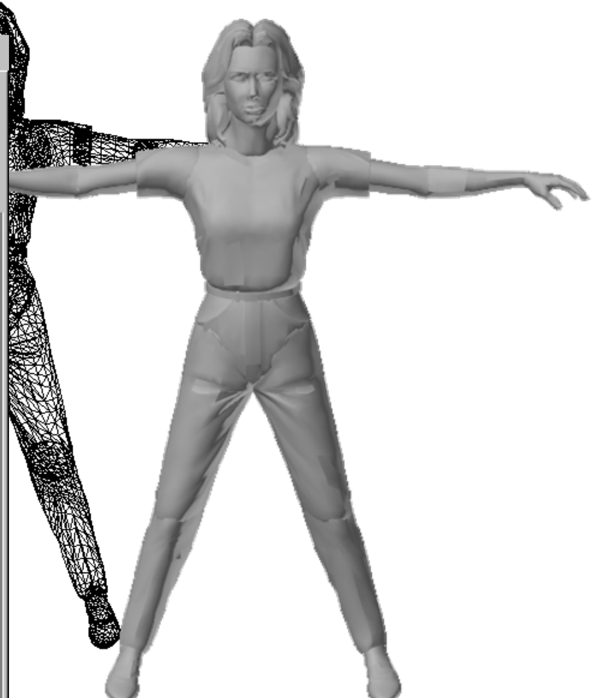
- Works only on bones using the IK controller.
- Calculates IK solutions for all frames in real-time as you make changes.
- Creates keys for end effectors only. This uses far fewer keys than Applied IK and is therefore easier to edit.
- Graphically displays active joint axes and joint limits.



Old IK method accessed through hierarchy panel

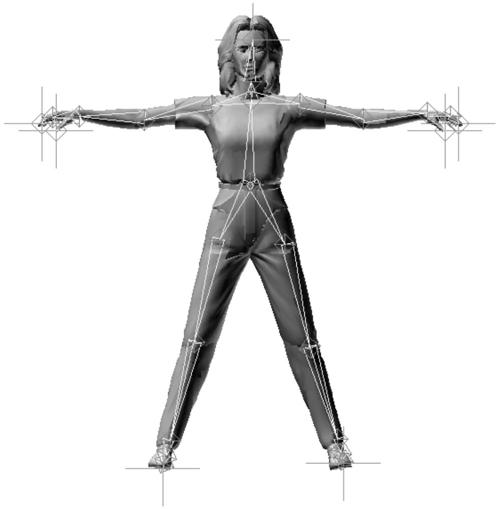


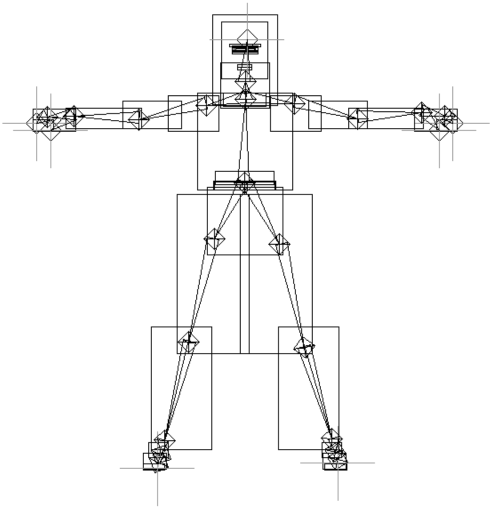
New IK method accessed through motion panel



Model suitable for applying bones

Defining IK Terms





Using inverse kinematics requires that you set parameters for a number of IK components. Brief definitions of these components follow; details are provided later in this chapter.

IK Joints

A joint controls how an object transforms with respect to its parent. You specify joint behavior with settings in three categories:

Object Pivot Point—The location of an object's pivot point defines where joint motion is applied.

Joint Parameters—Changing the IK settings in the Hierarchy command panel determines the direction, constraints, and order of how the joint operates.

Parent Pivot Point—The location of an object's parent pivot point defines the origin from which the joint constraints are measured.

The commands you use to place the pivot points for both the object and its parent are described in chapter 23, "Building Hierarchies."

Kinematic Chains

Inverse kinematics calculates the position and orientation of objects in a *kinematic chain*. The kinematic chain is defined as a single branch of the hierarchy that starts with a selected child object and continues up through its ancestors until it reaches the base of the chain. The base of the chain is either the root of the entire hierarchy or an object that you specify as a *terminator* for the chain. 3D Studio MAX automatically determines the kinematic chain when you select and transform an object with IK.

End Effector

For any IK solution, you explicitly move one object. 3D Studio MAX uses IK calculations to move and rotate all other objects in the kinematic chain to react to the object you move. The object that you move is the *end effector*.

Terminators

You can explicitly set the base of an IK chain by defining one or more objects as terminators. A terminator object stops the IK calculations so that objects higher up the hierarchy are unaffected by the IK solution.

Terminator objects are also used to define hierarchies that use multiple kinematic chains.

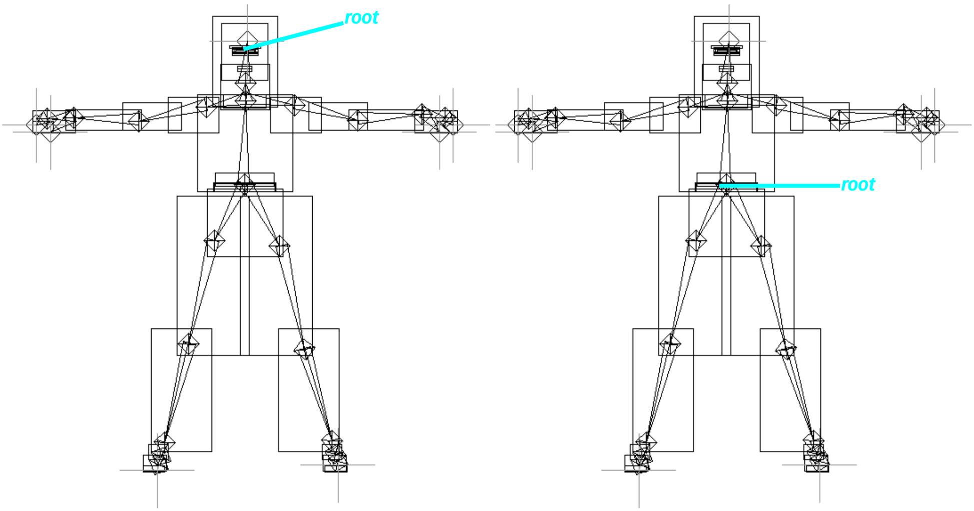
Bound Objects

Objects in your hierarchy can be bound to the world, or they can be bound to other objects called *follow objects*.

Binding allows objects in your hierarchy to be influenced by objects that are not part of the hierarchy.

- An object bound to the world will attempt to maintain its current position and orientation.
- An object bound to a follow object will attempt to match the position and orientation of the follow object.

Linking Objects for Inverse Kinematics



Both structures are suitable for forward kinematics. The structure on the right is best for most inverse kinematics

The same principles you use to analyze linking for forward kinematics also apply to linking for inverse kinematics. Inverse kinematics is less forgiving and is highly dependent on the linking strategy for performing IK calculations.

Two principles for hierarchical linking were described in chapter 23, “Building Hierarchies”:

- The hierarchy follows a logical path from parent to child.
- Parent objects move less than their descendants.

You need to consider two additional principles when linking hierarchies for use with inverse kinematics:

- Links and pivot placement simulate real-world joint locations.
- Choose an object near the structure’s *center of gravity* as the root of the hierarchy. The center of gravity in the real world is the point on an object about which reactions to external forces are applied.

The above figure shows two approaches to linking a skeletal structure. Either structure is suitable for working with forward kinematics. The structure on the right, however, is a better choice for working with inverse kinematics.

- The root object is located near the body’s center of gravity.
- The link order more closely simulates the connections of a real body.

The structure on the left has the arms and torso linked to the neck. The structure on the right links the arms and neck to the torso, a more realistic approach.

Choosing the Root of a Hierarchy

You can find the best candidate for the root of your hierarchy by asking the following question:

If I move this object, should all of the other objects in the hierarchy move with it?

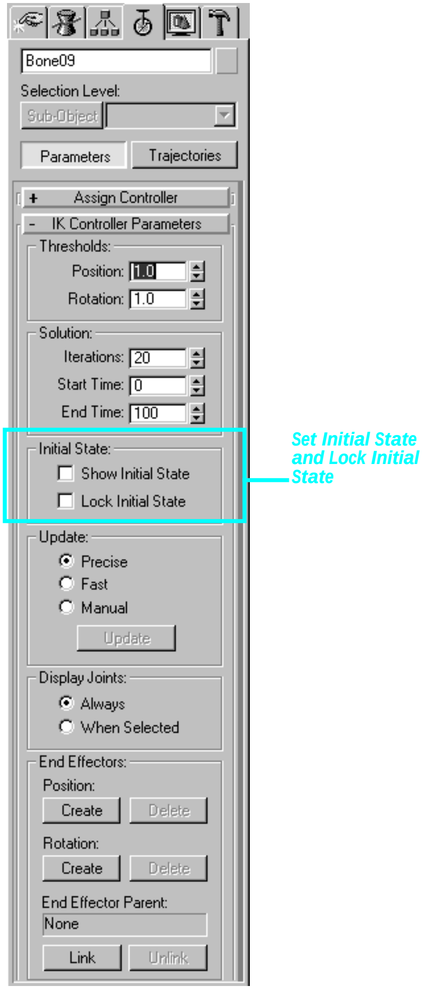
- If the answer is "not often," then you are probably looking at a child object. Examples of this type of object are hands, lamp shades, and tree leaves.
- If the answer is "almost always," then you are looking at a likely candidate for the root object. Examples of this type of object are a torso, a lamp base, and a tree trunk.

Once you have a few candidates for the root object, you can examine them in greater detail. Use these criteria to determine a good root object for IK animation:

- The root object is near the hierarchy's center of gravity.
- Moving the root object usually has a great effect on all other objects in the hierarchy.
- The root object is often unaffected by movement of other objects in the hierarchy.
- The root object is rarely the primary object animated.

The one object that best satisfies these criteria is your root object. You then create your hierarchy with all of the other objects as descendants of that root object.

Creating Bones with the IK Controller



You must create bones if you want to use New IK animation. Because you must use bones with the IK controller, this method works best for the following types of animation:

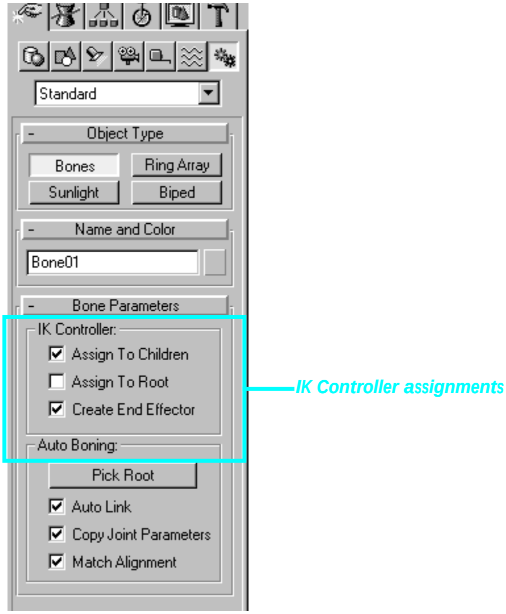
- Animating skinned objects. Skinned objects are usually one-piece meshes that are deformed by a modifier such as Physique, from the Character Studio plug-in. The modifier applies deformation based on the motion of bones associated with the mesh object.
- Animating segmented models linked to a bones structure. Segmented models use a separate object for each jointed portion. For example, the upper arm, forearm, and hand are all separate objects in a segmented model. If you link these objects to a matching bone structure, you can animate the model with IK controllers.

Assigning IK Controllers to Bones

When you create bones, you select the Assign To Children option on the IK Controller rollout on the Create panel.

There is no way to change the controller assigned to bones after they have been created. If you create a bone structure without IK controllers and then change your mind, you must create a new one with Assign To Children enabled.

Tip: You can quickly create a new bone structure with IK controllers by Auto Boning, then deleting, the old bone structure. See chapter 23, “Building Hierarchies.”



Assign To Children—Assigns a slave IK controller to each bone you create—except for the first bone. Slave IK controllers in a bone hierarchy are controlled by a master IK controller. When you edit the IK controller for any bone, you're actually editing the IK settings for all the bones.

Assign To Root—Assigns an IK controller to the root object. Usually you want to clear this option to assign a standard Transform controller to the root bone. When the root bone uses a standard transform controller, it acts as an anchor for the IK solution and does not move during IK. You animate the root bone by selecting and transforming it directly.

You can also select Create End Effector to automatically create a position end effector at the end of each chain. You animate an IK controlled bone structure by animating the end effectors. Position and rotation end effectors can also be created on the Modify panel.

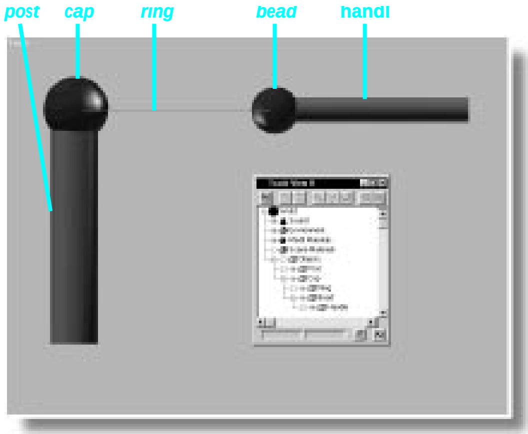
Setting the Initial Position of Bones

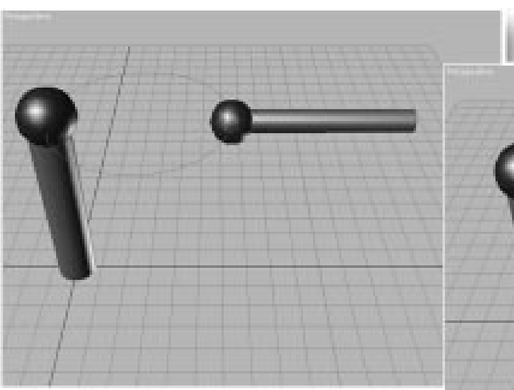
When you first create a chain of bones, the position of the bones is the initial state. You can change the initial state of the bones by transforming bones that do not have end effectors assigned.

When you transform a bone assigned an end effector, you actually transform the end effector, and an IK solution transforms the bones. This alters the position of the bones in the viewport, but not their initial state.

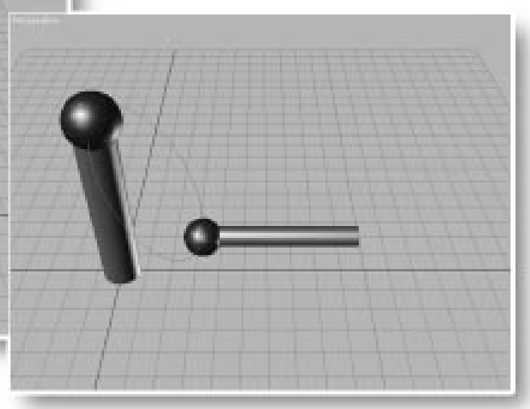
- To change the initial state of a bone that is assigned an end effector, you must select Show Initial State on the Motion panel.
- To prevent bones from being accidentally transformed to a different initial state, select Lock Initial State on the Modify panel. Do this once you have set all bones to the initial state that you want.

Setting Joint Parameters

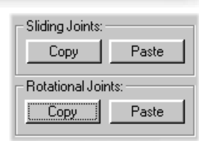




handle before IK



handle after IK



You set whether a joint behaves as a hinge, a drawer slide, or another type of joint by setting joint parameters for each object in the kinematic chain. Joints control the rotation and position of an object with respect to its parent.

Any object has a maximum of two joint-type rollouts. One rollout contains settings to control the object's position. The other rollout controls the object's rotation. There can be many different types of positional and rotational joints. The types of joints available are determined by the position and rotation transform controllers assigned to an object.

Not all controller types can participate in IK, and some controllers place added restrictions on IK. With the possibility of adding many controller plug-ins to 3DS MAX, the best advice is to experiment and see what you can do.

See chapter 27, “Working with Controllers.”

Common Joint Types

The most common joint types are Rotational and Sliding joints. Other common joint types are Path and Surface joints. Each joint type displays its own set of joint parameters.

Rotational joints—Control rotation of objects using many of the standard rotation controllers. The parameters for rotational joints set an object’s ability to rotate about a given axis.

For many IK structures, consider using an Euler XYZ controller. Quaternion based controllers tend to freeze if an object is moved beyond rotational joint limits before turning on IK.

Sliding joints—Control the position of objects using many of the standard position controllers. The parameters for sliding joints control whether an object can move along a given axis.

Surface joints—Control the position of objects using a Surface position controller. The parameters control how the object travels along its assigned surface.

Path joints—Control the positional motion of objects using the Path position controller. The parameters for path joints control how far an object can travel along its assigned path.

Tip: When you are using an object with a path controller in an IK chain you may want the path to appear as if it is part of the IK chain. You achieve this effect, by linking the object using the path controller and the path to the same parent. The path object should have no children, and other objects in the IK chain should be linked to the object using the path controller.

In the figure, the bead uses a path controller to hold it to the ring. The IK chain from parent to child is Post->Cap->Bead->Handle. The ring is a child of the Cap but is not part of the IK chain.

Copying and Pasting Joint Parameters

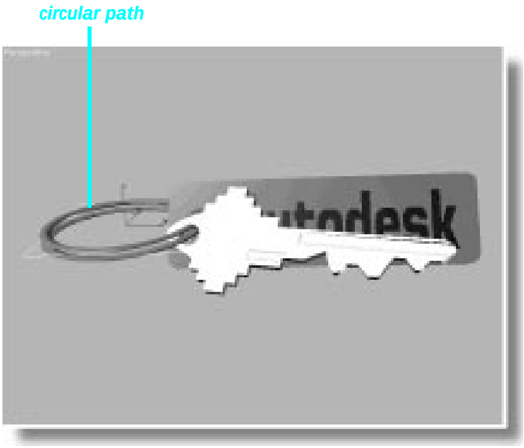
You can Copy and Paste at the bottom of the Object Parameters rollout in the Hierarchy panel to copy and paste a complete set of joint parameters from one object to another.

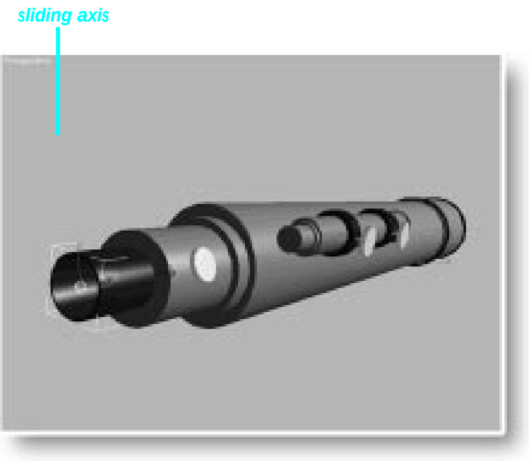
There are separate sets of Copy and Paste for Sliding Joints and Rotational Joints. Each joint type saves copied parameters in separate clipboards.

Select one of the Mirror Paste options if you want to mirror the joint parameter settings as you paste them. This is very useful when pasting from one side of an object to another, such as left arm joints to a right arm.

You can also copy joint settings from a non-IK controller to an IK controller, but you can't copy from an IK controller to a non-IK controller.

Activating Joint Axes





You set whether an object can move or rotate about a given axis by using the Active check box in the joint rollouts. Joints have a maximum of six possible axes: three for rotation and three for position. You constrain the motion of a joint by setting which axes are active.

- A joint with all axes active can move and rotate freely, independent of its parent.
- A joint with all axes inactive is locked to its parent and cannot move independently.
- The setting of IK joint axes overrides Inherit and Lock settings on the Link Info rollout.

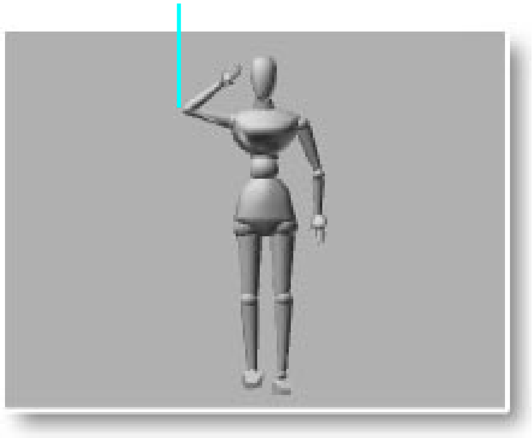
Understanding Joint Axis Orientation

IK joint axes for an object are defined by the Local axes of the object's parent. That means if you activate the *X* axis of an object's rotational joint parameters, the object rotates about its parent's *X* axis, not its own *X* axis.

You might have a problem when you set joint parameters for an object whose local coordinate system is oriented 90 degrees from its parent's coordinate system. In such a case, the rotation angle about one axis becomes indeterminate. The result is that one axis will often cause rotation about one of the other two axes.

The following techniques will make setting joint parameters easier and more successful:

- Assemble IK hierarchies with objects aligned along the World axes.
- Consider using the Adjust Pivot functions to align your object pivots with either the World axes or the root object of the hierarchy. See chapter 23, "Building Hierarchies."
- Set the transform managers to Parent coordinate system and Use Pivot Point Center, while setting joint parameters. This helps you see the orientation of the joint axes by displaying the parent axis icon at the selected object's pivot point.
- When activating joint axes, drag the From spinner up and down. This causes the object to move or rotate about the active axis and is a quick check that you chose the right axis.



Activating Rotational Joints

When you select Active for one of the *X, Y, Z* axes of a rotational joint, the object can rotate about that axis of its parent's coordinate system.

Joints that rotate about multiple axes are very common. A ball joint, like your shoulder, is a rotational joint active about all three axes. A pin joint, like your elbow, is a rotational joint active on a single axis.

Activating Sliding Joints

When you select Active for one of the *X, Y, Z* axes of a sliding joint the object can move along that axis of its parent's coordinate system.

Most sliding joints are active only along a single axis. A telescope is an example of a sliding joint active on a single axis. You rarely see a sliding joint active along all three axes.

If a sliding joint is active along all three axes, it moves independently from its parent. It's almost as if there is no joint connection at all.

Path and Surface Joints

When you select Active for Path or Surface joints you are setting whether the object can move along the assigned path or surface.

A house key on a ring is an example of an active path joint.

Limiting Joint Action

Most joints are limited in their range of motion along an active axis. For example, a hinge may only open to 120 degrees or a piston only slides within the length of its cylinder. To limit the range of motion allowed on an active axis, you select Limited and set values for the From and To fields on the joint rollouts.

You can also view joint limits in the viewport by pressing and holding the mouse button on the From or To label of a limit field. The object will move or rotate to the limit value until you release the mouse button.

Limiting Rotational Joints

Limits for a rotational joint define how far the object can rotate about its parent's axes. The values in the From and To fields represent the rotation angle about the active axis measured from 0 degrees on the parent object.

For example, an elbow joint rotates the forearm with respect to the upper arm. In the figure the limits on *X*-axis rotation are from 0 to 135 degrees. The *Y*, *Z* axes are inactive because an elbow joint rotates about a single axis.

Limiting Sliding Joints

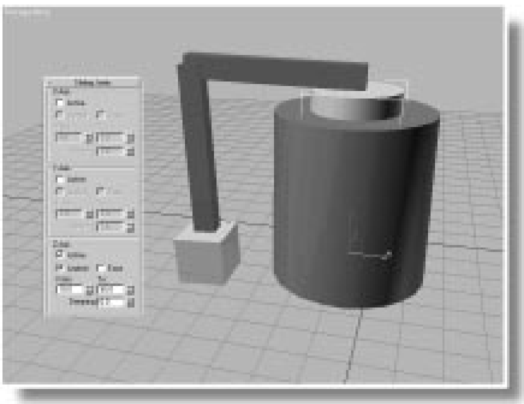
Limits for a sliding joint define how far a joint can move along its parent's axes. The values in the From and To fields represent a distance for movement along the active axis measured from the pivot point of the parent to the pivot point of the selected object.

For example, a sliding joint on a piston moves the piston in and out of the cylinder. In the figure, the limits on *Z*-axis movement are from 10 to 90. This prevents the piston from hitting the bottom or moving past the end of the cylinder.

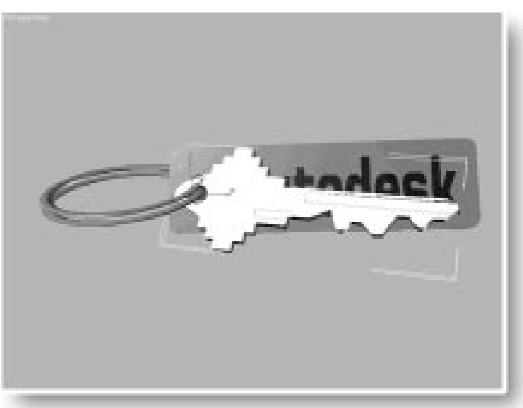
Limiting Path and Surface Joints

Limits for path and surface joints define how far along the path or surface an object can move. The values in the From and To fields represent a percentage of the total distance measured along the path or surface.

For example, a path joint for a house key moves the key along a key ring. In the figure, the limits on the path joint are from 5% to 95%. This prevents the key from traveling along the ring where the fob is attached.

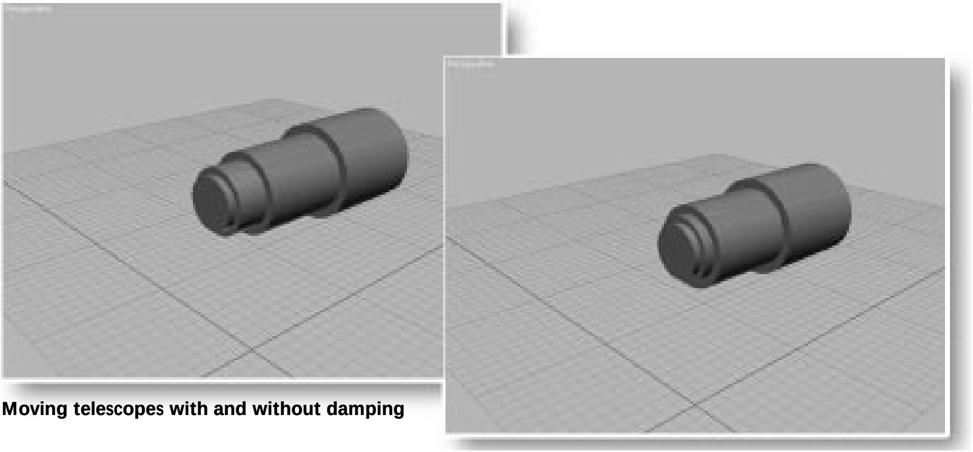


Example of limiting a sliding piston joint



Example of limiting a path joint

Setting Joint Resistance and Spring Back



Moving telescopes with and without damping

You control a joint's resistance to motion, and its tendency to return to its original position, by setting Ease, Damping, and Spring Back options.

Easing a Joint at its Limits

An organic joint, or a worn mechanical joint, moves freely in the middle of its range of motion but moves less freely at the extremes of its range. Use Ease to cause a joint to resist motion as it approaches its From and To limits.

For example, your forearm might move freely in the middle of its range of motion, but it resists movement when you try to squeeze it against your upper arm or extend it all the way out. Ease simulates this effect.

Damping Joint Action

As a joint corrodes, dries out, or is put under a heavy load, it resists motion along its active axes. Damping simulates the natural effect of joint friction or inertia. Enter a value greater than zero in the Damping field to apply resistance over a joint's full range of motion.

As damping increases a joint resists motion and other joints are required to move more. A damping value of 1.0 means there is extreme resistance and a joint will not move on that axis.

For example, a telescope with no damping at all allows each cylinder to move to its maximum limit before the next cylinder moves. If the cylinders have damping values assigned, then each cylinder causes its parent to begin moving before it reaches full extension.

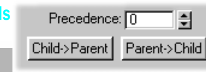
Setting a Joint to Spring Back

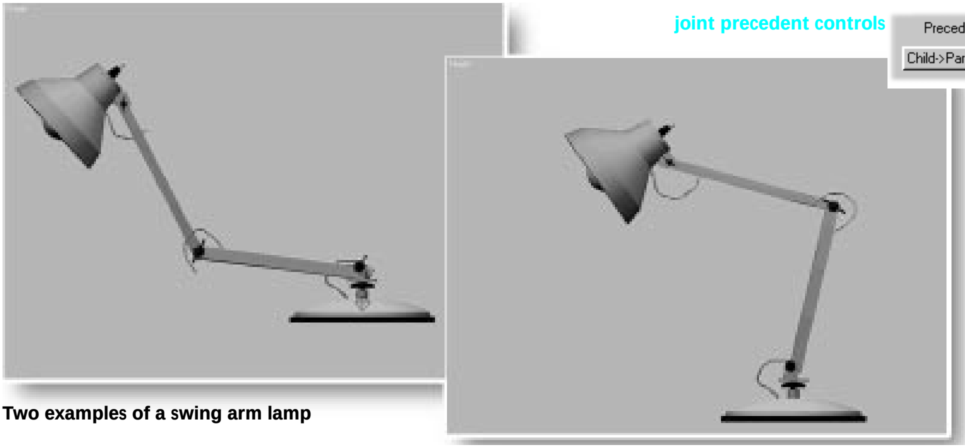
When a joint resists motion, it also has a tendency to return towards its at-rest position. You simulate this by setting Spring Back tension in the joints. As the joint moves further from its rest position, an increasingly larger force pulls the joint back, like a spring.

When you set Spring Tension higher, the spring pulls harder as the joint moves farther away from its rest position. Very high settings can turn the joint into a limit, because you can reach the point where the spring is too strong to allow the joint to move any farther.

Setting Joint Precedence

joint precedent controls





Two examples of a swing arm lamp

You set Joint Precedence to control the order in which joint calculations are applied to the kinematic chain. When 3D Studio MAX calculates an IK solution the result is dependent on the order of calculation for each joint.

For any given position of an end effector, there are many possible IK solutions. The figure shows two equally valid IK solutions for the placement of a lamp.

The three joint precedence controls in the Object Parameters rollout are:

Precedence—Sets joint precedence manually.

Child–>Parent—Automatically sets joint precedence to decrease in value from child to parent.

Parent–>Child—Automatically sets joint precedence to decrease in value from parent to child.

Determining Order of Calculations

You control the IK solution by setting joint precedence to determine which joints contribute the most to the IK solution and which joints contribute the least.

- Joints with high precedence values are calculated first, and contribute more motion to the IK solution.
- Joints with low precedence values are calculated last and contribute the least motion to the IK solution.
- Joints with equal precedence values are calculated by order in the hierarchy. Joints closer to the end effector are calculated first and joints closer to the root are calculated last.

Of all of the joint parameters, Joint Precedence is the most subtle. The following topics describe various joint precedence strategies and how to use them.

Using Default Joint Precedence

3D Studio MAX uses a default joint precedence that is suitable for many IK solutions. It assumes that joints closest to where a force is applied (the end effector) will move more than joints farther from the force.

The default joint precedence is a value of 0 for every joint: *all joints have the same precedence value*. This default starts calculations with the end effector and progresses up the IK chain until the base object is reached.

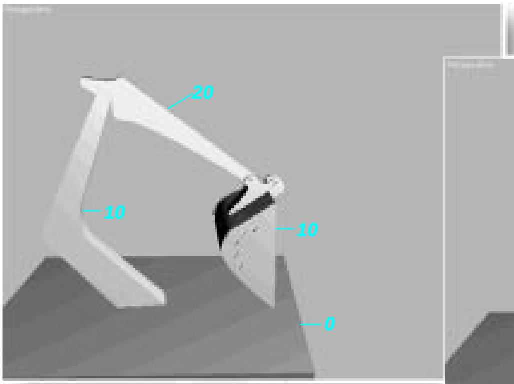
Using methods described in the following topics you can change the default precedence values. You can also set your kinematic chain back to its default precedence.

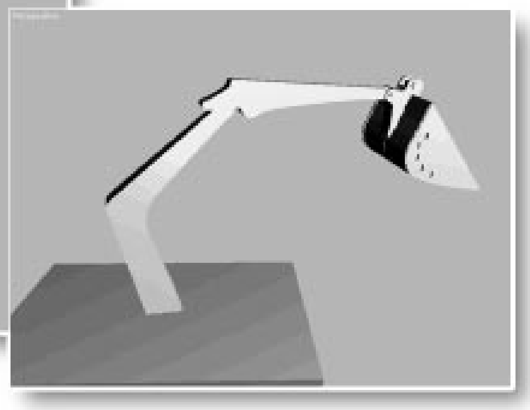
To reset the default joint precedence:

1. Select all objects in the kinematic chain.
2. On the Hierarchy panel, expand the Object Parameters rollout.
3. Set Precedence to 0.

Note: Default joint precedence occurs whenever all joints in the kinematic chain have the same precedence value. Assigning a value of 100 to all objects in the kinematic chain is exactly the same as assigning a value of 0.

Choosing Child->Parent Precedence





Child->Parent precedence values

Child->Parent precedence causes joints closest to where a force is applied (the end effector) to move more than joints that are farther away from the force. Like the default precedence, Child->Parent precedence produces a natural result that is suitable for many IK solutions.

The difference is that Child–>Parent precedence assigns *unique* values to each object in the kinematic chain (default precedence uses the value of 0 for every object). Child–>Parent precedence is more flexible if you want to go back and manually change precedence values.

Assigning Child->Parent Precedence

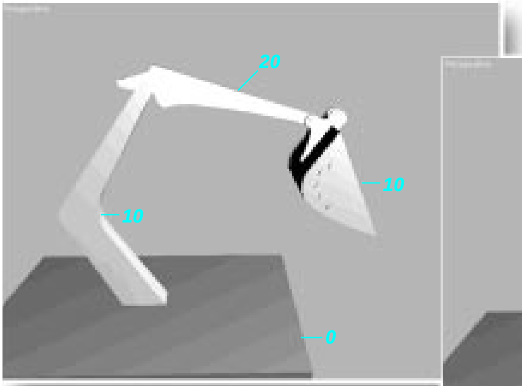
Clicking Child–>Parent sets joint precedence based on a child having a higher precedence than its parent. The values are calculated by setting the root of the entire hierarchy to a precedence value of 0 and each child to a value equal to 10 times its depth from the root.

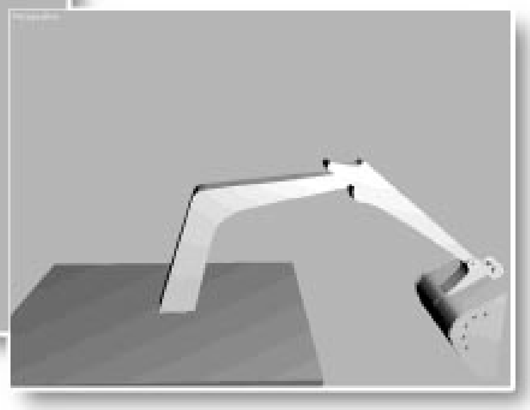
You almost always assign Child->Parent precedence to an entire kinematic chain. The chain for the structure in the figure uses the block as the root object and the scoop as the end effector. Using Child–>Parent precedence, the block is assigned a precedence of 0 and the scoop is assigned a precedence of 30.

Each object receives a value equal to its depth from the root of the hierarchy times 10. Note that precedence value calculation is based on the root of the hierarchy whether or not the root is selected as part of the kinematic chain.

You might want to assign Child->Parent precedence to just a single object in the kinematic chain. For example, if you have been manually changing precedence values, you might want to set an object to its original Child->Parent value. The value assigned to the object is equal to its depth from the root of the hierarchy times 10.

Choosing Parent–>Child Precedence





Parent->Child precedence values

Parent->Child precedence causes joints closest to where a force is applied (the end effector) to move less than joints farther away from the force. This is the opposite of Child->Parent precedence. Parent->Child precedence assigns the highest precedence to the base object and the lowest precedence to the end effector.

You might use this type of precedence for a number of special cases. For example:

- When joints near the end effector are less flexible than joints near the base. You use Parent->Child precedence in conjunction with limits and damping.
- When animating a kinematic chain where the real-world object is moved by applying force to the base of the chain. Imagine an animal's tail. Using IK you animate the tail by moving the tip of the tail. However, a real animal moves its tail using muscles at the tail's base. The real-world motion is applied from the base to the tip in a Parent->Child order.

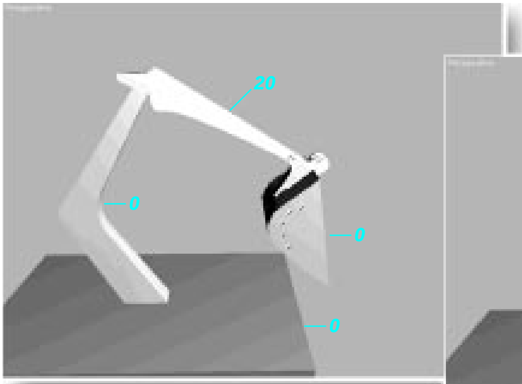
Assigning Parent->Child Precedence

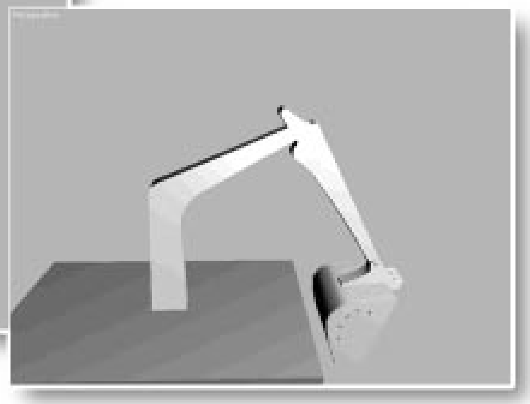
Clicking Parent–>Child sets joint precedence based on a child having a lower precedence than its parent. The values are calculated by setting the root of the entire hierarchy to a precedence of 0 and each child to a value equal to –10 times its depth from the root.

You almost always assign Parent->Child precedence to an entire kinematic chain. The chain for the structure in the following figure uses the block as the root object and the scoop as the end effector. Using Parent->Child precedence, the block is assigned a precedence of 0 and the scoop is assigned a precedence of –30.

You might want to assign Parent->Child precedence to a single object in the kinematic chain. The value assigned to the object is equal to its depth from the root of the hierarchy times –10.

Setting Manual Precedence





Parent->Child precedence values assigned manually

Some models and animated motions don't fit neatly into a Child->Parent or Parent->Child precedence. In such situations you can manually assign precedence values to any object in the IK chain. For example:

- Animating models with a combination of light, flexible joints and heavy, resisting joints. Imagine a model of heavy iron balls linked together with lengths of chain. Setting the precedence values of the chains higher than the precedence values of the iron balls simulates the balls' inertial resistance to motion.
- Animating a motion where certain joints must move before other joints. Imagine a golfer's arm where the elbow should remain locked while swinging a golf club. You could accomplish this by setting the precedence of the elbow lower than the precedence of the wrist and shoulder.

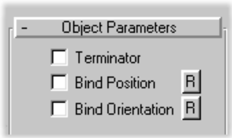
High precedence values are calculated before low precedence values. Precedence values that are equal are calculated in Child->Parent order.

The precedence calculations only consider the relative IK values. This means that an IK chain of three objects with precedence values of 0, 30, and 200 would have the same solution if the precedence were changed to 1, 2, and 3.

Assigning Precedence Manually

You assign precedence manually on a joint-by-joint basis. For example, the chain for the structure in the figure uses the block as the root object and the scoop as the end effector. Joint number 3 has been assigned a precedence value of –10, leaving the other joints at their default value of 0.

Terminating Chains



Terminator option in the Object Parameters rollout

3D Studio MAX automatically defines the kinematic chain using the object you select as the end effector and working up the hierarchical tree to use the root object as the base of the kinematic chain. Sometimes you might not want the kinematic chain to go all the way to the root of the hierarchy. This is especially true when you are animating multi-limbed or branching structures, such as an octopus or a tree.

You use the Terminator option in the Object Parameters rollout to stop calculation of the kinematic chain before it reaches the root object of the hierarchy. A terminator object stops calculation at the terminator's child object; the terminator itself is not affected by the IK solution. This gives you very precise control over the behavior of the kinematic chain.

For example, look at the mannequin in the following figure. The mannequin is linked with its waist as the root of the hierarchy. If you use inverse kinematics to move the hand, all objects from the hand to the waist will be affected by the IK solution. However if you define the torso as a terminator object, then only the objects from the hand to the shoulder are affected.

Animating with Inverse Kinematics

Once you have set up the IK structure, you are ready to begin animating with IK. There are three methods for creating IK animation:

Standard Interactive IK—You animate your IK structure by activating IK transform mode and manually animating the position of end effectors. IK solutions are only calculated for the key frames you set; all other motion is interpolated as set by the object's controllers.

Standard Applied IK—You animate follow objects and have 3D Studio MAX calculate the IK solution on every frame of a specified range.

New IK—You animate your IK controlled bone structure by turning on Animate and transforming the end effectors of the bone structure. IK for bones is always interactive and IK solutions are calculated for every frame.

Examining the Types of IK

The Interactive IK method combines the expressiveness of forward kinematics while taking advantage of the joint constraints and calculation ability of inverse kinematics. With IK mode turned on, it is easy to position your IK structure at specific keyframes. 3D Studio MAX then interpolates positions between keyframes the same as it does for any other object.

You can use the interactive method to easily create expressive animations with just a few keyframes. If the interpolated results don't meet your requirements, then you have to add more keyframes, just as with regular animation using forward kinematics.

The Applied IK method quickly creates very precise animation with little or no interpolation. Applied IK uses follow objects to define all motion and then calculates an exact IK solution for every frame.

This method is fast and accurate but it creates keys for every object in the kinematic chain on every frame. The many keys that are generated make it difficult to adjust the animation manually once IK has been applied.

The New IK method combines the expressive control of Interactive IK with the precision of Applied IK. Because this method uses an IK controller for all of the bones in the structure, it only requires keyframes for the end effector.

The following topics describe how to use these methods of IK animation.

Binding Objects

Use the binding functions to cause an object to attempt to maintain its position and orientation relative to the world or to another object. A bound object will resist motion in the kinematic chain until the chain is stretched near its limits and the bound object must move to complete the IK solution.

Setting Binding Options

You can select to bind an object by position, orientation, or both. Once an object is bound you can set binding axes, weight, and whether the binding is absolute or relative.

Bind Position—Causes an object to attempt to maintain its location or to match the location of a follow object.

Bind Orientation—Causes an object to attempt to maintain its rotational orientation or to match the orientation of a follow object.

Relative—Changes how an object reacts to a bound follow object. See the following discussion about binding an object to a follow object.

Axis and Weight—Set these options to control which axes are affected by the binding and the binding's influence over other bindings in the kinematic chain.

Binding an Object to the World

You can bind an object to the world if you want the object to hold its position and orientation as long as possible during IK operations.

If you are animating a walking figure you want one foot to remain in place while you position the other foot. Without binding, the whole hierarchy would have a tendency to slide around when you attempt to position a foot. Binding both feet to the world ensures that the unselected foot stays in place while you position the other foot.

Binding an Object to a Follow Object

You can bind a selected object in your hierarchy to any other object that is not a descendant of the selected object. This other object is called the *follow object*.

The behavior of the bound object varies depending on the state of the Relative buttons next to the bind options.

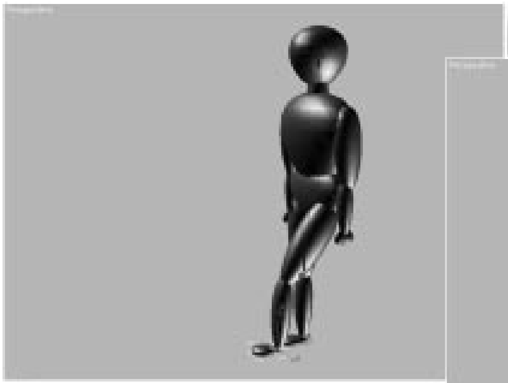
- When Relative is inactive, the bound object tries to match the exact position and orientation of its pivot point to the position and orientation of the follow object's pivot point.
- When Relative is active, the bound object mimics any changes in position or rotation of the follow object but does not try to match it exactly.

You might want to animate a figure that always points to another object. Bind the hand of the figure to the other object with Relative active. Turn IK on, and as you move the object, the hand and arm of the figure move to point at it.

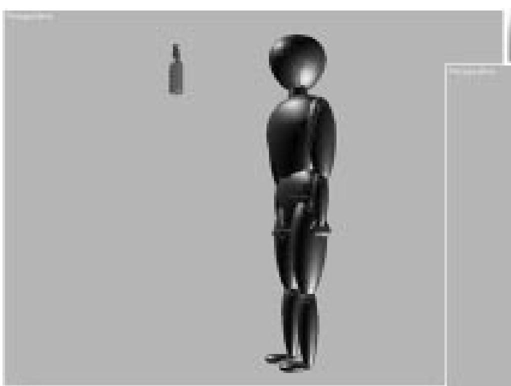
Unbinding Objects

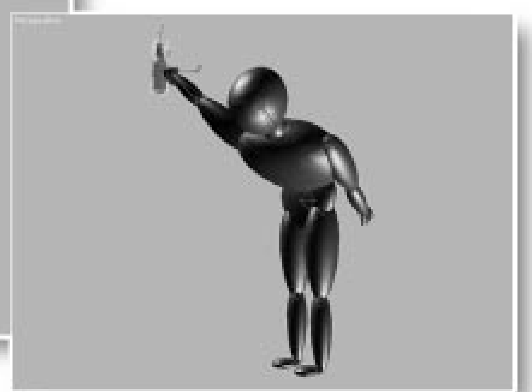
If you decide that you do not want an object to be bound you can either clear the Bind Orientation and Bind Position boxes, or unbind the object from its follow object.

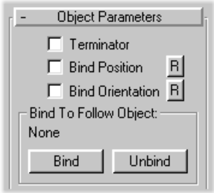
- If you clear the binding options, you disable the effect of binding without removing the binding to a follow object. This method is handy if you decide to reposition a follow object after you have bound an object to it.
- If you unbind an object, it permanently removes the binding to a follow object.


Unbound foot slides


Bound foot stays put




Hand bound to bottle



Inverse kinematics bind controls

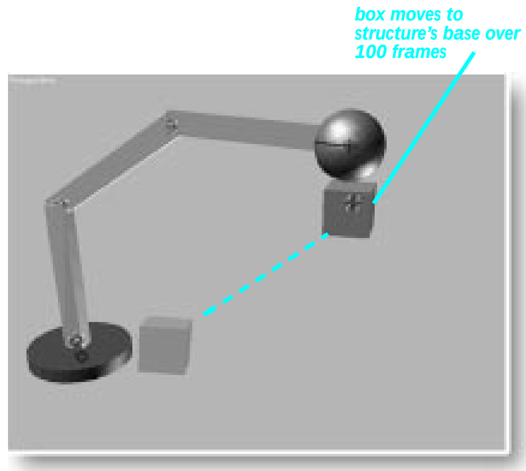
Animating with Interactive IK

With IK and Animate turned on, you position your model on keyframes, and 3DS MAX interpolates the IK solution between those keyframes. Because the IK solution accounts for multiple objects and the joints between objects, the interpolated animation of an IK object is usually different from the animation of objects without IK.

The interactive method of IK animation is best suited for expressive character animation or animations where the exact placement of objects is restricted to a few keyframes.

Example of Standard Interactive IK

The following example demonstrates how an interactive IK animation works and how it might differ from what you expect. The figure shows an IK structure with its end effector resting on top of a box. The box moves in a straight line over 100 frames.



IK structure

Results of animating the box and IK structure.

Turn the Animate and IK buttons on and then move the end effector of the bone structure to rest on top of the box at frame 100. The interpolated animation of the end effector follows a natural looking curved path.

You might have reasonably expected the IK bone structure to follow the same path as the box. However the IK solution is only calculated at the keyframes. The positions and rotations of all the objects in the kinematic chain are interpolated between the keyframes to produce the curved result.

To make the end effector closely follow the path of the box, you would have to add more keyframes. You can also use binding and standard applied IK. See “Animating with Applied IK,” to make the end effector precisely follow the motion of the box.

Behavior of Objects in IK Mode

As you move and rotate objects using standard interactive IK you notice that some objects might not be able to move or rotate about all axes. This is because the objects are constrained by the joint parameters you have set. If the joint parameters specify that motion cannot occur in a certain axis, the end effector will not move.

Standard interactive IK looks at the effects of moving and rotating the end effector. Scaling the end effector has no effect on the kinematic chain. Scale transforms always use forward kinematics regardless of the state of the IK button.

Behavior of Root Objects in IK Mode

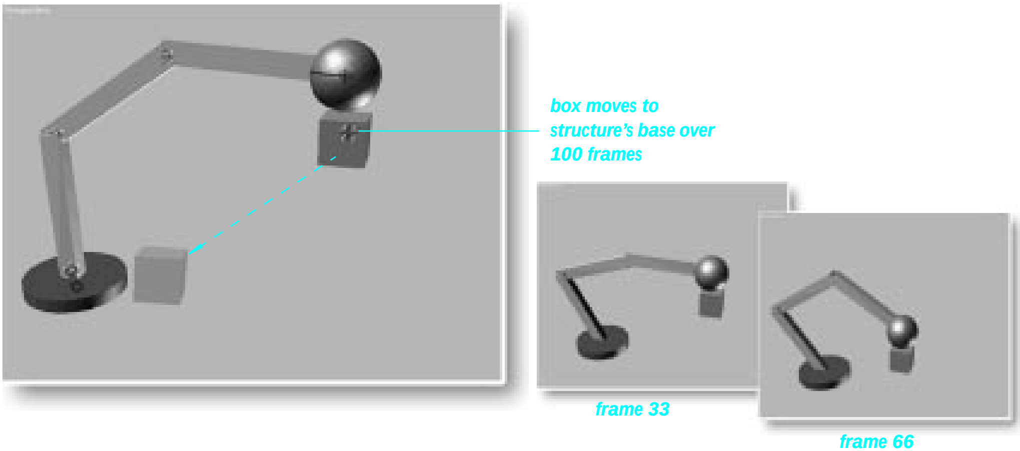
3D Studio MAX uses an option on the Inverse Kinematics panel of the Preferences Settings dialog to enable the special case of moving and rotating root objects when IK mode is on. The name of this control is Always Transform Children Of The World.

While you are transforming an end effector using IK mode, the settings for the root object's joint parameters are used and the root object does not move with respect to the World. If you select the root object, its joint parameters are released and you can transform the root object.

If you decide you want root objects to always use their joint parameters, you can clear the Always Transform Children Of The World option.

Single, unlinked objects are hierarchies of one. An unlinked object is its own root and is also a child of the World. Clearing Always Transform Children of the World will prevent you from transforming single objects in IK mode.

Animating with Applied IK



Applied IK requires you bind one or more parts of your IK structure to animated follow objects. Once bound, you can select any object in your kinematic chain and click Apply IK. 3D Studio MAX then calculates the IK solution for each frame of the animation and places transform keys for every object in the IK chain.

The Apply IK method of animation works best when you want objects in the kinematic chain to match the motions of other objects exactly.

Example of Standard Applied IK

The following example demonstrates how a standard applied IK animation works. Just as in the example in "Animating with Interactive IK," the following figure shows an IK structure with its end effector resting on top of a box. The box moves in a straight line over 100 frames.

To Bind the end effector to the box click Bind. Once the end effector is bound to the box, it will try to match the location of its pivot point with the pivot point of the box.

Select the end effector and click Apply IK. 3DS MAX matches the end effector with the box and calculates the IK solution for every frame. Playback of the animation shows that the end effector perfectly follows the box.

Clearing Keys from Previous Animation

If you have animated any members of the IK chain interactively, or run Apply IK previously, the existing animation keys will affect the new IK solution. Sometimes that is exactly what you want. You can use manual animation to subtly nudge the IK solution towards a particular result. However, it is more likely that you want to remove old keys so that 3DS MAX begins the IK calculations with a clean slate.

The Clear Keys option in the Inverse Kinematics rollout controls whether or not old move and rotate animation keys are removed.

Constraining the IK Solution to Specific Frames

You use the Start and End fields in the Inverse Kinematics rollout to set the range of frames used to calculate the applied IK solution. Using these fields, you can restrict IK solutions to specific frames and solve for different solutions in different time segments.

You can set the Start and End fields to include frames outside of the active time segment.

Constraining the IK Solution to Specific Frames

You can also constrain an IK solution to frames where a key exists for an end effector. This is useful if you want to animate a hierarchy using end effectors but do not want keys generated on every frame.

Select the Apply Only To Keys option to constrain the IK solution to frames with end effector keys.

Watching Progress of the IK Solution

You can select Update Viewports on the Inverse Kinematics rollout to watch the progress of Apply IK frame by frame. 3DS MAX normally calculates all frames for Apply IK before updating the viewports.

Updating the viewports greatly slows down the Apply IK process but it can help you troubleshoot complex animations.

Animating with New IK

Animating with New IK combines the expressiveness of Interactive IK with the precision of Applied IK, and does it with fewer keys.

Animating the End Effectors

You animate a bone structure using IK controllers by animating special end effectors located at the bone joints. There are two types of end effectors—Position and Rotation. They are displayed as three intersecting blue lines at the joint.

When you select and transform a joint that carries either end effector, only the end effector itself is transformed. The bones in the chain then use IK to calculate the IK solution.

Animating New IK end effectors is much like animating with Interactive IK except interpolation between keyframes uses correct IK solutions.

Note: The end effectors used with IK controllers are not compatible with IK mode. If you want to combine New IK with IK mode, use follow objects instead of end effectors.

Linking End Effectors to a Parent

You can link New IK end effectors to a parent to achieve results similar to using bound follow objects with Interactive and Applied IK. To link a New IK end effector, you click Link on the Motion panel.

There are two reasons to link an end effector to a parent:

- You can rescale your entire hierarchy. If you animate a character, and then later decide you want to scale the entire character and its animation, link the character's root object to a dummy, link all of the character's end effectors to the same dummy, and then scale the dummy to scale all of the bones as well as the animation of the end effectors.
- You can use linked end effectors for things like feet or hands. End-effectors are automatically bound to the world, so when you move a root object in your hierarchy, the end effectors remain behind. This is good for keeping feet on the ground, but not very good for hands, where you want them to move along with the character. To have hands follow the root object, link the end effectors of the hands to the root object.

Linking Bones to Follow Objects

You can also use follow objects instead of end effectors and bind New IK bones to any number of follow objects.

Remember, if you use follow objects with New IK you do not need to click Apply IK. The IK controller takes care of everything.

Controlling IK Precision

Even with carefully defined joint precedence and joint parameters there are many valid IK solutions for any placement of an end effector. You set position and rotation thresholds to control IK precision and solution speed.

- If you are using standard IK, the threshold and iteration settings are found on the Inverse Kinematics panel of the Preferences dialog.
- If you are using the IK controller method, the threshold and iteration settings are found on the Motion panel.

Position Threshold—Sets how close the end effector has to get to the follow object or cursor position to be considered a valid solution. The value expresses a distance in the current unit system. Small values increase accuracy but take longer to solve.

Rotation Threshold—Sets how accurately the end effector has to match the orientation of the follow object to be considered a valid solution. The value expresses a rotation angle in degrees. Small values increase accuracy but take longer to solve.

Iterations—Sets the maximum number of times 3D Studio MAX repeats the IK calculations to find a valid solution. A high Iterations value increases the chance that 3DS MAX can calculate a valid solution, but takes longer to complete. The following rules apply:

- Calculations stop as soon as a valid solution is found, even if the maximum number of iterations have not been performed.
- The last solution calculated is used if the maximum number of iterations is reached, regardless of whether the solution satisfies the position and rotation thresholds.

Setting Standard Applied and Interactive Thresholds

You set the threshold and iterations settings as a trade-off between accuracy and speed. You have two groups of settings so you can individually tune the behavior of standard applied IK and interactive IK.

- Use interactive IK if you want fast, real-time response. Your interactive IK preferences should be set for speed.
- Use applied IK if you want the solution to match the follow objects very closely. Your Applied IK preferences should be set for accuracy.

The default applied IK and interactive IK settings are shown in the following table:.

Thresholds	Applied IK	Interactive IK
Position	0.100	1.000
Rotation	0.100	1.000
Iterations	50	10

Chapter

25

Basic Track View Use

Track View is the tool that you use to see a data-driven view of your scene and animation.

The standard views—such as Top, Front, User, and Camera—are geometry-driven views. These views display the surfaces of your objects and show them changing over time.

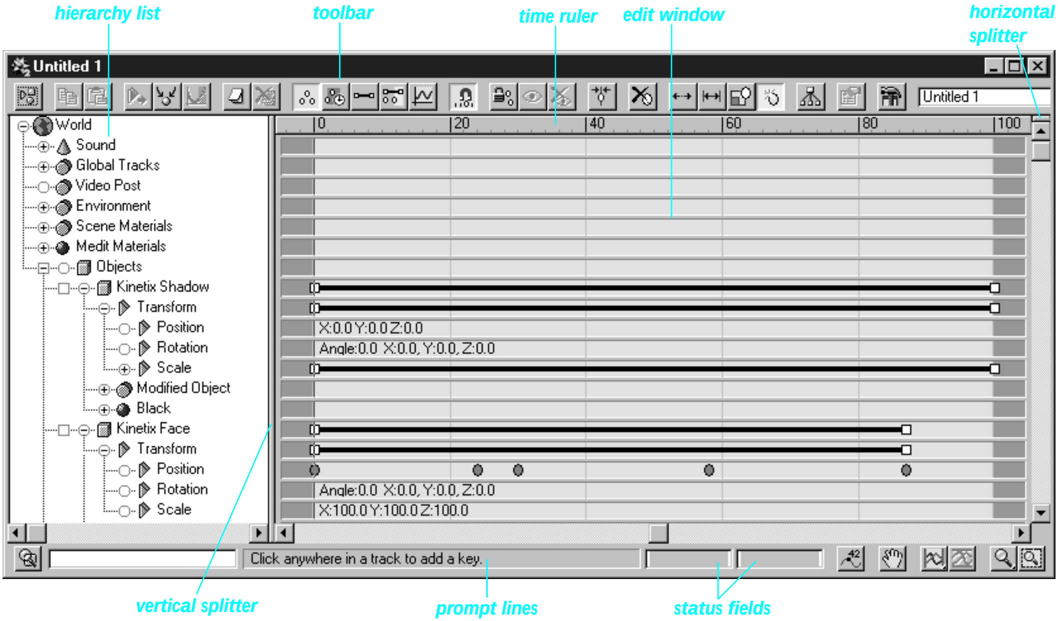
Track View displays the values and timing that produce the geometry and motion you see in standard views. Using the data-driven Track View, you have very precise control over every aspect of your scene.

The traditional way of looking at animation is to break it down into a chart of the entire scene, divided into blocks of time, with main animation keys and ranges of animated effects drawn on the chart. Traditional animators call this the "dope sheet" or "bar sheet." Four of the five animation edit modes in Track View use this method of working with animation data.

These edit modes are Edit Keys, Edit Ranges, Position Ranges, and Edit Time. This chapter covers techniques for working with an animation using the Edit Keys, Edit Ranges, and Edit Time modes of Track View. Certain buttons in the Motion panel that work well with Track View are also covered in this chapter. (The Position Ranges mode is most useful when working with function curves, so it is covered in chapter 26, "Function Curve and Trajectory Editing.")

See the Online Reference for details about using Track View and editing animation.

Understanding Track View Concepts



As described in chapter 22, “Animation Concepts and Methods,” you create animation by setting keys on specific frames and allowing 3D Studio MAX to interpolate the animation on frames between your keys. Track View contains the tools to view and edit all the keys that you create and to set the methods used by 3DS MAX to interpolate between those keys.

The left side of Track View presents a list of everything in your scene. Every object, modifier, transform, material, environment effect, and sound appears in the list, along with its associated animatable parameters. Select items from this list to apply changes to the animation values.

The right side of Track View charts the changes applied to parameters over time. Any change you make to one of these parameters, when the Animate button is on, appears as a key in the right side of Track View. You select keys to apply changes to one or more specific keys.

Use Track View to perform the following tasks:

- Display a list of everything in your scene and their parameters.
- Change key values, timing.
- Change interpolation between keys.
- Edit ranges of multiple keys.
- Edit blocks of time.
- Add sound to your scene.
- Create and manage notes about the scene.
- Animate object visibility.
- Change the behavior of the animation outside the range of keys.
- Change controllers for animated parameters.
- Select objects and hierarchies.
- Store animated global variables for use by other controllers.

Opening a Track View

Track View is like an application window with its own workspace, toolbar, and navigation buttons. You spend most of your time in Track View working with the following main components:

Hierarchy list—Displays all items in your scene in a hierarchical fashion. You select items in the Hierarchy list to select objects and materials in your scene, and to select tracks for display and editing in the edit window.

Edit window—Displays tracks and function curves where you change animation values and timing. Nonanimated parameter values are displayed in tracks until you add animation keys. The edit window also indicates the active time segment with a light gray background.

Toolbar—Contains the tools for working with items, tracks, and function curves.

Status line—Contains areas for prompts, key time and value fields, and navigation control.

Time ruler—Measures time in the edit window. Markings on the time ruler reflect the settings in the Time Configuration dialog. You can drag the time ruler vertically in the edit window to align it with any track.

Opening a Track View Window

You can click Track View on the toolbar for the fastest and easiest way to open a Track View window for quick animation editing.

The first time you click Track View in a scene, a new untitled Track View is opened. After opening the first Track View, clicking Track View again has the following effect:

- If all previous Track Views are open, another untitled Track View is created.
- If you have closed some Track Views, the most recently closed Track View is reopened.

Displaying a Track View in a Viewport

You can display a Track View in a viewport by right-clicking a viewport label and choosing a Track View on the shortcut menu. A layout with a wide viewport—such as default viewport layout B—is usually best for displaying a Track View.

Creating and Deleting Track Views

You can force the creation of a new untitled Track View window by choosing New Track View on the Track View menu.

You can delete a stored Track View by choosing Delete Track View on the Track View menu.

Saving and Restoring Track Views

Any Track Views that you open are stored in a buffer and saved with your scene file. You can selectively open and close these stored Track Views at any time.

You can enter a name for a Track View in the field on the top right of the Track View toolbar.

To restore closed Track View windows, do one of the following:

- Click Track View on the toolbar to reopen the most recently closed Track View in a window.
- On the Track View menu, choose the name of a Track View to open.

The Track View storage buffer holds 13 Track Views and saves the buffer with your scene file. Once you have created 13 Track Views, you must delete one before creating another.

Defining Track View Terms

Track View is a data-driven hierarchical display of your scene and animation. Any description of how Track View works uses terms shared with other chapters throughout this guide, as well as terms unique to Track View.

The following definitions are some of the terms unique to a discussion of Track View.

Items—Everything in your scene displays as an item in the Track View Hierarchy list. This includes objects, transforms, modifiers, materials, and parameters. What you can do with an item varies according to the item's type.

Controllers—The only type of item that contains animation values is a controller. Controllers manage all animated parameters in your scene. Some controllers manage single values, while others manage expressions or lists of multiple controllers. If you can assign a value to an item, and animate that value, you are using a controller.

Hierarchy—The Track View hierarchy follows the traditional example of organizational headings in an outline. The highest levels of the hierarchy represent the main groupings of Sound, Global Tracks, Video Post Environment, Materials, and Objects. Subordinate levels of the hierarchy show details of your scene, such as material and object parameters.

The Track View hierarchy is described as having superior levels with subordinate levels and branches below them.

- Superior levels group subordinate levels.
- Subordinate levels display below and indented to the right of their superior level.

Branch—An item, and all the subordinate items below it represent a branch of the Track View Hierarchy.

Tracks—Every item in the hierarchy has a track that displays what happens to the item over time. There are two types of tracks:

- *Range tracks* indicate when items below the track are animated. A range bar in the track displays the range of time over which the animation occurs.
- *Animation tracks* contain the actual animated values for an item. Only controller items have an animation track.

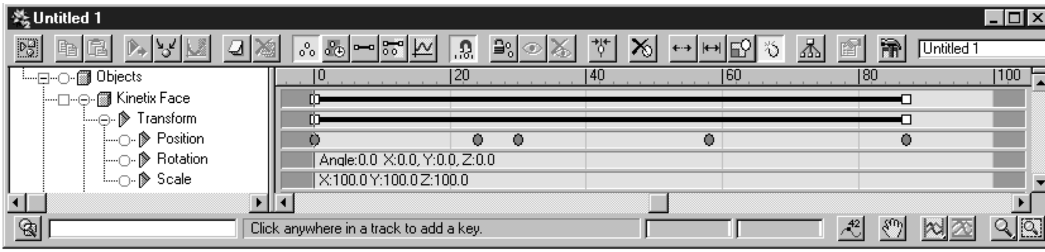
The values in an animation track are usually displayed as keys. Some controllers don't use keys and instead display their values as a range bar or some other graphic symbol. For example, the Wave Form item displays a sound file as a two-channel sound wave.

Keys—Whenever you change a parameter with the Animate button on, you create a key at the current time in that parameter's track. The key contains the parameter value at that time as well as information about how to interpolate values between it and the previous and following keys.

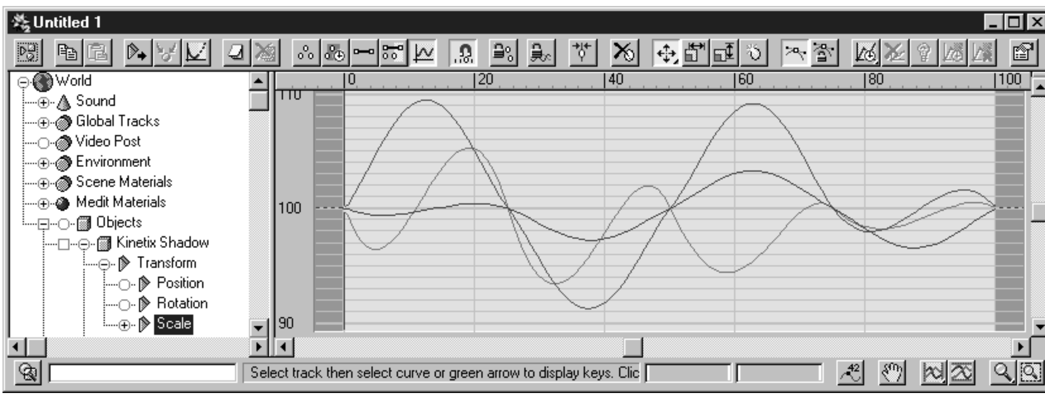
Keys are normally displayed in their track as dots, but they can also be displayed as ranges and vertices on a function curve.

Range Bars—When animation keys have been created, range bars appear to indicate the range of time over which the animation occurs. Ranges always appear in Range tracks, but in some display modes and certain controllers, Animation tracks also display ranges.

Function Curves—You use Track View to chart the values of keys and the interpolated values between keys as a curve. These curves fully express how a parameter varies over time. Only Animation tracks can display function curves.



Range and Animation Tracks



Function curves in Track View

Frames—3D Studio MAX generates animation by rendering multiple still images called *frames*. Playing back the animation displays those frames in rapid succession. The frame is therefore the basic method for measuring output and time in 3DS MAX.

Ticks—The foundation of all time measurement in 3DS MAX is the tick. There are 4800 ticks in a second and because 4800 can be divided evenly by all the standard frame rates, you can change from film to video frame rates without altering the timing of your animation.

Tangent types—You can specify tangent type options for certain controllers to adjust how values are interpolated between keys. The easiest way to work with Tangent Types is in Function Curves mode. See chapter 26, “Function Curve and Trajectory Editing.”

Out-of-Range types—When you specify values and keys for a controller, you are defining animation over a range of time. You choose Out-of-Range Types to determine how the animation continues outside a specified range. The easiest way to work with Out-of-Range Types is in the Track View Function Curves mode. See chapter 26, “Function Curve and Trajectory Editing.”

Using the Hierarchy List

The Track View Hierarchy list, shows everything in your scene and all the animation parameters. The list is organized in a scene hierarchy that shows the relationship between various items.

You use the Hierarchy list for multiple tasks:

- Selecting items for Track View commands.
- Selecting object icons for transform and modify operations. Selecting an object's icon in Track View also selects it in the scene.
- Selecting materials in the Material Editor. Selecting a material's icon in the Medit Materials branch makes it the active material in the Material Editor.
- Analyzing object links. The Hierarchy list in Track View accurately reflects linked object hierarchies. You can use the Hierarchy list to examine linked tree structures.

Reading the Scene Hierarchy

At the root of the scene hierarchy is the World. This item collects all tracks in your scene as a single track for quick global operations.

By default, the World track shows the tracks of all branches except the Objects branch. Selecting Modify Subtree causes the World track to include the Objects branch as well.

Note: The default location of the time ruler covers the World animation track. Move the time ruler to see the World track.

The Sound branch contains items for loading and synchronizing a single sound file.

The Global Tracks branch contains lists where you can store controllers, much as you might store global variables. Using expression controllers, you can point to a controller in Global Tracks from several other tracks.

The Video Post branch contains items for Video Post filters with animated parameters. This branch is empty until you assign a Video Post filter that uses animated parameters.

The Environment branch contains items that control the background and scene environment effects. Examples of such items include, Ambient Light, Background definition, Fogs, and Volumetric Lighting.

The Medit Materials branch contains material definitions in the Materials Editor.

The Scene Materials branch contains the definitions for all materials in the scene. It is empty until you begin assigning materials to objects. When you select materials in this branch, you are working with instances of materials assigned to objects in your scene. These materials might not be in any of the Material Editor samples.

Each material has its own hierarchy of Parameters, Maps, and Sub-materials. See chapters 20 and 21 for information about defining complex material hierarchies.

The Objects branch contains a dual hierarchy for all the objects in your scene.

The first hierarchy is represented by the square icon. It displays the structure of linked objects in your scene. Child objects display as subordinate levels, below and indented to the right, of their ancestors.

The second hierarchy in the Objects branch is represented by the circle icon. It displays the parameter hierarchy below each object.

Identifying Item Categories

Each type of item in the Hierarchy list is represented by an icon. You can use these icons to quickly identify what each item represents.

Sound—Green cones indicate sound parameters. Track View has only one sound source.

Material—Blue spheres indicate material definitions. Branches below a material definition are part of that material. Because a material can be composed of multiple materials, it is possible to have nested material definitions. Blue spheres also appear in an object's branch when a material is assigned to an object.

Map—Green parallelograms indicate map definitions. Branches below a map definition are part of that map. This includes values used by parametric maps and other map definitions that are part of a map tree.

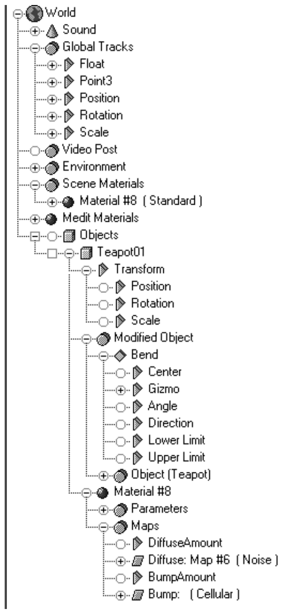
Object—Yellow cubes indicate objects in the scene. Branches below an object contain linked descendents and the object parameter hierarchy.

Container—Blue cylinders indicate a container of other items. Blue cylinders are how 3D Studio MAX organizes complex items in the Hierarchy.

Modifier—Orange diamonds indicate modifiers and Space Warp bindings. Branches below a modifier contain the modifier's sub-objects and parameters.

Controller—Green triangles indicate controllers, the animation workhorses of Track View. They contain the animation values for parameters and are the only item in the Hierarchy list that can have a track containing keys.

Certain types of controllers can contain other controllers. Examples of these are Transform Controllers and List Controllers. See chapter 27, "Working with Controllers."



Icons in the Hierarchy list

Expanding, Collapsing, and Filtering the Hierarchy

Expanding and collapsing the branches of the hierarchy is a valuable technique when using the Hierarchy list. The hierarchy can become quite complex and the ability to collapse branches you are not currently focused on makes working with the hierarchy much easier.

You can also set display filters to prevent the display of certain items even when a branch is expanded. Filter settings are stored with each Track View so it is useful to set up multiple Track Views with your most commonly used filters.

Clicking Branch Expansion Icons

To the left of every item icon are one or two branch expansion icons. You click these icons to expand or collapse branches one level beneath an item.

Circle icons—Expand and collapse tracks for parameters and modifiers below the item.

Square icons—Appear next to object items. Expand and collapse linked children of an object.

Choosing Shortcut Menu Commands

You can right-click an item in the hierarchy to choose commands on the shortcut menu to expand and collapse all branches below an item. The number of expand and collapse choices in the menu vary depending on the type of item you click.

Filtering the Hierarchy

You can filter the display of the Hierarchy list to show only certain types of items.

Click Filters to display the Filters dialog. Use this dialog when you want to change settings for multiple filters.

Right-click Filters to display a shortcut menu containing commands for many of the settings in the Filters dialog. Use this menu to change filters quickly.

Using the Filters Dialog

Use the Filters dialog to set multiple filters and apply them to the Track View. Controls in the dialog are grouped into the following areas:

Show—Filters items by type and level of detail.

Show Only—Filters items based on their animation and selection state.

Hide By Controller Type—Filters items by the name of the controller assigned to the items.

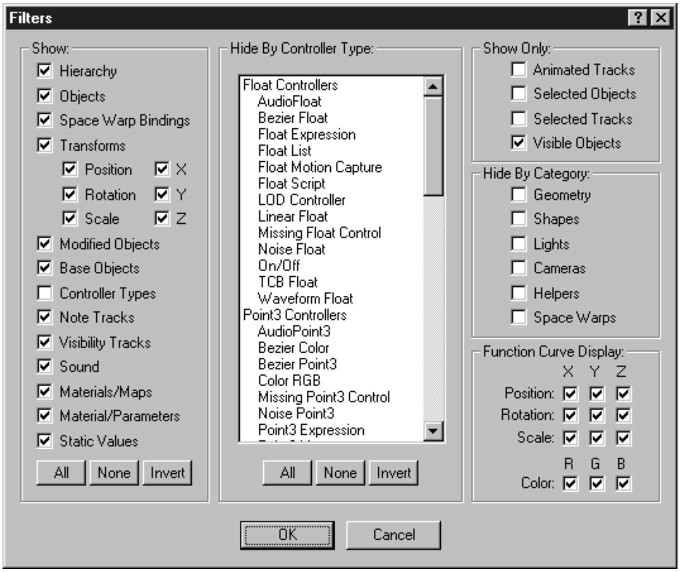
Hide By Category—Filters items by categories similar to those found on the Display panel.

Function Curve Display—Filters the display of function curves. See chapter 26, "Function Curve and Trajectory Editing."

Using the Filters Shortcut Menu

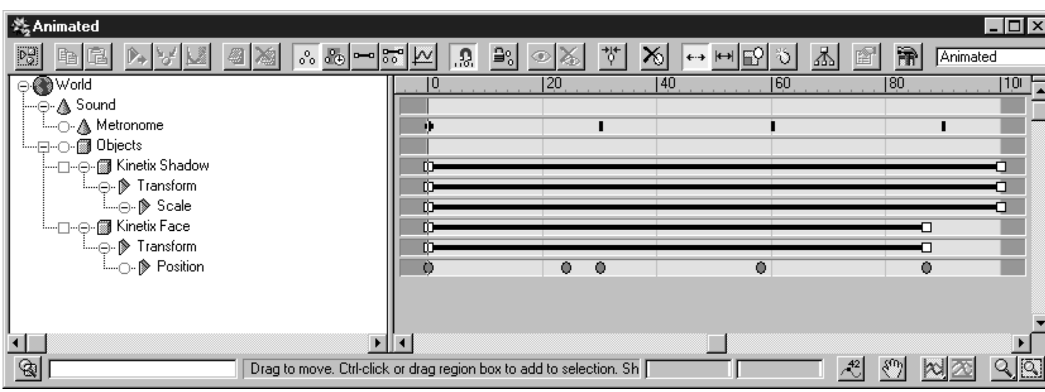
This menu is divided into three sections with choices similar to the Filters dialog.

- The top section contains the Show Only choices similar to the Filters dialog.
- The middle section contains macro choices setting useful filter combinations in the Show area of the Filters dialog.
- The bottom section contains a subset of filters from the Show area of the Filters dialog.



Filters dialog

Filters shortcut menu



Example of selecting filter, Show Only Animated Tracks

Selecting Items in the Hierarchy List

You select items in the Track View Hierarchy List to do the following:

- Select item labels for Track View operations.
- Select object icons to select objects in the scene.
- Select material or map icons when working with the Material Editor.
- Select object modifier icons to navigate the Modifier Stack on the Modify panel.

Selecting Item Labels for Track View Operations

When selecting items from the Hierarchy list for use with Track View operations you usually click the label of the item. Other options for selecting item labels include:

- Right-click an item label to display a shortcut menu with commands for selecting all items, or clearing, or inverting the selection.
- Type a search string in the field at the left of the status line to select item labels by name. You can use wild cards in the string.

Selecting Objects and Materials

You can select objects and materials for non-Track View operations by clicking the object or material icon.

When objects are selected in the scene, their icons are also highlighted in Track View. Selecting object icons in Track View follows the same rules as selecting objects in the scene:

- You cannot select the icon of a hidden object.
- You cannot select the icon of a frozen object.
- If you select the icon of an object that is part of a closed group, the icons for all objects in the group plus the group's icon are selected.

Unlike selecting objects, you can select only one material at a time from the Medit Materials branch. The selected material or map becomes the active item in the Material Editor dialog.

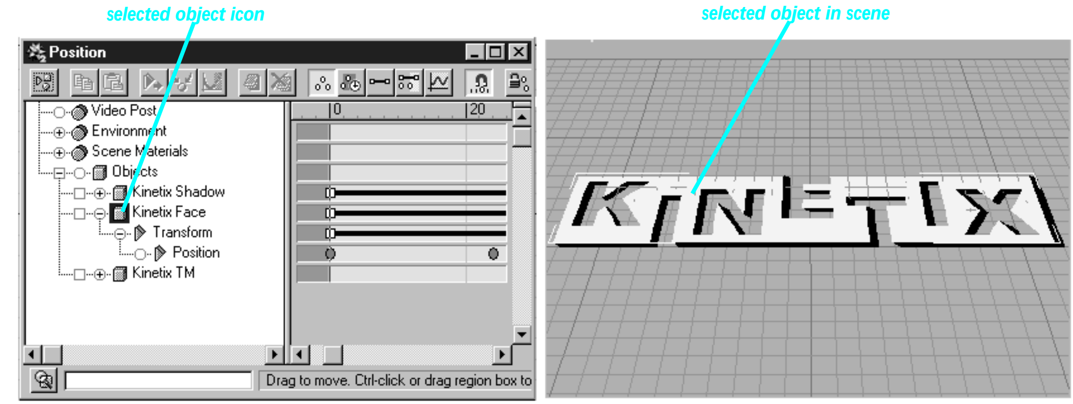
Navigating the Modifier Stack

You can navigate to different levels in the Modifier Stack by selecting icons in the Objects branch of the Hierarchy List. This way you can easily navigate the Modifier Stack using the Track View hierarchy.

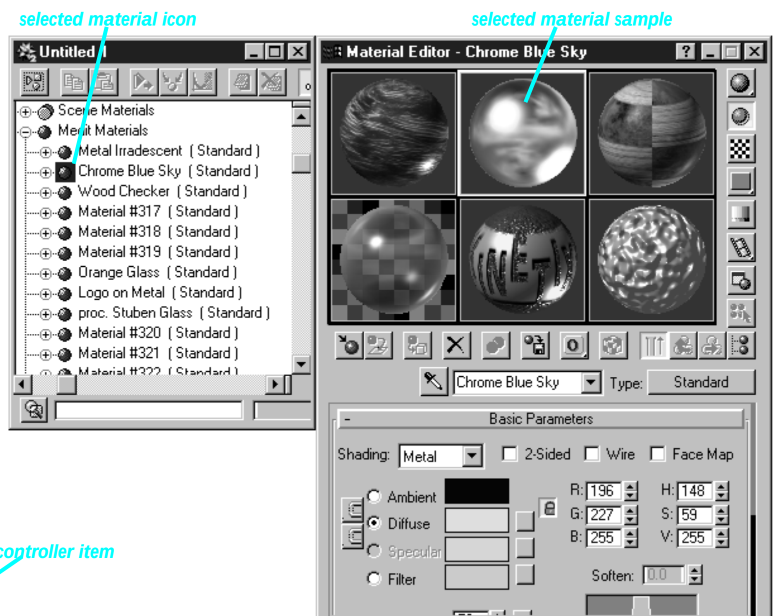
Clicking icons when the Modify panel is open, does the following:

- Click an orange modifier diamond to display parameters for that modifier.
- Click a blue object cylinder to display the Base Parameters of that object.
- Click a yellow object cube to display the top level of the Modifier Stack for that object.

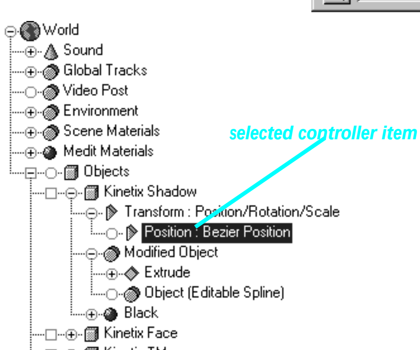
This system is particularly useful for navigating compound objects—such as Boolean and Morph hierarchies.



Selecting objects in the scene and the Hierarchy list



Selecting maps and materials in the Hierarchy list



Selecting item labels in the Hierarchy list

Choosing Edit Modes

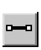
The edit window is where you change the values and timing of your animation. To make changes to your animation, you choose one of the Track View Edit modes.

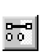
Choosing Track View Edit Modes

You choose an Edit mode by clicking one of the Edit mode buttons on the Track View toolbar. Each mode has its own uses; you will probably spend most of your time in the Edit Keys and Function Curve modes.

 **Edit Keys—**Displays your animation as a series of key dots and range bars. Only controller items (items with a green triangle icon) can display key dots. All other items display range bars indicating the time span of key dots for controller items in their subordinate branches.

Edit Keys is most useful for getting a global view of your animation because it displays animation timing for all tracks. Use this mode for key and range editing when you want to view your changes in the context of the total animation.

 **Edit Ranges—**Displays all tracks as range bars. This mode is useful for quickly scaling and sliding complete animation tracks. This is the only mode where you can use range bars to affect linked object hierarchies. You cannot access individual key values in this mode.

 **Position Ranges—**Displays a hybrid of Edit Keys and Edit Ranges— it only displays key tracks, and superimposes range bars over the key dots. Use this mode to change range bars independent of their associated keys.

You use Position Ranges to set up looping effects, padding the ends of a range with extra frames, or placing a key outside of the range.

 **Edit Time—**Displays key dots and range bars in the background. You use this mode to select and manipulate blocks of time.

 **Function Curves—**Displays selected controller items as curves, charting the controller's change of value over time. Noncontroller items cannot be displayed in Function Curves mode.

Function Curves mode is useful for examining the behavior of a controller on every frame. Use Function Curves mode for detailed key editing and for comparing animation values between a few selected tracks. Details about function curve editing are presented in chapter 26, "Function Curve and Trajectory Editing."

Identifying Your Place in Time

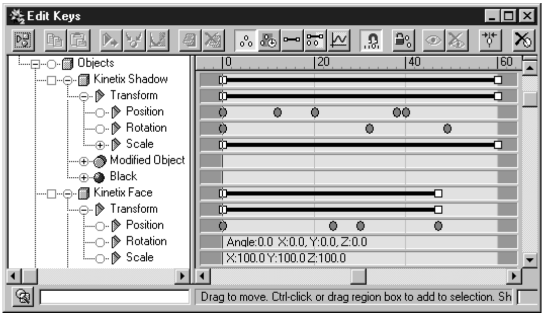
With all the different ways of viewing your animation values, it might become difficult to tell when in time you are working. You can use different tools to help you measure and identify time.

Time ruler—Drag the time ruler in the edit window to position it near any track. Markings on the time ruler reflect the settings in the Time Configuration dialog.

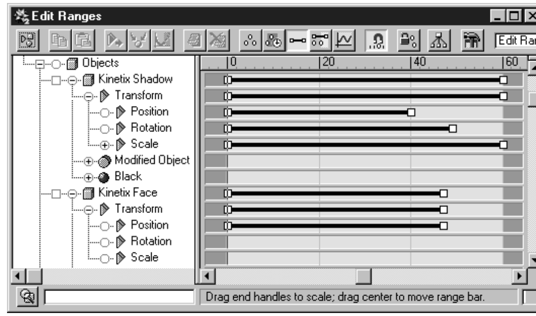
Current time line—The current time set by the time slider is shown as a dark vertical line running through the edit window.

Time cursor—When you drag a key or range bar, one or two yellow vertical lines appear centered on the key you are dragging or at both ends of a range bar. Use this cursor is to line up your edits with markings on the time ruler.

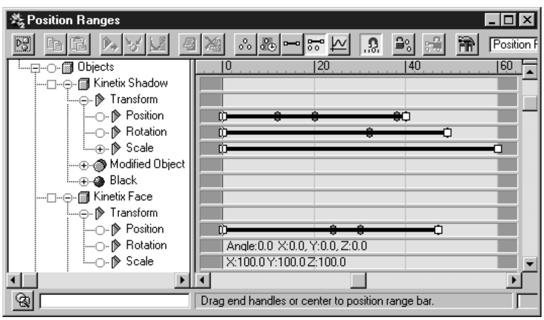
Time Status fields—When you select a key, its time is displayed in the Time Status field. This field is to the right of the prompt line.



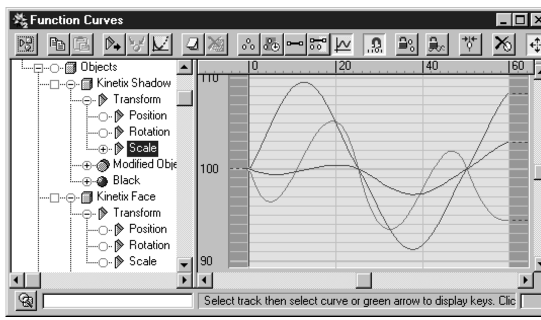
Track View Edit Keys display



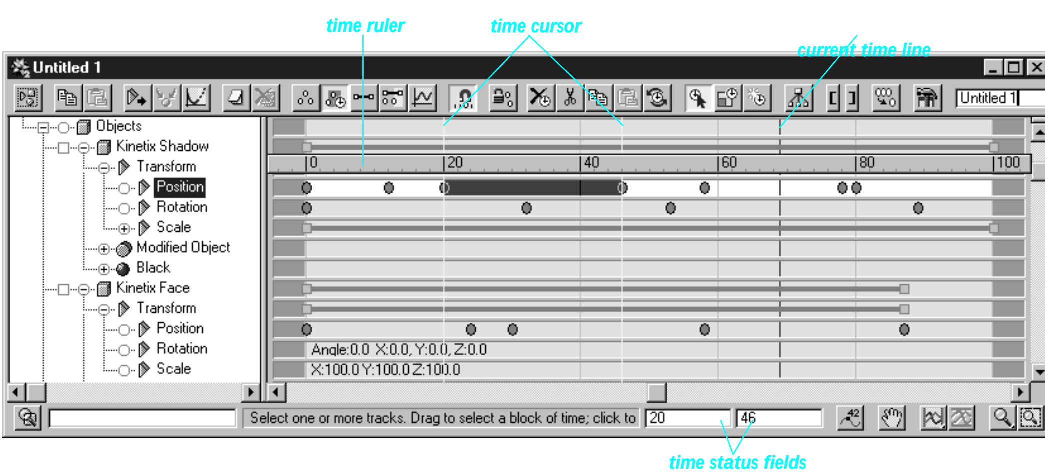
Track View Edit Ranges display



Track View Position Ranges display



Track View Function Curve display



Identifying Time Position indicators in Edit Time mode

Navigating Track View

Track View has its own set of navigation buttons. Use these buttons to scale your view of time and value in the edit window.

All modes respond to horizontal (time) scaling. Function Curves is the only mode that responds to vertical (value) scaling.

 **Zoom Selected Object—**Scrolls the hierarchy list, and edit window, to place the selected object at the top of the viewable area.

 **Zoom Horizontal Extents—**Scales your view horizontally to display the Active Time Segment.

 **Zoom Horizontal Extents Keys—**Scales your view horizontally to display the full range of all animation keys. Depending on your animation this view could be larger or smaller than the Active Time Segment.

 **Zoom Value Extents—**Scales your view vertically to display the full height of function curves.

 **Pan—**Drag your view in time and value.

 **Zoom Time—**Drag horizontally to zoom your view of time.

 **Zoom Values—**Drag vertically to zoom your view of animation values. This button is only available in Function Curves mode.

 **Zoom—**Drag vertically to simultaneously zoom your view of time and value. This button behaves like Zoom Time in all modes except Function Curves mode.

 **Zoom Region—**Drag a region in the edit window to scale that region to fill the window.

 **Min/Max toggle—**Only appears when Track View is displayed in a viewport. Click it to toggle the Track View viewport between filling the 3DS MAX window and its normal size.

Using the Sound Track

You can synchronize your animation to a sound file using the 3D Studio MAX sound track. The Sound track contains two sound controllers:

- *Metronome* generates a series of beeps using your computer's system speaker.
- *Wave Form* displays a sound file that plays on your installed sound card. The Wave Form item does not appear in the Hierarchy list until after you have selected a sound file.

You set the Metronome and the Wave Form parameters by selecting the Sound track and clicking Properties to display the Sound Options dialog.

Setting Up the Metronome

The Metronome track produces a regular beat using two tones. You specify settings in the Sound Options dialog to enable the Metronome and set its speed.

Selecting a Wave Form

The Wave Form track displays a selected sound file as a dual wave form. The blue wave form is the left sound channel and the red wave form is the right sound channel.

You click Wave Form controls in the Sound Options dialog to select, remove, and reload sound files.

Positioning a Wave Form

Once you have a wave form displayed in Track View, it cycles continuously. If the sound is shorter than your active time segment, it keeps repeating to the end. The original position and length of the wave form displays in bright red and blue. Repeating portions of the wave form display in dark red and blue.

You can change the position of the wave form in time by sliding the range bar in the Sound track.

Playing a Wave Form

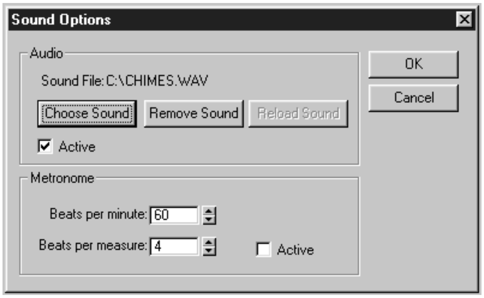
If you enable Real Time Playback in the Time Configuration dialog, you can hear the Metronome beats, the Wave Form, or both when you click Play in the playback buttons.

Scrubbing a Wave Form

You can listen to just part of a sound by *scrubbing* the time slider over a portion of the wave form. Scrubbing is the process of repeatedly dragging the time slider forward and back over a short range of frames. 3DS MAX plays the sound as you drag the slider forward. This technique is particularly useful for zeroing in when a specific sound occurs.

Scrubbing does not work with metronome beats.

Display of sound items in Track View



Sound Options dialog

Hiding Objects with Visibility Tracks

You can use functions in the Display panel to hide objects and prevent them from rendering. If you want to animate the appearance and disappearance of an object you use a *visibility* track.

As the name implies, a visibility track controls when you can see an object. Visibility tracks do not exist until you add them to objects using Edit Keys mode of Track View.

To add a visibility track to objects:

1. Select one or more object item labels.

2.  On the Track View toolbar, click Add Visibility Track. Controller items, labeled Visibility, and their tracks appear directly below the selected object items.

Setting Visibility Keys

Unlike most animation in 3D Studio MAX, you cannot animate visibility by turning on the Animate button and manipulating objects in your scene. All visibility values are set in Track View by adding keys and adjusting key values.

To add visibility keys:

-  On the Track View toolbar, click Add Keys, then click in a visibility track to add a key at that time location.

By default, Visibility keys use the On/Off controller. The first key you create is an Off key. Additional keys are set to the opposite of the key that precedes them. You can easily tell the on/off state of an object by the following:

- The Visibility track is shaded blue where the object is visible.
- Invisible objects display with screen door transparency in shaded viewports.

Fading Objects In and Out

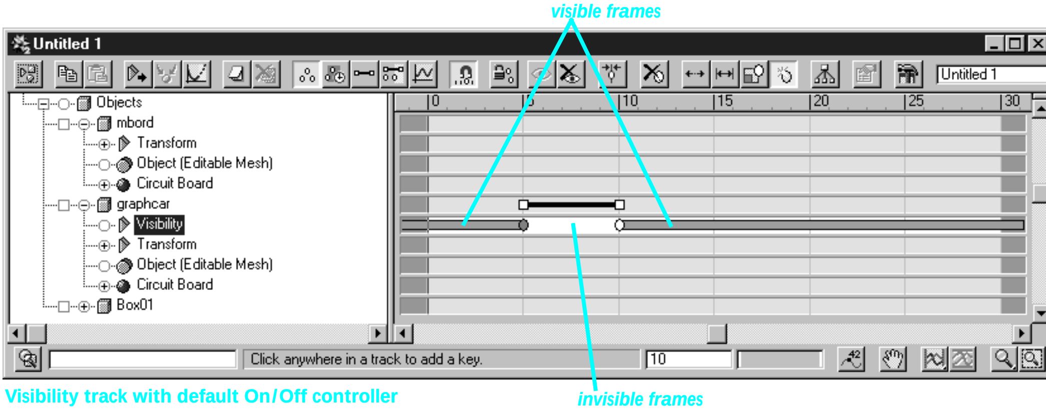
You can create variable visibility by replacing the default On/Off controller with another controller—such as a Bezier Float controller.

- Objects are invisible at a value of zero.
- They grow increasingly visible at values greater than zero.
- They achieve full opacity at a value of 1.0.

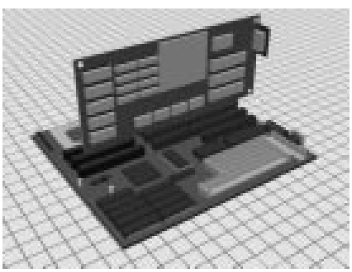
Inheriting Visibility from a Parent

When you assign a Visibility track to a parent object, all the descendents of that object inherit the visibility. This also works when assigning Visibility tracks to groups because the group is a parent to the objects it contains.

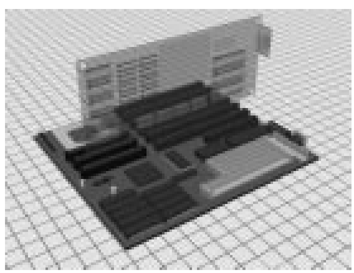
You can prevent some objects in a hierarchy or group from inheriting visibility, by clearing the Inherit Visibility property for those objects. Right-click an object and choose Properties to set object properties.



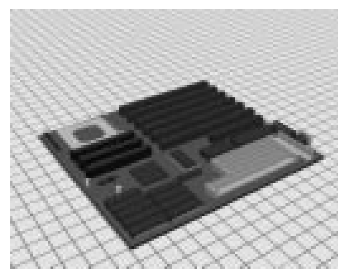
Visibility track with default On/Off controller



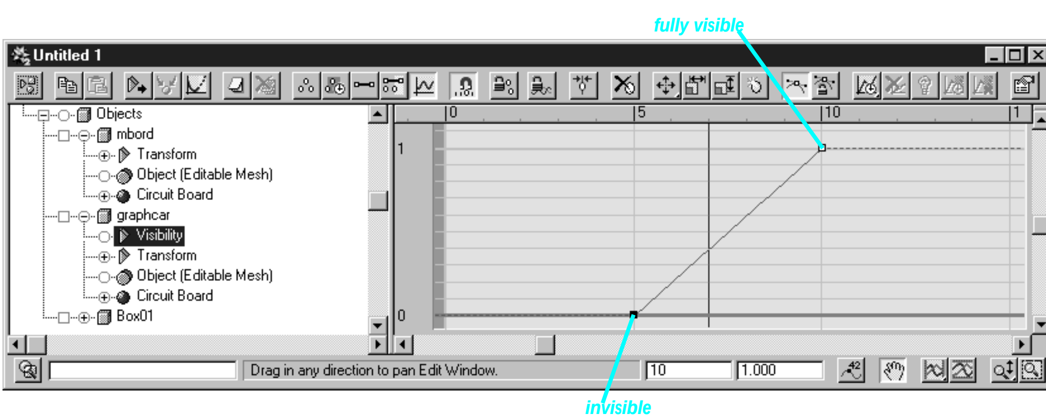
Circuit board visible at frame 0



Circuit board with Bezier invisibility partially visible at frame 7



Circuit board invisible at frame 7

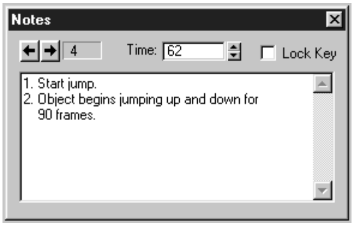


Function curve of visibility track with Bezier Float controller

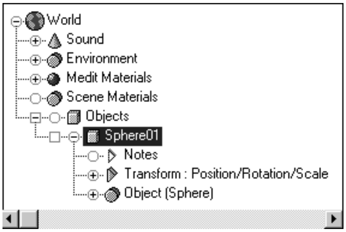
Adding Note Tracks

Successfully producing any but the simplest model or animation requires that you take careful notes about everything you do. You can conveniently keep those notes inside your 3D Studio MAX file by using Notes tracks.

Any item in the Hierarchy list can have a Notes track added as a branch below it. You can add a Note track to an item in any of the Track View modes but the most convenient mode to work with Notes is the Edit Keys mode.



Editing a note in the Notes dialog



Note track added to an item

To add a Note track to items:

1. Select one or more item labels.

2.  On the Track View toolbar, click Add Note Track.

Note tracks are inserted as branches directly below the selected items. The icon for a note track is a yellow triangle, indicating that it is a special type of nonanimated controller.

Writing Notes

Once you have created a Note track for an item, you insert note keys using Add Keys, and edit the notes using the Notes dialog box.

To add a Note key to a Note track:

-  Click Add Keys, then click in a Note track to place a note at that location.

To edit notes:

- Right-click a Note key to display the Notes dialog, then enter your note.

The first line of your note appears as a label to the right of the note key. You should type a short one or two word description as the first line of your note and then press ENTER to add the body of the note.

Storing Controllers in Global Tracks

You can store controllers, much as you might store global variables in the Global Tracks branch.

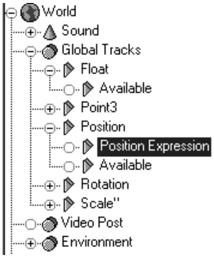
- Use expression controllers assigned to other items to reference a controller in the Global Tracks branch. Changing the controller in the Global Tracks branch affects the controllers that reference it.
- Use the Global Tracks as a storage place for copying and pasting a tracks.
- Paste a Global Tracks controller as an instance, and it becomes a central controlling place for all the instanced controllers.

Global Tracks contains List controllers for the different controller types.

To assign a controller in Global Tracks do the following:

1. Expand the Global Tracks branch, and then expand one of the List tracks.
2. Select the Available track, and then click Assign Controller.
3. On the Assign Controller dialog, choose the type of controller you want and click OK.

Once the controller is assigned, you can point to it from an expression controller assigned to any other track, or you can copy it to the Track View buffer, and then paste it to any number of matching controller types.



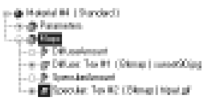
Copying and Pasting Items

You can copy and paste geometry, materials, and animation controllers between items in the Track View Hierarchy List. You can copy two categories of items in Track View:

Containers—Items with multiple branches that completely define something in your scene. Container items that can be copied include:

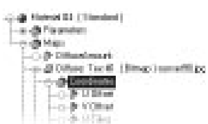
- Material Parameters containing the Basic Parameters for a material definition.

- Material Maps containing the entire set of maps and map parameters assigned to a material.

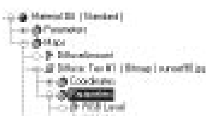


- Map definitions containing a single map type with its associated parameters and coordinates.

- Map Coordinates containing the map *XYZ* and *UVW* coordinate offset, tiling, and angle settings.



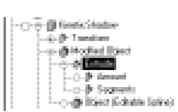
- Map Parameters containing the parameters for a specific map type.



- Objects, one level below a named object item, containing creation parameters for an unmodified object.

- Modified Object containing modifiers applied to an object and the object creation parameters.

- Modifiers containing modifier parameters.



Controllers—These are items with a green triangle icon. They control actual animated values for each parameter. Keep in mind that when you copy containers, you are actually copying and pasting groups of related controllers. Details about copying and pasting single controllers are presented in chapter 27, "Working with Controllers."

Whether you copy Objects, Materials, or Controllers, the same basic principles apply. Here are restrictions for using Copy and Paste:

- Only a single selected item can be copied.
- A copied item can only be pasted into another item of the same type. An exception to this restriction involves pasting Object and Modified Object containers.
- A copied item can be pasted into a selection of multiple items only if all the items are of the same type.
- When pasting items you can choose to make an instance or a copy of the pasted item.

Copying Items

You copy an item by selecting it in the Hierarchy list and then clicking Copy the on the Track View toolbar to copy it to the clipboard. If Copy is available, the selected item is a valid copy source. 3DS MAX disables Copy when the selected item is not a valid copy source or if multiple items are selected.

Pasting Items

Pasting items involves a few more choices than copying. You select one or more items from the Hierarchy list. If Paste is available the selected items are valid paste targets. 3DS MAX disables Paste if the selected targets are not all the same type or if they do not match the type of item in the clipboard.

You can use Paste as follows:

- Paste into a single target item.
- Paste into multiple target items.
- Paste copies or instances.
- Choose to convert other instances in the scene automatically.

Clicking the Paste button displays the Paste dialog, with controls for determining how the Paste operation will be carried out.

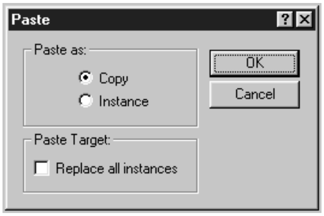
Copy—Pastes the item in the clipboard *as an independent copy*. The target item will have no connection to the copied source item.

Instance—Pastes the item in the clipboard *as an instance of the copied source item*. The target item will be an instance of the source item. Any change you make to either item affects the other.

Replace All Instances—Controls whether existing instances of the target item are also converted to the paste source or left as they are.

Making Instances Unique

You can convert instanced items to unique items by clicking Make Unique on the Track View toolbar. If the selected item is not an instance, or if a selection of multiple items does not contain similar instances, Make Unique is disabled.



Paste dialog

Copying and Pasting Objects

Copying and pasting objects involves working with the item that represents the object at the top of the Modifier Stack. This item is always a branch immediately below the named object item. The name of the item changes based on whether or not you have applied modifiers to the object:

Object—If you have not applied any modifiers to the object, then the Object container is at the top of the Modifier Stack.

Modified Object—After you apply a modifier to an object, the Modified Object container is at the top of the stack. The Object container moves to the bottom of the Modifier Stack inside the Modified Object container.

Using these two container types you can copy geometry between objects.

To copy one object to other objects:

1. Expand a named object item.

2. Select the Object or Modified Object item one level below the named object.

3.  Click Copy in the Track View toolbar.

4. Select one or more Objects or Modified Objects one level below the named object.

5.  Click Paste.

6. Set the Paste options in the Paste dialog.

Here are some examples of how you might use object copy and paste:

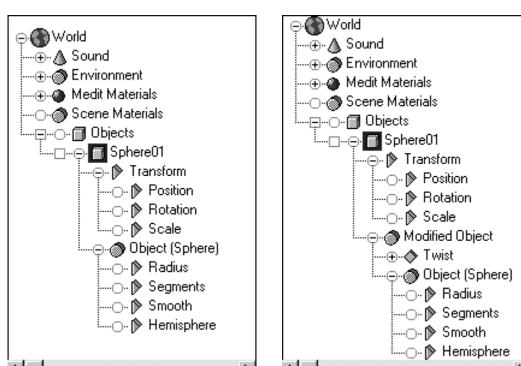
- Paste an Object into another Object to replace one simple object with another. For example, you animate the transforms of a box object and want to replace the box with a torus.
- Paste an Object into a Modified Object to replace a complex, modified object with a simple stand-in object. For example, you model a complicated vehicle and you want to replace it with a box while you animate its transforms. Be sure to save the Modified Object to a file (using Save Selected).
- Paste a Modified Object into another Modified Object to replace one complex modified object with another. For example, you have animated a flight of jets and you want to replace them with helicopters.
- Paste a Modified Object into an Object to replace simple stand-in objects with complex modified objects. For example, you have animated a box and now want to replace it with a complex model of a jet.

Copying and Pasting Object Modifiers

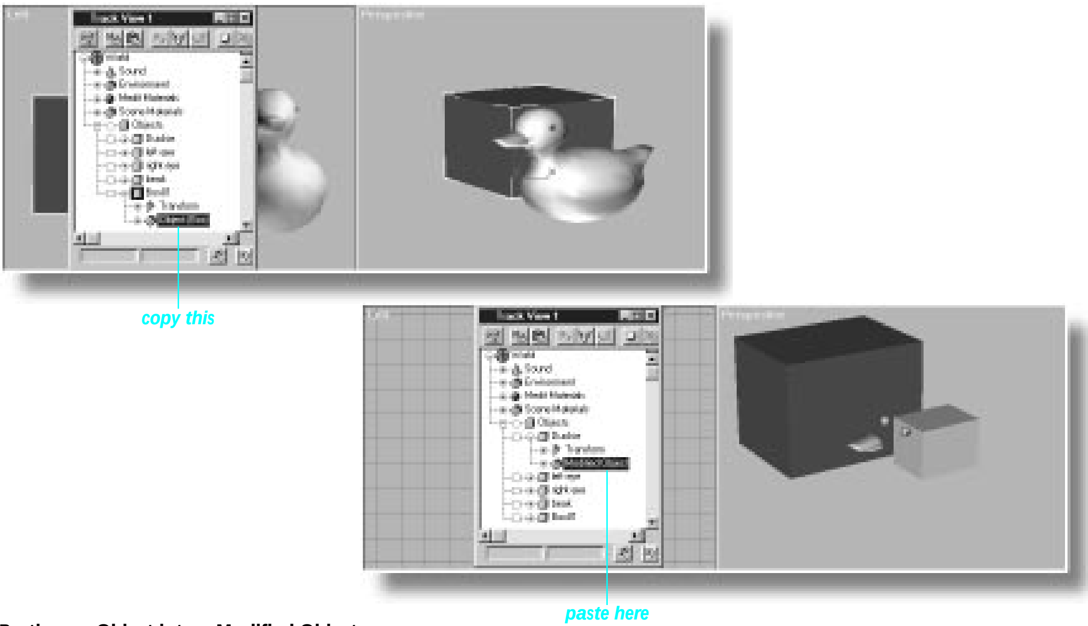
You can also copy and paste modifiers below the Modified Object container. Copy and paste modifiers to do the following:

- Copy modifiers within the Modifier Stack of a single object.
- Copy modifiers between objects.

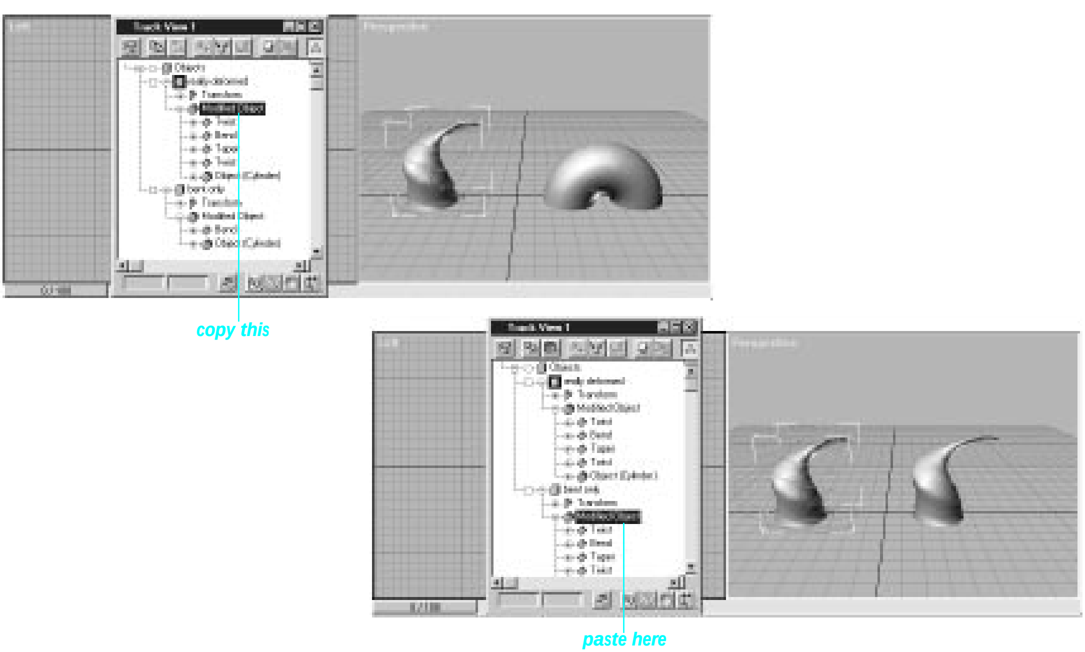
Unlike pasting other items, modifiers do not replace the selected item when you click Paste. Instead, the pasted modifier is inserted above the selected item.



Unmodified object compared to a modified object



Pasting an Object into a Modified Object



Pasting a Modified Object into another Modified Object

Making Instance and Reference Objects Unique

You can convert instance and reference objects into unique objects by clicking Make Unique on the Track View toolbar. The result of clicking Make Unique depends on how the object was instanced or referenced, and which object container you select in the Hierarchy list.

When you make an instance or reference object, you cause the data flow to branch on its way from the master object to two or more named objects.

- In Track View, these branches usually occur at Modified Object containers and sometimes at the base Object container.
- In the Modifier Stack, these branches display as a horizontal line, called a *derived object* line.

For descriptions of object data flow and instance objects, see the Appendix.

You can make an object unique by selecting a Modified Object or an Object container and clicking Make Unique in the Track View toolbar. Your result depends on whether or not the data flow branches at the selected container.

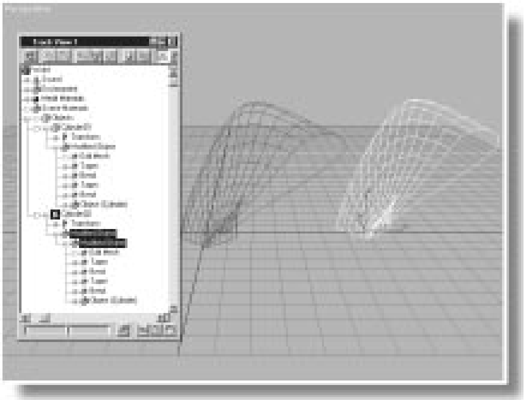
- If the data flow does not branch anywhere below the selected container, nothing happens.
- If the data flow does branch at or below the selected container, the data flow above the branch is split from the current data flow as a unique object. The data flow from the selected container to the master object is copied and attached to the new unique object.

The figures on the following page use Track View terms to show the dataflow of some instance objects and on reference objects and a possible result of using Make Unique on those objects.

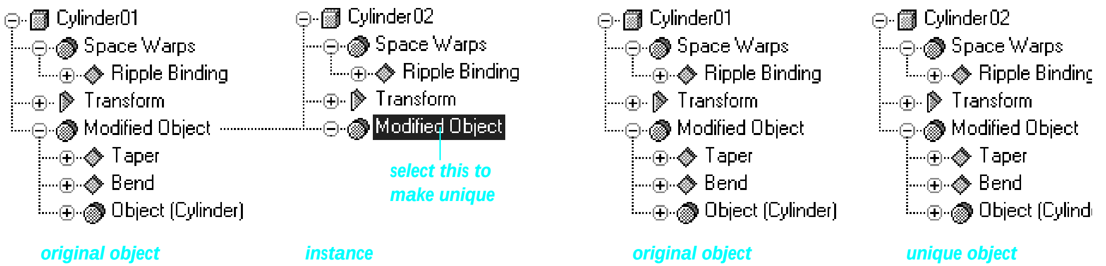
Making a Selection of Multiple Instances Unique

If you select multiple objects that are instances of each other, or share instanced modifiers, you can choose how to make them unique. After you click Make Unique, the Make Unique dialog appears with the question, “Do you want to make the selected items unique with respect to each other?”

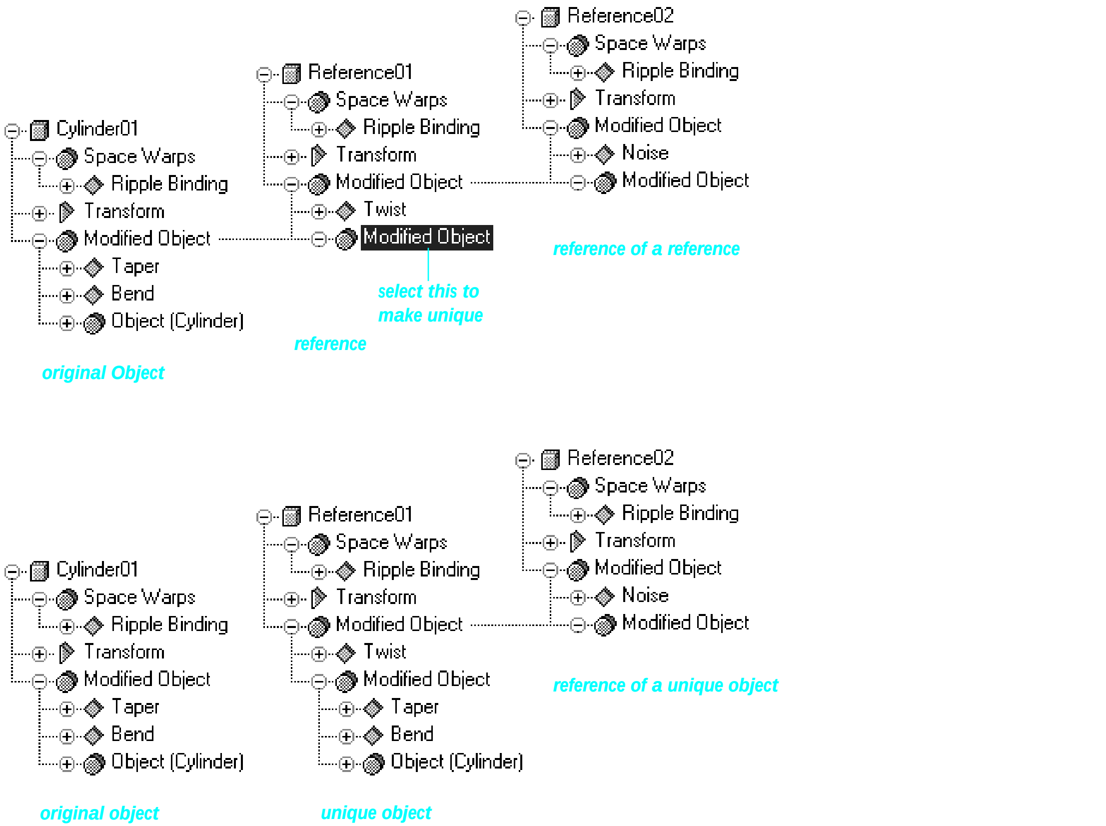
- Click Yes to make each object in the selection completely unique.
- Click No to leave the objects in the selection as instances, but to make them unique from other objects not in the selection.



Possible Make Unique points for a reference object

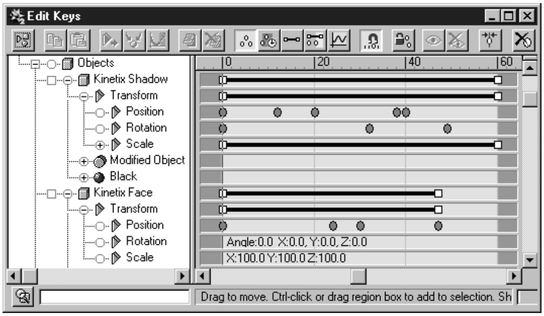


Making an instance unique

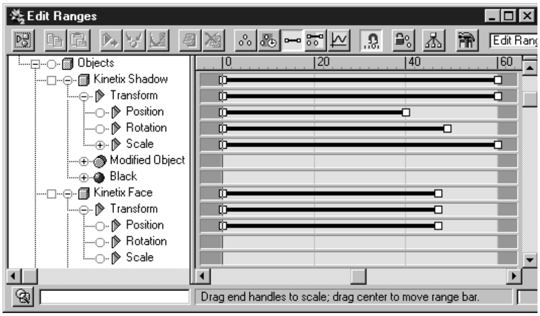


Making a reference unique

Editing Keys and Range Bars



Track View Edit Keys display



Track View Edit Ranges display

The traditional way of looking at animation is to break it down into a chart of the entire scene, divided into blocks of time, with animation keys and ranges of effects drawn on the chart. Traditional animators call this the "dope sheet" or "bar sheet." Four of the five Track View edit modes use this method.

These four modes are Edit Keys, Edit Ranges, Position Ranges, and Edit Time. You use these modes to change when an animated effect occurs. You use two modes in Track View to directly change the time location and time relationships of keys and range bars:

 **Edit Keys—**Displays key dots and range bars for most track types. Use Edit Keys mode to:

- Select and change one or more keys.
- Drag range bars to change all animation in multiple tracks.

 **Edit Ranges—**Displays all animation values as range bars. Use Edit Ranges mode to:

- Drag the range bar of an animation track to change all animation in that track.
- Drag range bars in higher-level tracks to change all animation in multiple tracks.

Understanding Key and Range Bar Timing

When you are changing keys and range bars, keep in mind that you are working with time. Horizontal distance in Track View represents time; you can measure time using the time ruler in the edit window.

- Dragging keys or range bars to the right moves them forward (later) in time.
- Dragging keys or range bars to the left moves them backwards (earlier) in time.

Velocity, or speed, is measured as the amount of change over time. Key values in Track View determine the amount of change from one key to the next. The distance between keys determines the amount of time it takes to change from one key to the next. Changing the distance between keys in your animation changes the velocity, or speed, of the animation.

- Shortening the distance between keys reduces the amount of time and speeds up the animation.
- Lengthening the distance between keys increases the amount of time and slows down the animation.

 Snapping to Frames

You have very precise control over your animation in 3D Studio MAX. Using Ticks you can locate keys using time increments as small as 1/4800th of a second.

Why would you require such precision when 3DS MAX always renders your animation to frames? Reasons for such fine precision include:

- Using *sub-frame* control, you can make subtle adjustments to your animation.
- Time critical animations used for legal or scientific presentations often require accurate placement of animation keys.
- Time critical animations used for legal or scientific presentations often require accurate placement of animation keys.
- You can scale your animations between large and small ranges of time with little chance of losing information.

There will still be times when you want to place a key exactly at a specific frame. Not a little before, not a little after, but right at a chosen frame. Use the Snap Frames mode to force keys to single frame increments in Track View.

To use Snap Frames in Track View:

-  Click Snap Frames to turn it on.

  The button remains highlighted to indicate you are in Snap Frames mode.

Turning Snap Frames on has the following effects:

- All key and range bar positions changed with Snap Frames on are forced to absolute frame increments. This includes selection sets of multiple keys.

  With Snap Frames on, each key in a selection set snaps to the nearest frame when the selection is moved or scaled.

- Snap Frames only affects operation in Track View. It has no effect on creating keys in the viewports or with the time slider.
- When using the Time Display Format of MM:SS:Ticks (which does not use frames), Snap Frames snaps to time values that match frame boundaries.

Selecting Keys and Range Bars

You select keys and range bars using the Move, Slide, or Scale buttons in the Edit Keys mode. Once you have made a selection it remains active and you can apply other functions to it.

Selecting Keys

You select keys in Edit Keys mode. Keys remain selected even if you switch to other modes.

You select keys as follows:

- Build selections of keys using the standard click and CTRL-click selection methods. Clicking keys is useful for selecting widely spaced keys scattered over multiple tracks.
- Select blocks of multiple keys using region-selection methods.

Selecting Keys by Time

To select keys within a specified start and end range of time, you can use the Track View utility, Select Keys By Time. Its main purpose is to select keys over a large range of time, when using the pointer might be awkward.

Click Track View Utilities on the Track View toolbar, and then choose Select Keys By Time. The Select Keys By Time dialog contains controls for specifying the start and end times of the selection range. See the online reference for details about this utility.

Selecting Subtree Keys

You can display all of the keys of subordinate tracks in their parent track by clicking Modify Subtree in Edit Keys mode. Parent tracks change from displaying a range bar to displaying keys matching the times of all keys in their subordinate tracks. You can then edit the keys in the parent track to change keys in all of the subordinate tracks.

The following Edit Key functions are restricted when Modify Subtree is turned on:

- Add Keys only works with controller tracks.
- Slide Keys is disabled.

Any changes you make to the keys in the parent track—cut, paste, move, delete—affect the subordinate keys as well. For example:

- Moving a key in a parent object track moves all keys on the same frame as the selected key in all tracks of all descendent objects.
- Deleting a key in the World track deletes all animation keys on that frame for all tracks in the scene—Sound, Materials, Objects, and so on.

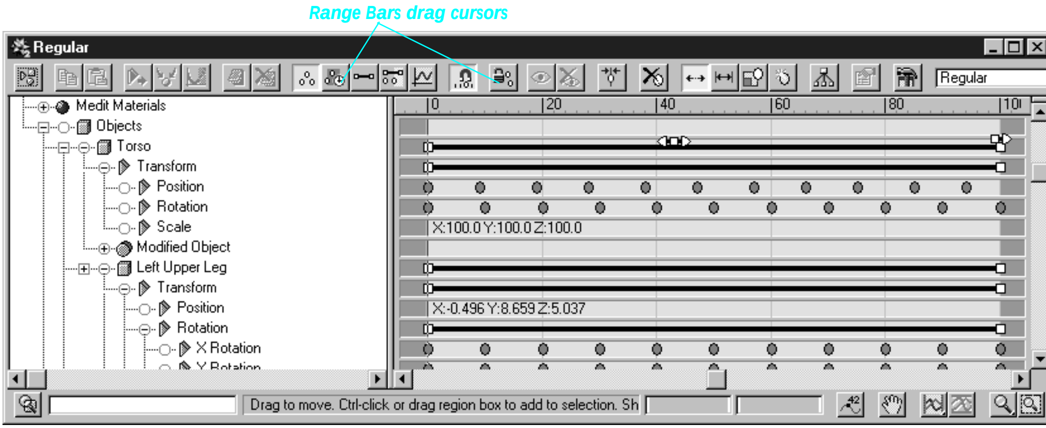
Selecting Range Bars

You don't actually select range bars in the traditional sense. A range bar is considered selected only while you are dragging it. As soon as you release the drag, the range bar is no longer selected.

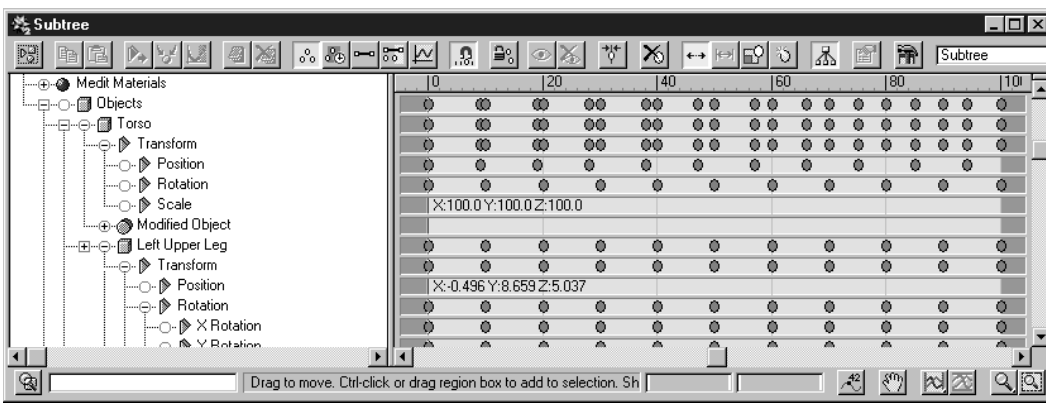
Dragging Ranges of Item Hierarchies

Dragging a range bar affects the animation of all non-object tracks subordinate to that range. For example:

- Dragging a named object range bar affects animation in all tracks subordinate to the object, including Space Warps, Transforms, and Modifiers. It does not affect any linked descendents or other objects in the scene.
- Dragging the World range bar affects animation in all of the tracks except tracks under the Objects branch.



Selecting range bars in Edit Keys mode



Subtree display in Edit Keys mode

Dragging Ranges of Object Hierarchies

If you want to change the range of an object and all of its descendent objects, click Modify Subtree in Edit Ranges mode.

With Modify Subtree on, a range bar appears in the Objects track. The Objects branch is the default parent of all named objects in the scene.

- Dragging a parent object range bar, with Modify Subtree on, affects all tracks subordinate to the object and all tracks of all of its linked descendents.
- Dragging the Objects range bar, with Modify Subtree on, affects all tracks of all objects in the scene.
- Dragging the World range bar, with Modify Subtree on, now affects all tracks in the scene, including Sounds, Materials, and all tracks of all objects.

Adding, Copying, and Deleting Keys

3D Studio MAX adds keys whenever you change the value of an animated parameter while the Animate button is on. You can also add and delete keys using buttons in Edit Keys mode and the Motion panel.

 All of the following Track View techniques are performed in Edit Keys mode.

 Adding Keys

Add Keys mode remains active until you activate another mode. While Add Keys is active, you click in any animation track to add a key at that location in time. The location where you click sets the time of the key as measured on the time ruler.

The value of the new key is set by one of the following conditions:

- Keys added before the first key of a track receive the same value as the former first key.
- Keys added between two keys receive an interpolated value based on the values of the original keys.
- Keys added after the last key in the track receive the same value as the former last key.

If you cannot add keys to a track, check the following conditions:

- Only animation tracks can accept keys. Animation tracks are identified as parameter items in the Hierarchy list with a controller icon (green triangle).
- The animation controller must be a type that uses keys. Not all controllers use keys. Examples of controllers that do not use keys include expression controllers, list controllers, and parametric controllers such as Noise.

Cloning Keys

You can also add keys to your animation by cloning existing keys. Press SHIFT while using either of the Move or Scale Keys buttons to make copies of the selected keys.

You can select keys across multiple tracks and use SHIFT to clone the keys within their own tracks. You cannot use SHIFT to clone keys from one track to another.

Copies of the selected keys are added at the new location. The value of the added keys is the same as the source keys.

 Deleting Keys

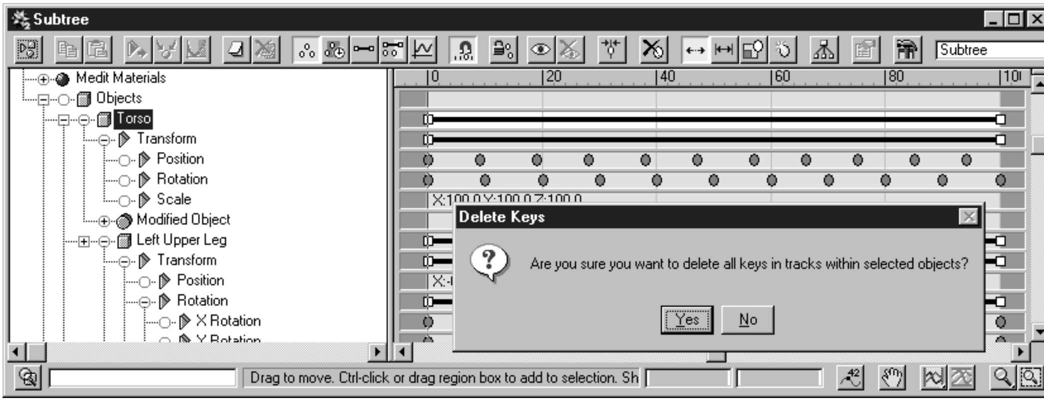
You delete keys by first selecting keys and then clicking Delete Keys.

If you delete all of the keys in a track, the value for the parameter associated with the track is set to the value the parameter had at the current time.

You can also delete all of the keys in all tracks associated with an object using the keyboard DELETE key.

To delete all keys for a selected object:

1. Click the object name label of the object from which you want to delete all keys.
2. Press DELETE, then click OK in the Delete Keys dialog.



Deleting all keys for an object

Using the Motion Panel to Add and Delete Transform Keys

You can also use the Motion panel to add and delete keys from the transform tracks of Position, Rotate, and Scale.

The PRS Parameters rollout contains buttons for creating and deleting Position, Rotation, and Scale keys. You add or delete keys by dragging the time slider to a specific time and clicking a transform button in either the Create Key or Delete Key area of the rollout.

Adding and Cloning Keys Using the Time Slider

You can also add and clone transform keys for a selected object using the time slider. See chapter 22, “Animation Concepts and Methods.”

To add and copy keys using the time slider:

1. Select an object to receive the new keys.

2. Press SHIFT while dragging the time slider.

3. Set options in the Create Key dialog.

Moving, Sliding, and Aligning Keys and Ranges

You can Move, Slide, or Align one or more keys horizontally in their track to change when the keys occur.

 Using the Move Keys Button

Click Move Keys in Edit Keys mode to drag a selection of keys to a new time location.

- Drag the selection to a new time location.
- Press SHIFT and drag the selection to add copies of the keys at the new time location.

The new time and time offset are displayed in the Track View prompt line as you drag.

Moving Keys with the Key Time Field

After selecting one or more keys you can enter a value in the Key Time field to move all of the selected keys to that time. The Key Time field is the first field to the right of the prompt line.

- If one key is selected, or if all of the selected keys share the same time, the time value appears in the Key Time field.
- If multiple keys with different times are selected, the Key Time field is blank.

All of the selected keys move to the same time location. If there are multiple keys selected in the same track, the key with the latest time value moves to the new time location and the remaining keys are deleted.

Moving Keys with Range Bars

You can drag range bars within a track to move keys in the following ways:

- In Edit Ranges mode, drag the range bar of an animated track to move all the keys within that track.
- In Edit Keys and Edit Ranges mode, drag the range bar of a superior level track to move all animation and keys of subordinate tracks.
- In Edit Ranges mode, set Modify Subtree on and drag the range bar of a parent object to move all the keys for that parent and all of its descendents.

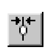
Any time you drag a range bar you are moving keys associated with the range bar. Range bars are independent of the current Edit Keys mode.

 Sliding Keys

When you slide keys the result is similar to moving a group of keys that includes your selected keys plus all the keys to one end of the animation. The direction that you drag determines which group of keys moves.

- Dragging to the right moves the selected keys plus all keys to the last key of the animation forward in time.
- Dragging to the left moves the selected keys plus all keys to the first key of the animation backward in time.

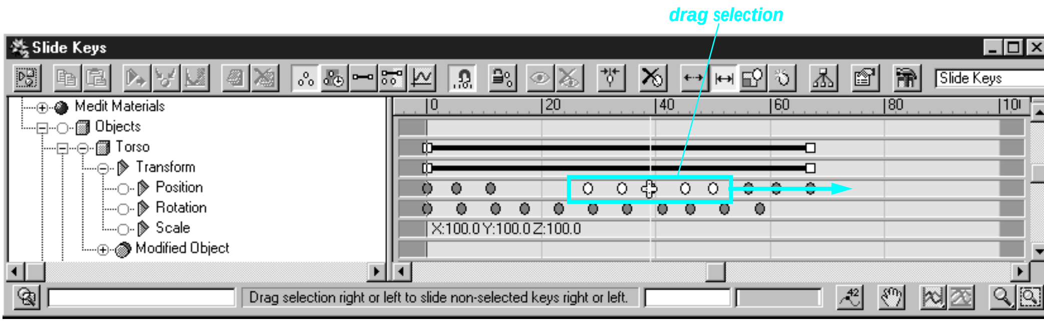
Think of Slide Keys as a way to split an animation at the selected keys and spread the ends apart.

 Aligning Keys

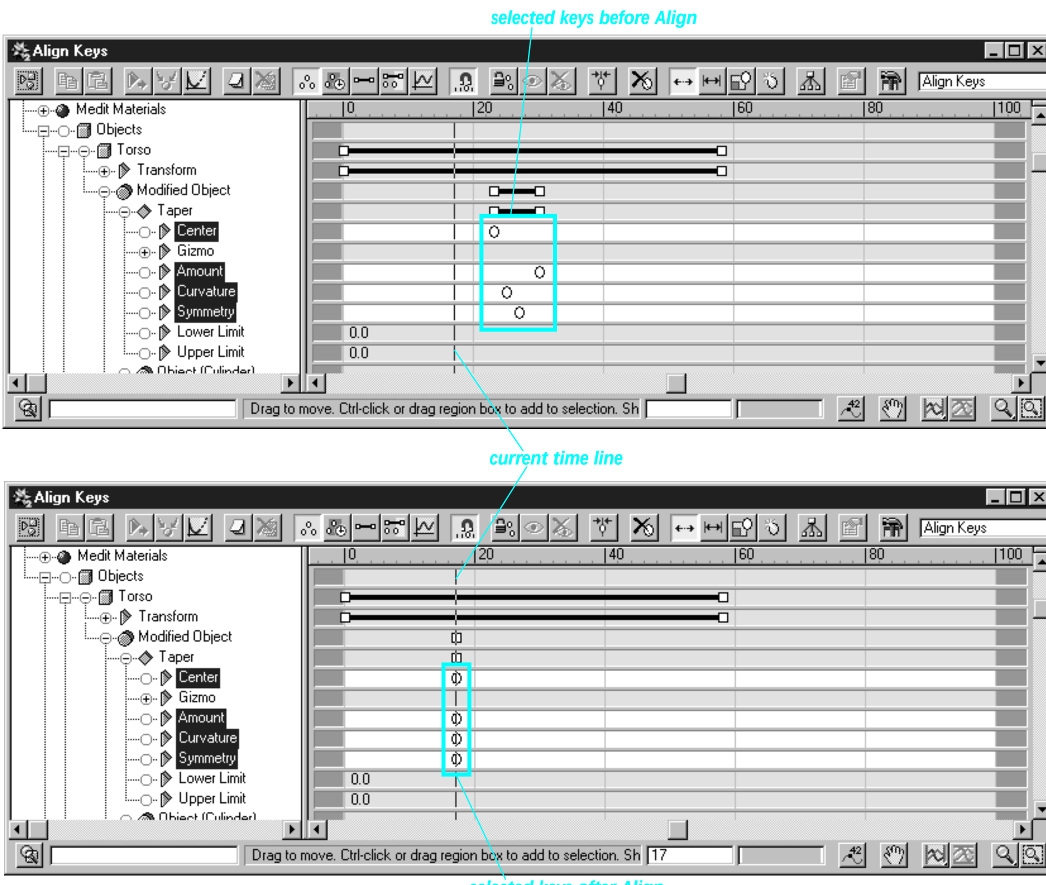
Use the Align Keys button to move selected keys to the current time. You first select the keys to align by using one of the Move, Slide, or Scale Keys buttons. Align Keys is useful for taking a group of scattered keys and moving them all to the same time location.

Align Keys ignores the state of the Snap Frames button and always uses the exact time set by the time slider.

Tip: If you want to use Align Keys to snap keys to frames, click the Time Configuration button and set Time Display to Frames.

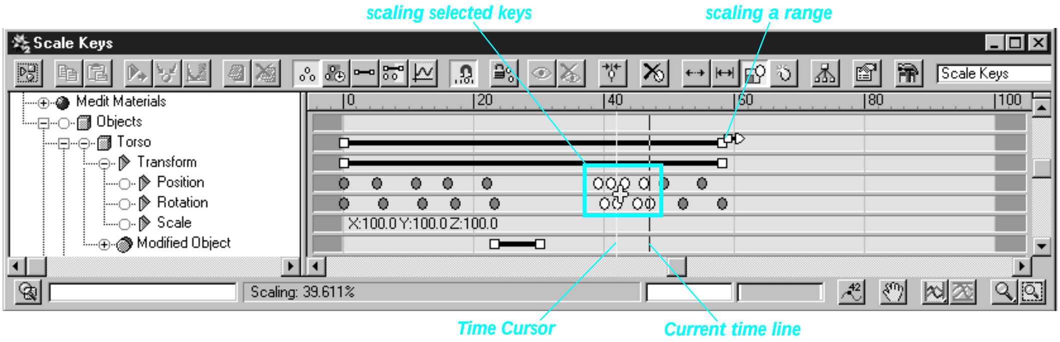


Sliding selected keys



Aligning keys to the current time

Scaling Keys and Ranges



You can scale one or more keys horizontally in their tracks to change the location and amount of time covered by the keys.

Using Scale Keys

Click Scale Keys in Edit Keys mode to change the location and amount of time covered by one or more selected keys.

- Drag to scale the selected keys.
- Press SHIFT and drag to add scaled copies of the selected keys.

The scale center is defined as the current time set by the time slider. You can scale keys about any moment in time by dragging the time slider before you use Scale Keys.

- Dragging away from the current time line expands the keys away from the current time. Expanding the selection increases time between the selected keys and slows that part of the animation.
- Dragging towards the current time line shrinks the keys towards the current time. Shrinking the selection reduces time between the selected keys and accelerates that part of the animation.
- Dragging through the current time reverses the keys and expands the keys away from the current time.

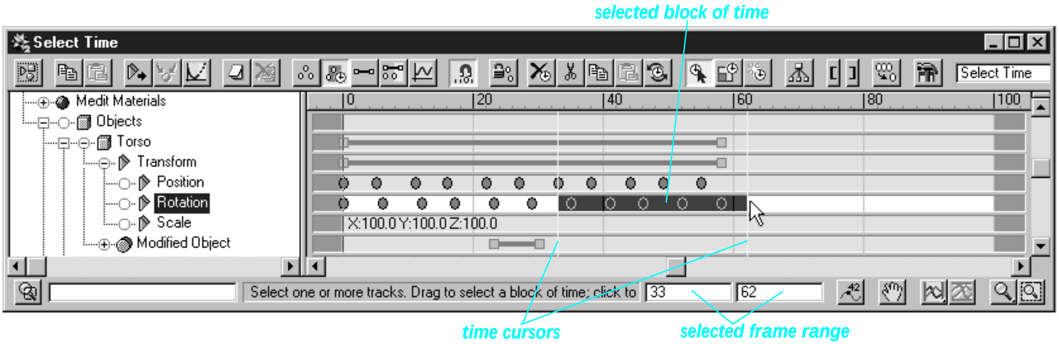
The scale percentage is displayed in the Track View prompt line as you drag the selection.

Scaling Keys with Range Bars

You can drag range bars within a track to scale keys in the following ways:

- Dragging the end of the range bar shrinks or expands all keys in the track using the opposite end of the range as the scale center.
- Dragging either end past the opposite end reverses the key order and negatively scales the selection.

Selecting and Editing Time



You use Edit Time mode to work directly with selected blocks of time as opposed to working with keys and range bars. A block of time is any contiguous time segment, across one or more tracks, and is independent of key locations.

Edit Time—Displays keys and range bars of your animation but they are there only for reference. All of your work is performed by selecting blocks of time and then applying the following time editing functions to your selection:

- Insert and Delete time.
- Cut, Copy, and Paste time.
- Scale time.
- Reduce keys in a block of time.

Selecting Time

Your first step in editing time is to select the block of time you want to work with.

To select time:

1. Select one or more item labels in the Hierarchy list to select those tracks for time editing.
2. Click Select Time or Scale Time, then click or drag in the edit window.

Clicking or dragging in the edit window while selecting time has the following effect:

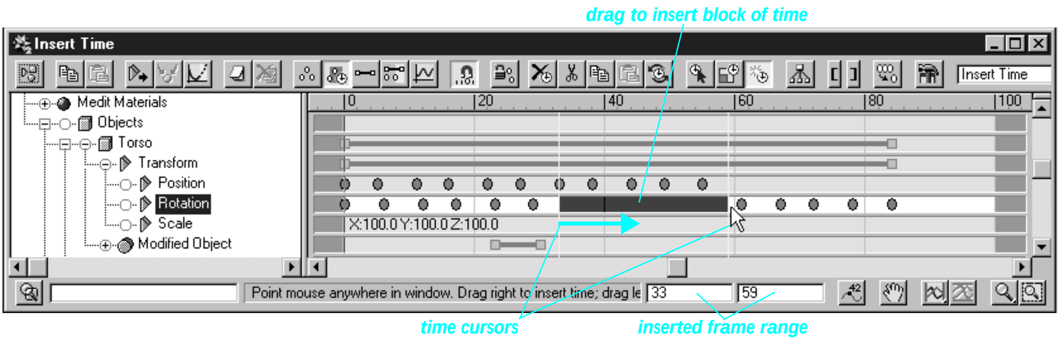
- Click anywhere in the edit window to set a time insertion point for all selected tracks. A time insertion point defines a moment in time. It has no length and it is primarily used to define where you will paste a block of time.
- Drag anywhere in the edit window to define a block of time for all selected tracks. Blocks of time are the workhorse of Edit Time selection. All Edit Time commands except Insert Time operate on a selected block of time.
- Double-click anywhere in the edit window to select all time in the Active Time Segment for the selected tracks.

If a selected track does not support time operations, time selection in that track is ignored.

Selecting Time with Modify Subtree

When you enable Modify Subtree in Edit Time mode, the keys for the subordinate tracks display in their parent tracks. Editing time in the parent track also affects the subordinate tracks.

Inserting and Deleting Time



Once you have created an animation, you might decide that you want to add or remove time from it. You make this change by inserting or deleting time at a selected point in time.

- Inserting time between two keys moves the keys apart and slows down the animation between those keys.
- Deleting time between two keys moves the keys closer together and speeds up the animation between those keys.

Inserting and Removing Time

Insert Time can be used to both insert and remove time from selected tracks.

To insert or remove time:

1. Select item labels in the Hierarchy list to select tracks for time editing.

2. Click Insert Time.

3. Do one of the following:

- *Drag to the right* anywhere in the edit window to insert time into the selected tracks.
- *Drag to the left* anywhere in the edit window to remove time from the selected tracks.

Dragging has the following effect:

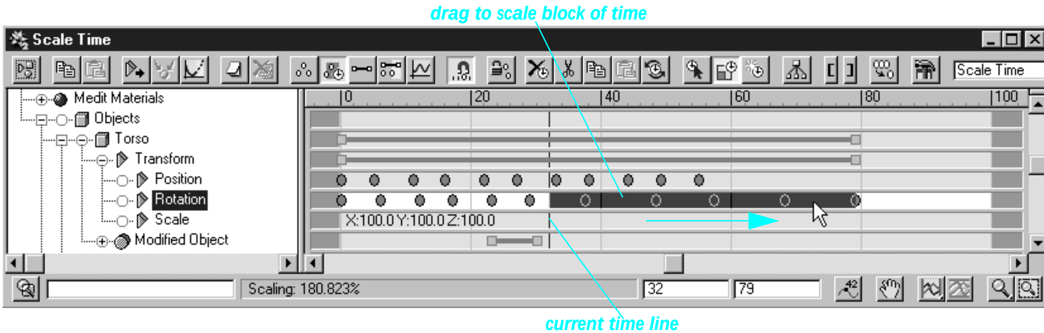
- The location of the cursor where you press the mouse button defines the start point where time will be inserted or removed.
- As you drag, inserted or removed time appears as a dark bar in the selected tracks.
- All keys to the right of the insertion point slide in the direction of the drag.
- As keys slide to the left, when removing time, they can overlap keys to the left of the insertion point. Keys to the left of the insertion point that are overlapped by the removed time are unaffected by the operation.

Deleting Time

Click Delete Time to delete a selected block of time along with any keys within the selection.

- Keys to the right of the deleted block slide to the left to fill the deleted time.
- If the deleted block of time contains all of the keys in a track, the parameter in that track is assigned the value it had at the current time as set by the time slider.

Scaling and Reversing Time



You can scale time, squeezing it to fit into less time or expanding it to fill more time.

Scaling a Selection of Time

You can scale a selected block of time by clicking Scale Time on the Track View toolbar and then dragging inside the selection.

- Drag to the right within the selection to expand time from the left edge of the selection. All keys to the right of the selection slide right as the selection expands.
- Drag to the left within the selection to shrink time toward the left edge of the selection. All keys to the right of the selection slide left as the selection shrinks.
- Drag past the left edge of the selection to reverse time and expand it with a negative scale factor. Keys inside the selection, and keys to the right of the selection, can overlap keys to the left of the selection.

Scaling Time for the Entire Scene

If you want to scale time for all tracks in the scene use, one of the following methods:

- In Edit Ranges mode, turn on Modify Subtree and drag either end of the World range bar.
- On the Time Configuration dialog, click Rescale Time. See chapter 22, “Animation Concepts and Methods.”

Reversing Time

You can reverse time by scaling a selection past its left edge. Be aware that this also changes the position of the selection and the remaining keys around it. Click Reverse Time on the Track View toolbar to reverse a selected block of time without changing position.

Using the Time Clipboard

An important feature of Edit Time mode is the ability to cut, copy, and paste time to and from the Time Clipboard. You can copy time to different places in the same track or from one track to another.

Placing Time on the Clipboard

Before you can paste time into a track you must have time in the Clipboard to paste. After selecting a block of time, you place it in the Clipboard by clicking Cut Time or Copy Time on the Track View toolbar.

Excluding Endpoints

If you paste the same block of time repeatedly, one after the other, you can create a looping segment in your animation. To create a smooth looping animation you need to exclude either the first or last key of the copied block to prevent keys from doubling up at the ends.

Click Exclude Left Endpoint or Exclude Right Endpoint to exclude the start or end key when copying a block of time. The exclude endpoints buttons work under the following conditions:

- The key to be excluded must be at the exact start or end time of the selected block of time.
- If you cut time with both the Exclude Left and Exclude Right buttons on, one of the keys is deleted because they both fall on the same time.
- If Exclude Left and Exclude Right are on and neither key is selected in Edit Keys mode, the right key is deleted.
- If Exclude Left and Exclude Right are on and one key is selected in Edit Keys mode, the non-selected key is deleted.
- If Exclude Left and Exclude Right are on and both keys are selected in Edit Keys mode the left key is deleted.

Pasting Time from the Clipboard

You can paste time into one or more tracks with these conditions:

- If you copied one track, you can paste only to selected tracks using the same controller as the copied track. Other selected tracks are ignored.
- If you copied multiple tracks, controllers in the copied tracks are matched to controllers in the selected tracks. Selected tracks with controllers that do not match the paste source controllers are ignored.
- If none of the selected tracks have controllers matching the copied tracks, Paste Time is disabled.

After clicking Paste Time, you can choose to paste in an absolute or relative manner:

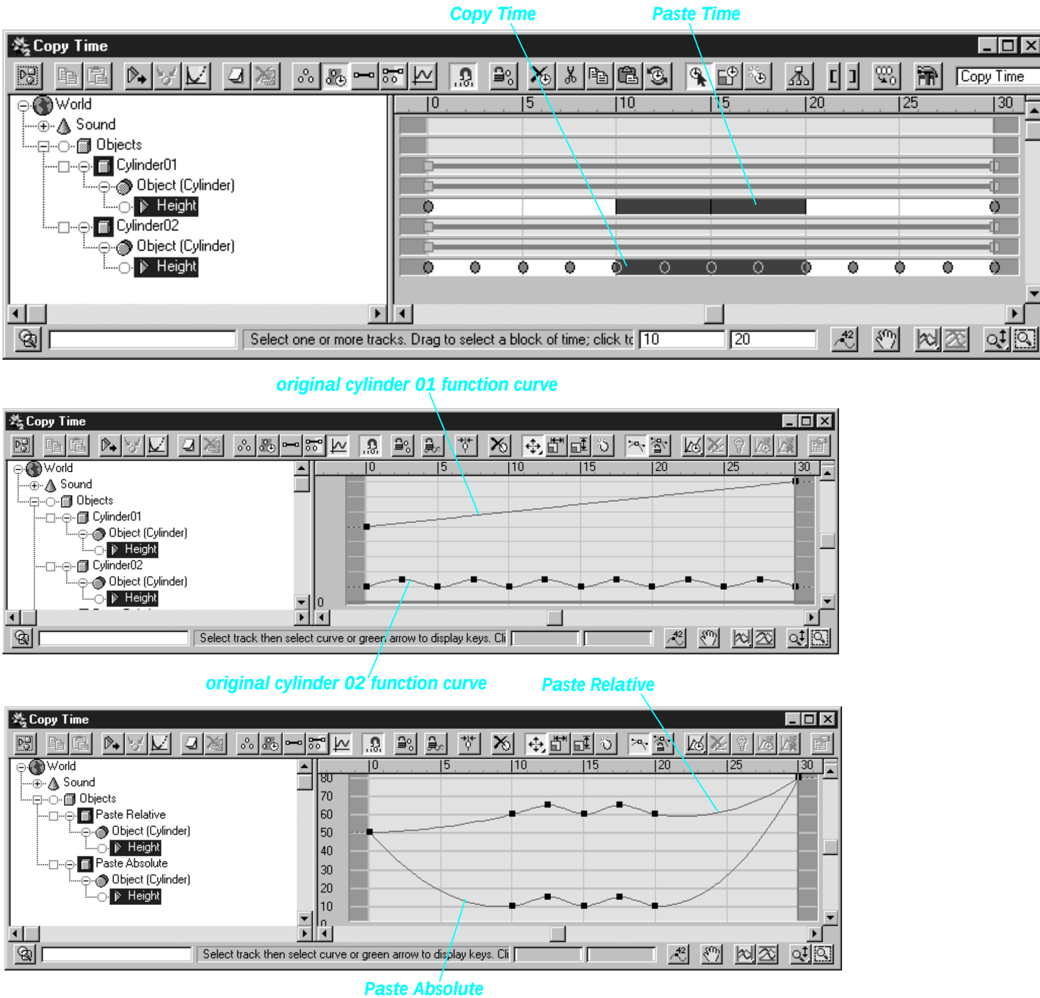
Paste Absolute—Replaces the target values with the values in the clipboard. Choose this when you want to replace one effect with another.

Paste Relative—Offsets the first value of the paste time to match the first value of the target time. The remaining paste time values are offset by the same amount and replace the target values. Keys to the right of the target are offset by the difference between the first and last value of the paste time. Choose this when you want to smoothly splice effects.

Keys to the right of the target slide to make room for the paste source.

Imagine that you have modeled two cylinders with animated heights:

- Cylinder_01 smoothly animates from a height of 50 units to a height of 80 units from frame 0 to frame 30.
- Cylinder_02 oscillates in height from 10 units to 15 and back to 10 every 5 frames.

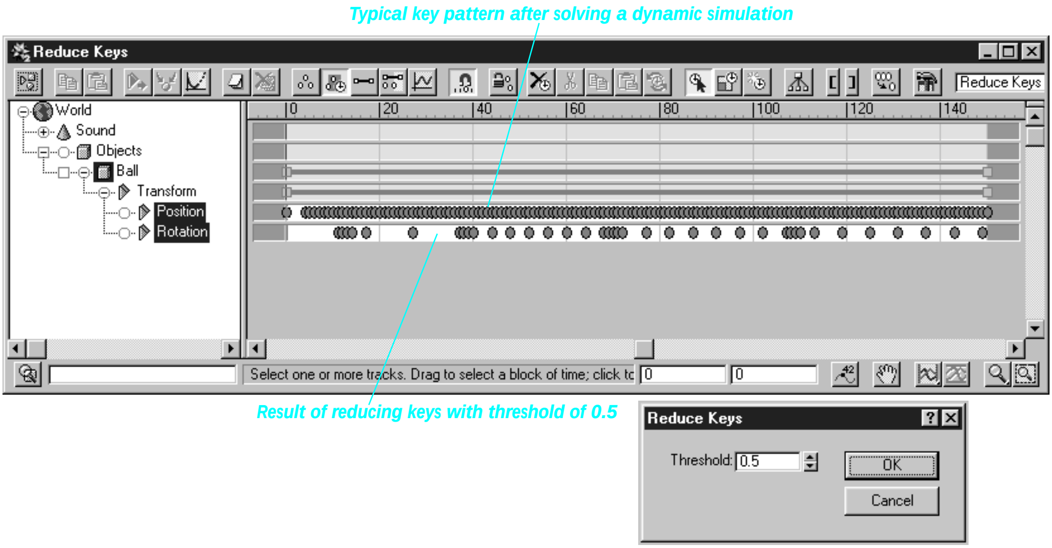


Examples of copying and pasting blocks of time

If you Copy Time for frames 10 to 20 from Cylinder_02 and Paste Time for frames 10 to 20 into Cylinder_01 you can see the differences between Paste Absolute and Paste Relative.

- Paste Absolute causes Cylinder_01 to drop from 50 to 10 units high by frame 20, oscillate for 10 frames, then shoot up in height after frame 20.
- Paste Relative causes Cylinder_01 to pause its growth at frame 10, oscillate between 60 and 65 units in height for 10 frames, and then continue its growth after frame 20.

Reducing Keys



Using inverse kinematics (IK), Dynamics, or creating any complex animation, can leave you with many keys. In the case of Applied IK, 3D Studio MAX generates a key on nearly every frame. Often, the same animation can be produced with fewer keys. Having fewer keys in a track makes it easier to change your animation.

- Click Reduce Keys in Edit Time mode to have 3DS MAX analyze keys in a block of time and create a set of fewer keys that produces nearly the same animation. You specify how closely the new animation matches the original.
- While 3DS MAX calculates the keys, a Key Reduction Progress Bar appears at the bottom of the 3DS MAX window. You can click Stop to the right of the progress bar to stop the key reduction at its current point.
- After stopping a Key Reduction, click Undo to restore the original keys.

Setting the Threshold Value

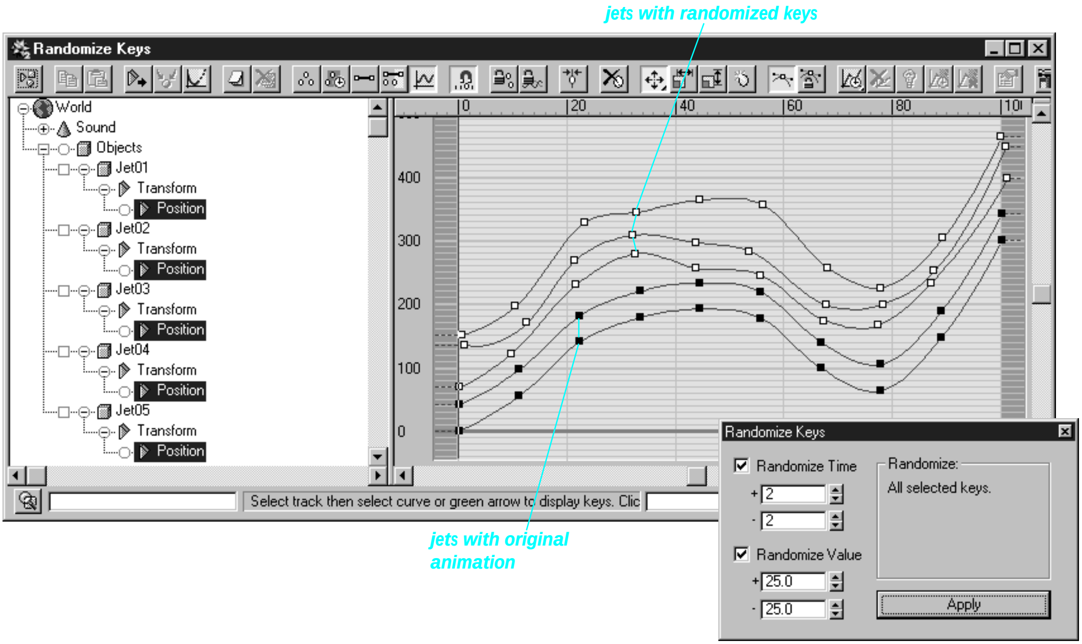
Enter a Threshold value on the Reduce Keys dialog to constrain how much the animation changes. The difference between the new animation and the original animation, at any frame, will be less than the threshold value.

- Low threshold values closely match the original animation but may not greatly reduce the number of keys.
- High threshold values produce the fewest keys but may not match the original animation with much accuracy.

For example, if you enter a threshold value of 0.25 and apply key reduction to a position track, the position of the animated object on every frame after key reduction will be within 0.25 units of its original position.

If you apply key reduction to multiple tracks, the threshold value is applied to each track using the track's value type.

Randomizing Keys



You can apply random offsets to the time or value of your animation keys with a Track View utility. A good use for randomizing keys is to introduce slight variations after copying animation to multiple objects.

Imagine that you want to animate aircraft flying in formation.

- You can position the aircraft in formation and then animate only the flight of the leader. Then you copy the animation from the leader and use Paste Time Relative to apply the same motion to the other planes.
- The other aircraft move in exactly the same way as the leader. So exact the motion looks a little fake.
- Apply the Randomize Keys utility to the aircraft to introduce subtle variations in the motion of each plane to enhance realism.

Setting Randomize Keys Parameters

After selecting keys or time in one of the Track View edit modes, click Track View Utilities on the Track View toolbar and choose Randomize Keys to apply random offsets to the selection.

On the Randomize Keys dialog you can set the following parameters:

Randomize Time—Enable this option to set positive and negative offsets in key time.

Randomize Value—Enable this option to set positive and negative offsets in key value.

Changing Animation Values

So far, everything presented in this chapter has been primarily concerned with changing the location in time of keys and animation ranges. Topics have covered many ways of moving, sliding, and deleting animation. What do you do if you want to change an animation value?

Clicking the Properties button on the Track View toolbar displays a dialog to change animation values. The type of dialog displayed depends on the type on animation track selected:

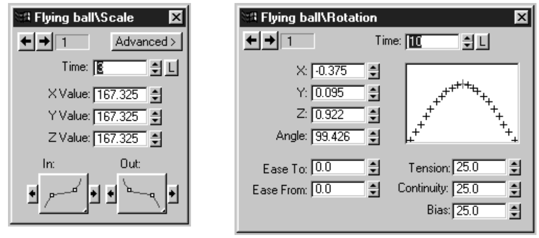
- Tracks with keys display a Key Info dialog. You can change the values in the Key Info fields to change the animation value, time, and interpolation methods of one or more selected keys.

  You can also change key values graphically by dragging their function curve. See chapter 26, “Function Curve and Trajectory Editing.”

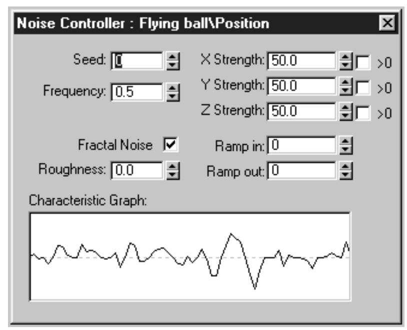
- Tracks with parametric controllers, such as Noise, display a properties dialog. You change the values in the properties dialog to change the behavior of the controller over its entire range.

- Some tracks do not display any properties dialog. In such cases you use other methods, such as editing function curves, to change the animation values.

The options and properties dialogs for the standard 3D Studio MAX controllers are described in chapter 27, “Working with Controllers.”



Typical Key Info dialogs.



Properties dialog for the Noise parametric controller

Chapter

26

Function Curve and Trajectory Editing

Function curves and trajectories are a graphic way of looking at animation keys and to see how animation values are interpolated between keys. Instead of seeing just a key at a certain time or values in a dialog, you can view a curve that shows you where each key is and how the animation flows from key to key.

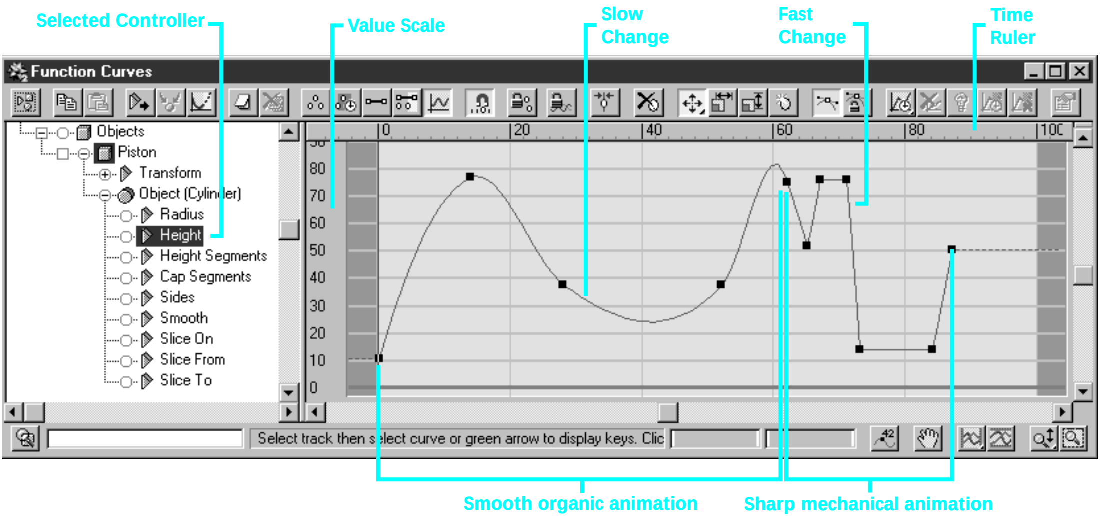
Using Function Curves mode in Track View you can chart any animated parameter as a value over time.

Trajectories displayed in viewports are a special type of function curve. Trajectories show the three-dimensional position path an object travels over time. Using both Function Curves mode and Trajectories you can easily “fine tune” the behavior of your animation.

The topics in this chapter discuss how to display and edit function curves and trajectories using tools in Track View and the Motion panel.

See the Online Reference for details about function curve and Motion panel features.

Understanding Function Curves



When you work in Function Curves mode and select an animation track, 3D Studio MAX displays a curve showing how animation values change over time. You use this display to gain insight into how your animation works and to edit your animation.

What Function Curves Represent

Function curves chart animation values over time. Studying these curves helps you understand the behavior of your animation.

The curve is charted in the following way:

- Horizontal distance represents time just as it does in the other Track View modes. Use the Time Ruler to measure horizontal distance.
- Vertical distance represents animation values. A scale appears at the left edge of the edit window to help you quickly determine function curve values.

Vertices on the curve are the same animation keys displayed in other Track View edit modes. Editing vertices in Function Curves mode is the same as changing key timing and value in the other edit modes.

- The horizontal location of a function curve vertex indicates time location and is the same as the location of the corresponding key in the other edit modes.
- The vertical location of a function curve vertex indicates animation value and is the same as the value field of the corresponding key in other edit modes.

Note: If you apply Ease or Multiplier curves to a function curve, its horizontal and vertical locations no longer match the key time and value settings. See “Using Ease and Multiplier Curves.”

Not all function curves display vertices. Only animation tracks that use keys can display vertices on their function curves. Controller types that produce animation tracks and function curves without keys include:

- Parametric controllers such as Noise.
- List controllers.
- Expression controllers.

Reading the Shape of a Function Curve

You can get a sense of what your animation is doing by looking at its function curves:

- Spiky curves with lots of peaks and valleys indicate a jittery, erratic animation.
- Gentle, stretched-out curves indicate a smoothly flowing animation.
- Straight lines and sharp corners on a curve indicate mechanical, linear animation.
- Smooth curves and rounded corners indicate organic, natural animation.

You get a feel for the speed of an animation by looking at the slope of its function curves:

- Steep curves indicate a fast rate of change. As a curve approaches vertical, the animation gets faster.
- Flat curves indicate a slow rate of change. As a curve approaches horizontal, the animation slows down.

By just looking at a function curve you can tell a lot about your animation. The following topics describe how you can display and edit these curves to control your animation.

Displaying Function Curves

Function Curves mode does not immediately show you information in all of the tracks. If it did, the mass of curves displayed would be unreadable. You choose which curves to display in the Function Curves edit window.

 Click Function Curves in the Track View toolbar to activate Function Curves mode.

Choosing Function Curves to Display

Only animation tracks can display a function curve. These are tracks with controller items (items with a green triangle icon). An exception is the Wave Form sound controller. It displays the sound wave form in Function Curves mode.

To display a function curve:

- Click a controller item label to display the item's function curve.

If choosing an item label does not display a function curve, check the following:

- The item must be a controller. You can choose any item in the Hierarchy list but only controller items display function curves.
- The controller must be able to display function curves. For example, many rotation controllers do not display a function curve.
- The controller might not be animated. Controllers without animated values display as a flat line and may be hard to see.
- The value range of a curve might not display when displayed with other curves.

  For example, a function curve for Height, animated from 5 to 10 units, will be too small to see when displayed with a function curve for Position animated from -200 to 500 units.

Reading Curves and Colors

When you display the function curve of an item, you might see one or more curves displayed. The number of curves you see is determined by the number of components making up the value animated by the item's controller.

Single-value controllers, such as Radius or Length, display one gray curve.

Multivalue controllers, such as the X, Y, and Z values of Position, display a colored curve for each of the XYZ components.

Function curve display colors are organized in the following way:

- Red indicates the first component.
- Green indicates the second component.
- Blue indicates the third component.
- When two or more curves occupy the same location, the color of the last curve drawn is displayed. Blue is displayed over Green and Green over Red.

Filtering Function Curve Display

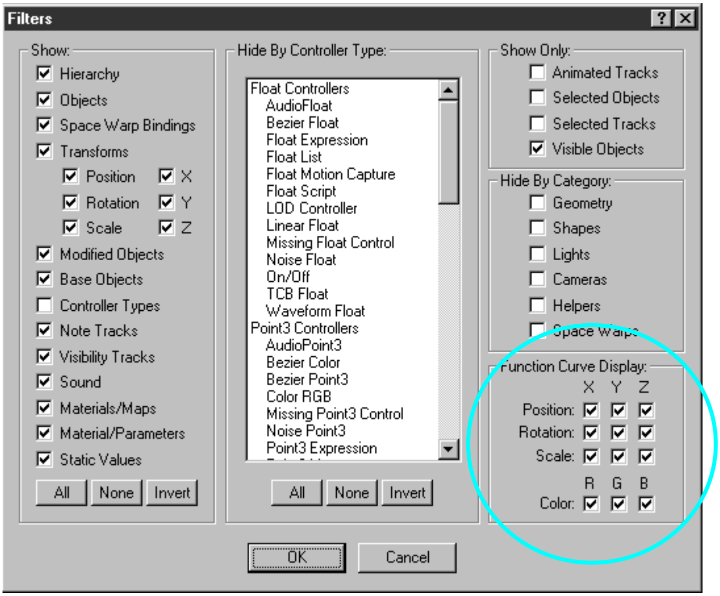
Use options in the Filters dialog to filter the display of certain three-component function curves. In the Function Curves area of the filters dialog are the following options:

Position—Filters display of the Position X, Y, and Z curves.

Rotation—Filters display of the Rotation X, Y, and Z curves. Because many 3DS MAX rotation controllers do not display function curves, these filters are meant for use by plug-in controllers.

Scale—Filters the display of the Scale X, Y, and Z percentage curves.

Color—Filters the display of Color RGB and HSV curves for the Color controller.



Function curve filter options

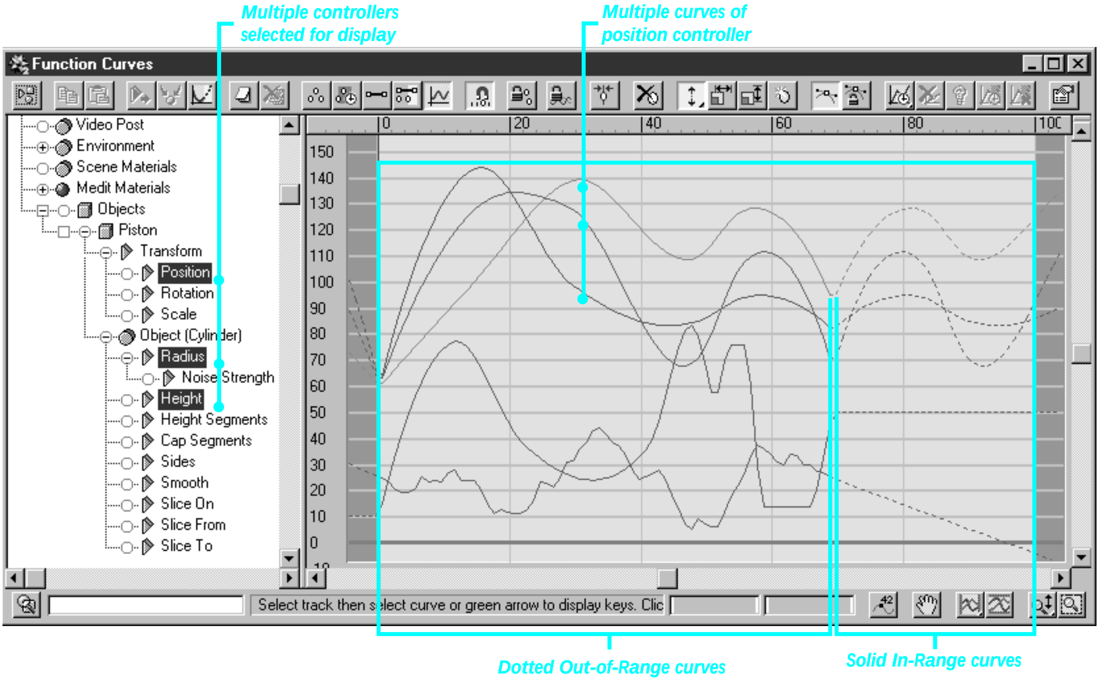
Function Curve Line Types

Function curves are displayed using two line types.

- A solid line charts the curve over the range of the animation. The range is set by each track's range bar.
- A dashed line charts the curve outside the range of the animation. That is, how the animation behaves before the beginning of the range and after the end of the range.

  You can choose many types of behavior for a curve outside the range of the animation. See "Using Function Curve Out-of-Range Types."

Selecting Function Curves and Vertices



Before you can edit a function curve you must select the curve and one or more of its vertices. Choosing items from the Hierarchy list only displays the curves. You use the following techniques to select curves and vertices.

Selecting Function Curves

The main purpose for selecting a function curve is to gain access to the curve's vertices. Once a curve is selected, its vertices appear as black squares along the curve. If any keys were selected in the track corresponding to a selected function curve, the vertices matching the selected keys will also appear as selected.

You can select function curves in either the Hierarchy list or the edit window.

To select function curves:

- Click the green controller icon to the left of a selected item label in the Hierarchy List.
- Click any curve in the Edit Window.

Depending on the number of values managed by the controller, vertices might appear on one or more curves when a single curve or item is selected.

Freezing Unselected Curves

You can freeze function curves to prevent you from accidentally selecting their vertices when you have multiple function curves displayed. This is useful when you want to use one curve as a reference while you select and edit the vertices of another curve.

 Turn on Freeze Nonselected Curves to prevent curve selection in the edit window. Freeze Nonselected Curves has the following effect:

- Selected curves display as solid lines with black squares at their vertices.
- Nonselected curves display as a dotted lines.
- You can only select vertices from selected curves in the edit window.
- You select other curves by clicking their icon in the Hierarchy list.

Selecting Vertices

The method of selecting vertices in Function Curves mode is similar to selecting keys in other modes. Click a transform button on the Track View toolbar, such as Move or Scale, and then select keys by clicking or dragging a region.

Locking Your Vertex Selection

Clicking a key or clicking in empty space releases the current selection of keys.

 Turn on Lock Selection to prevent any changes to the current vertex selection set. Lock Selection has the following effect:

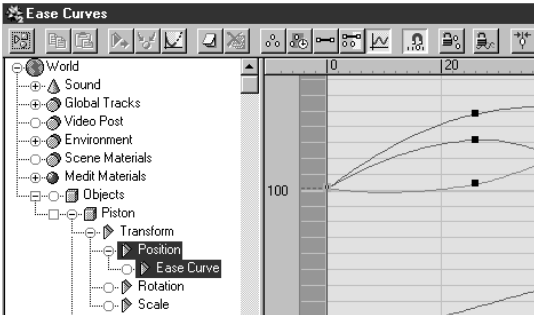
- Clicking in the edit window has no effect on vertex selection.
- Dragging anywhere in the edit window applies the current active transform to the vertex selection.

Displaying Vertex Values

The value assigned to each vertex on a function curve is displayed in the Key Value field when the vertex is selected. You can also turn on Show Selected Key Stats to show time and value for every selected vertex.

 Turn on Show Selected Key Stats to display Time and Value next to all selected keys.

Using Function Curve Out-of-Range Types



An important labor-saving concept in traditional animation is the creation of loops and cycles. The idea behind loops and cycles is to create a short pattern of keys that produce an animated effect and then repeat those keys continuously throughout the animation.

In 3D Studio MAX, you create loops, cycles, and other effects by adjusting range bars and applying Out-of-Range types.

 Click Parameter Out-of-Range Types to select how your animation behaves outside the time covered by the range bar.

You can apply Out-of-Range types to an animation track in any of the Track View modes. You get the best view of an Out-of-Range type's effect in Function Curves mode.

Setting Range with Position Ranges

Range bars indicate the total time covered by an animation track. Usually the range is measured from the first key of an animation track to the last key. 3D Studio MAX calculates your animation in two parts:

In-Range Animation—Uses animation values you define between the ends of a range bar.

Out-of-Range Animation—Uses the Out-of-Range types to determine what happens beyond the ends of a range bar.

 You click Position Ranges mode to adjust the range bar independent of its keys. The edit window in Position Ranges mode shows information only in animation tracks. Each track displays its keys and its range bar. You can drag range bars to define ranges that are longer or shorter than, or offset from the pattern of keys.

Decoupling a Range Bar

The process of adjusting a range bar independent of its keys is called *decoupling* the range. You decouple a range bar for two reasons:

- To ignore some keys at the start or end of an animation range. Keys outside of the range bar are ignored during animation playback. The keys outside the range still affect interpolated values inside the range, but the applied Out-of-Range type is used to animate time outside the range bar.

- To add extra time before the first key or after the last key that is not affected by the applied Out-of-Range type. Time beyond the first or last key of the track, but still within the range uses, the constant value of the nearest key within the range.

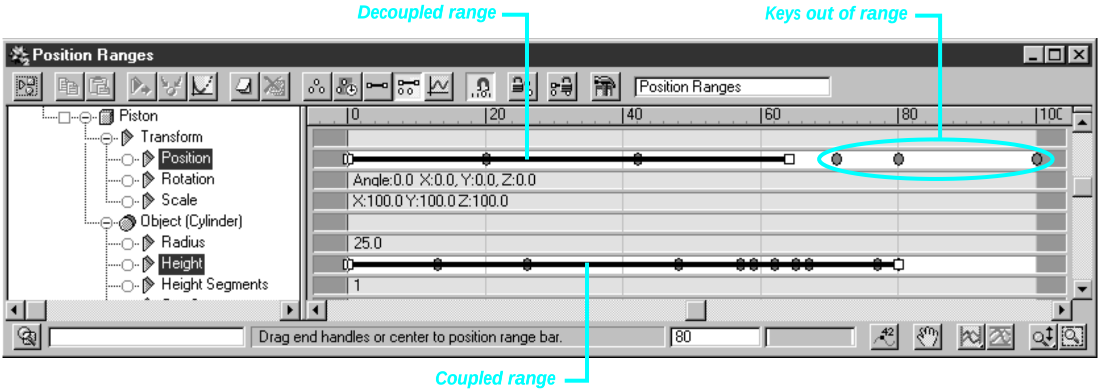
To decouple a range bar from its keys in Position Ranges mode, do one of the following:

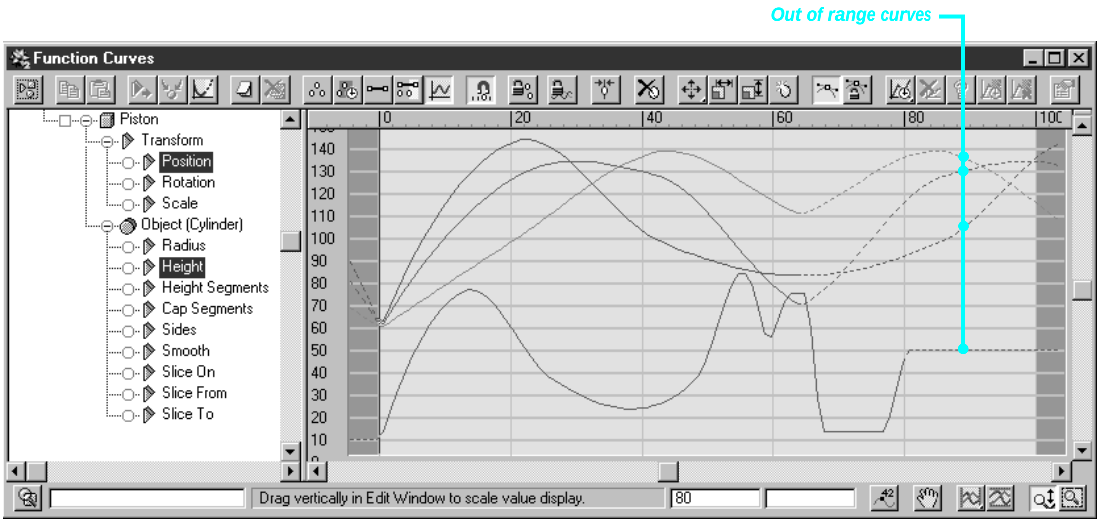
- Drag the entire range bar to shift it to the left or right of the keys.

- Drag either end of the range bar to make it longer or shorter than the keys.

Recoupling a Range

Positioning a range bar so it matches the first to last keys of a track is called *recoupling* the range. You can easily recouple a range bar with its keys using the Recouple Ranges button. This restores the range to its default condition.



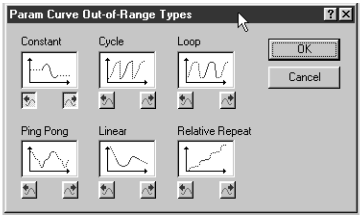


To recouple a range in Position Ranges mode:

1. Select one or more item labels in the Hierarchy list to select tracks to be recoupled.

2.  Click Recouple Ranges.

Applying Out-of-Range Types



Choices in the Param Cures Out of Range Types dialog

You choose Out-of-Range types to control the behavior of your animation outside the range set by a track's range bar.

 Click Parameter Out-Of-Range Types to display the Param Curves Out-of-Range Types dialog, where you choose out of range behavior. Arrow buttons beneath each Out-of-Range type apply the effect before the animated range, after the animated range, or both.

You choose and apply out of range types by doing the following:

- Click an Out-of-Range Type image to apply that type both before and after the function curves of the selected items.
- Click the Left Arrow below an Out-of-Range Type image to apply it only before the function curve of the selected items.
- Click the Right Arrow below an Out-of-Range Type to apply it only after the function curve of the selected items.

The Out-of-Range types and their uses are:

Constant—Holds the value at the range ends. Use constant when you want no animated effect at the range ends.

Linear—Projects the animation value along the tangent to the function curve at the range ends. Use linear when you want animation to enter and leave the range at a constant velocity.

Cycle—Repeats the animation within the range. If the beginning and end of the range have different values, the animation jumps from the end of one cycle to the beginning of the next. Use cycle when you want an animation to repeat but do not need the ends to match.

Loop—Repeats the animation within the range and interpolates between the values of the last key and first key to create a smooth loop. If the first and last key are at the ends of the range, Loop behaves like Cycle. If you use Position Ranges to extend the range bar beyond the keys, the added length adds time used to interpolate between the last and first key. Use loop with an extended range bar when you want smoothly repeating animation.

Ping Pong—Alternates between a forward and backward repeat of the animation within the range. Use ping pong when you want your animation to alternate back and forth.

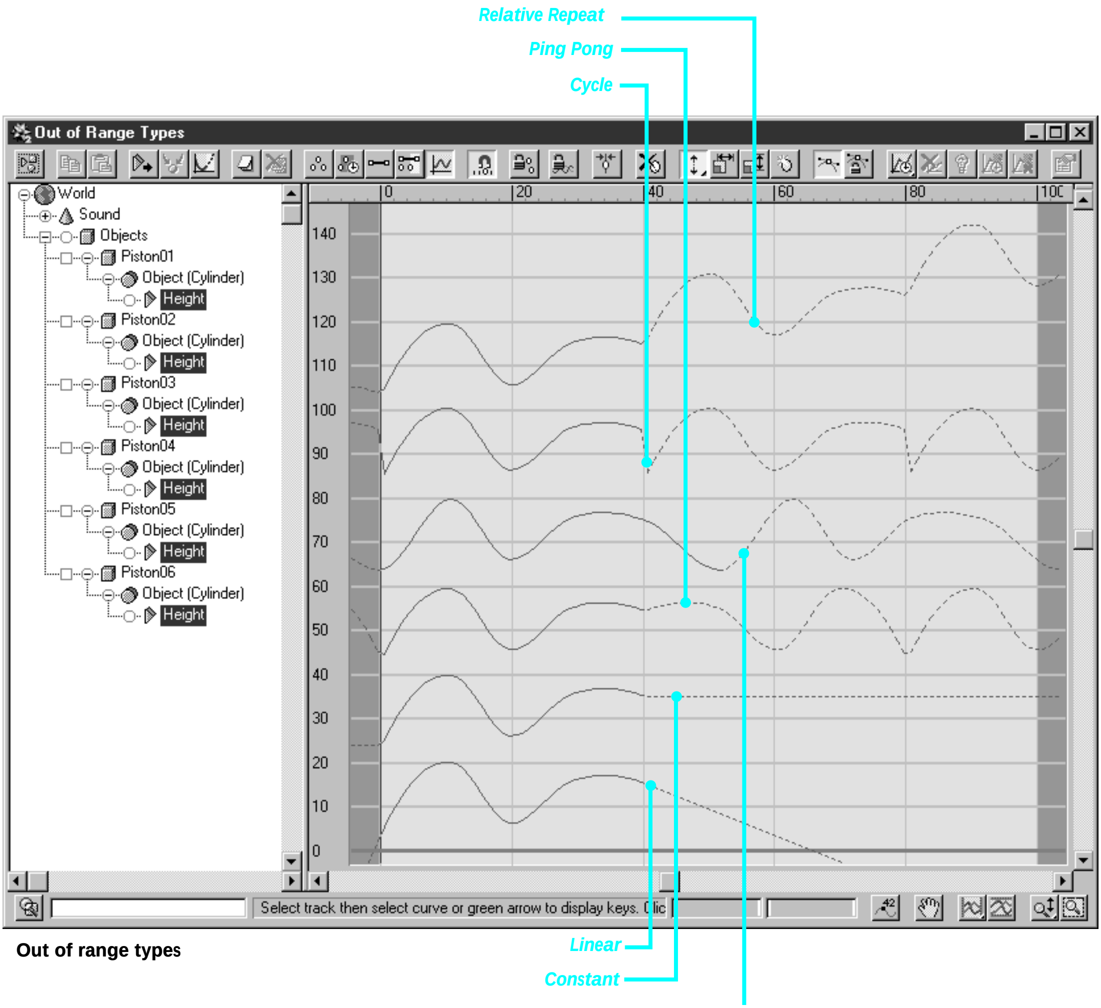
Relative Repeat—Repeats the animation within the range and offsets each repetition by the value at the end of the range. Use relative repeat when you want animation that builds on each repetition.

Converting Out of Range Types to Keys

You can use the Create Out Of Range Keys Track View utility to convert out of range types to animation keys. Some useful applications of this utility include:

- Limiting the time of a repeating out of range type and switching to a constant or linear.
- Converting an out of range type to keys that you can edit.

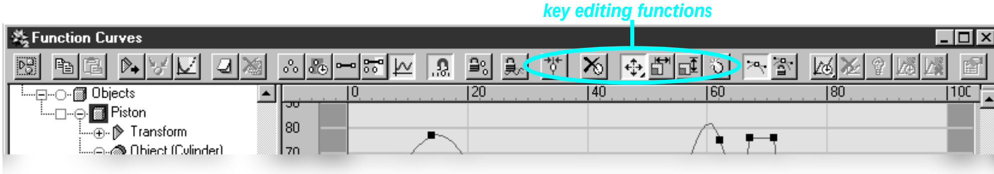
You set the amount of time when keys are created and how many keys to create on the Out Of Range Keys dialog.



Out of range types

Note: Keys are not created for the Constant out of range type.

Editing Function Curve Keys



Key editing in Function Curve mode works much the same as key editing in Edit Keys mode with the following exceptions:

- You can change key value by moving keys.
- You can change In and Out tangent by dragging tangent handles. See the next topic for information about changing key tangents.

Adding Keys

You add keys to a function curve by clicking Add Keys on the Track View toolbar, then clicking anywhere on a function curve to add a key at that location.

If the curve is part of a multiple curve controller, keys are added to each curve. For example, adding a key to the red X curve of a Position track also adds keys to the green Y and blue Z curves.

Key values are set by the following:

- Keys added between keys receive the interpolated value of the function curve.
- Keys added before the first or after the last key receive the same value as the nearest key.

Deleting Keys

You delete keys from a function curve by selecting keys and then clicking Delete Keys.

Deleting a key on one curve of a multicurve controller will delete keys at the same time location on the remaining curves.

Moving Keys

You move selected keys to change timing and value of the keys using the following methods:

- Click one of the three Move Keys buttons to move selected keys. Press SHIFT and drag the selection to copy the keys.
- Enter a value in the Key Time field to move selected keys horizontally.
- Enter a value in the Key Value field to move keys vertically.

Aligning Keys

You move all selected keys to the current time using the following methods:

- Click Align Keys on the Track View toolbar.
- Enter a value in the Key Time field to move selected keys horizontally.

If more than one key is selected on the same function curve, the left-most selected key moves to the current time and the remaining selected keys maintain the same relative distance from the left-most key.

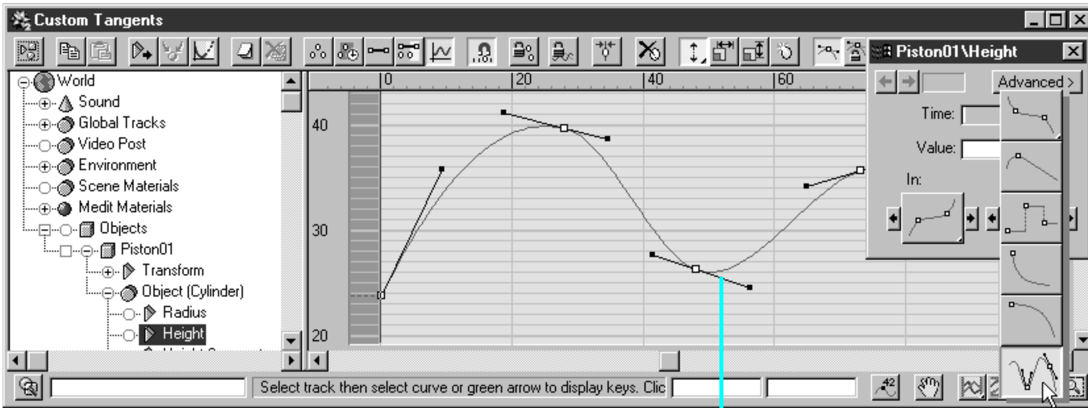
Scaling Keys

You scale the time or value of selected keys by clicking Scale Keys or Scale Value on the Track View toolbar.

Scale Keys—Drag the selection to change the time of selected keys. The scale center for Scale Time is the current time set by the time slider.

Scale Value—Drag the selection to change the value of selected keys. The scale center of Scale Values is always the horizontal base line at the value of 0.

Assigning Custom Key Tangents



Custom tangent handles

The tangent of a function curve affects the interpolated values between keys of an animation. Most animation controllers use fixed tangents to define the function curve at a key location. Your main method of changing the interpolated values of your animation is by assigning different controllers. See chapter 27, "Working with Controllers."

In 3D Studio MAX the Bezier controller type can display *custom tangent handles* at key locations. You can adjust these tangent handles much as you adjust the vertices of Spline objects.

Assigning Custom Tangent Handles

You can assign the Custom Tangent type only to Bezier controllers. The easiest way to tell whether a displayed function curve uses a Bezier controller is to turn on Controller Types in the Show area of the Track View Filters dialog.

Most animated items in 3D Studio MAX use the Bezier controller type. You can change many non-Bezier controllers to Bezier controllers using the Assign Controller button. See chapter 27, "Working with Controllers."

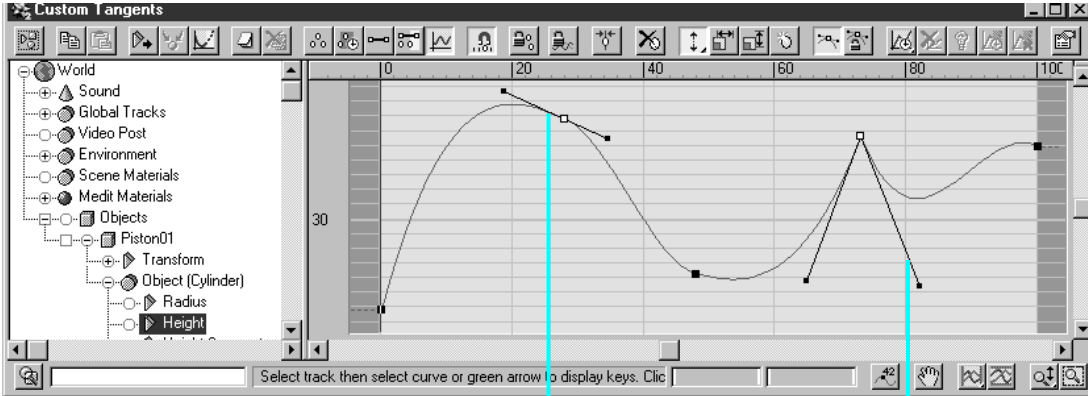
To assign a Custom tangent:

1. Select a function curve of a Bezier controller.
2. Select keys on the curve.
3.  Click Properties on the Track View toolbar to display the Key Info dialog.
4. Choose the Custom tangent type from either the In or Out flyouts on the Key Info dialog.

Displaying Custom Tangent Handles

 When Show Tangents is enabled, you can see key tangent handles on selected keys that use the Custom tangent type.

Adjusting Custom Tangent Handles



Continuous handles   Discontinuous handles

You adjust Custom tangent handles by dragging ends of the handles using Move Keys. The tangent handles can be either continuous or discontinuous.

- Continuous handles behave like a lever with its pivot point at the key location. Moving either end of the handle also moves the opposite end. The function curve passes smoothly through a key with continuous handles.
- Discontinuous handles behave like leaves on a hinge with the hinge pin at the key location. Either end moves independent of the other. The function curve abruptly changes at a key with discontinuous handles.

Dragging Tangent Handles

Click any of the Move Keys functions to drag either end of a tangent handle. You can use Lock Tangents and the SHIFT key to modify the effect of dragging.

Drag tangent handles to do the following:

- Drag continuous handle ends to rotate both handles about the key location.
- Drag the end of one discontinuous handle to rotate it about the key location.
- Press SHIFT and drag an end of a continuous handle to break continuity and rotate only that handle about the key location.

 Lock Tangents has the following effect:

- Dragging a continuous handle affects all selected keys.
- Dragging a discontinuous handle affects both handles of all selected keys.
- Pressing SHIFT while dragging the end of a discontinuous handle rotates the matching handle about all selected key locations.

Converting Between Continuous and Discontinuous Tangents

You can convert back and forth between continuous and discontinuous tangent handles.

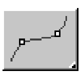
To convert continuous handles to discontinuous:

- Press SHIFT while dragging a continuous handle to break continuity. The tangents are now discontinuous.

Converting a discontinuous tangent to a continuous tangent requires changing the tangent type from Custom to Smooth and back to Custom again. This technique is performed at the key level and can affect unselected vertices if they belong to the same key as one of the selected vertices.

To convert selected discontinuous handles to continuous:

1.  Click Properties to display the Key Info dialog.

2.  Choose Smooth for both the In and Out tangent types.

3.  Choose Custom for both the In and Out tangent types.

The selected tangents are now continuous.

Using Ease and Multiplier Curves

You use Ease and Multiplier curves, in Function Curve mode, to apply time and value displacements to other function curves. Using Ease and Multiplier curves, you apply global effects across the entire range of an animation.

What Ease Curves Do

Ease curves vary the timing of an affected function curve. A normal function curve charts changes to a parameter over time. An Ease curve charts changes to the time of it's affected function over time.

The horizontal scale of an Ease curve represents normal time, just as it does for all function curves. The vertical scale of an Ease curve represents the time scale of the affected function curve. Changing the shape of an Ease curve changes the way time is interpreted by the affected function curve.

Imagine you have animated a flying bird. Later you decide to change the position of the bird so it moves quickly at the beginning of the animation and then slows towards the end.You can make the change by editing the bird's position track, but applying an Ease curve is easier.

The first figure shows function curves for the bird's position track with an identity Ease curve applied. The second figure shows the result of editing the Ease curve. Dragging the Ease curve above the identity line causes time to compress near the beginning of the animation and stretch out near the end. You can tell the bird's motion starts out fast and then slows down by looking at the effect the Ease curve has on the Position function curves.

Reading the Ease Curve

You read an Ease curve's affect by comparing value and slope to the identity curve. The identity ease curve causes no change to its affected function curve. It has a slope of 1 so that its value and time location are always equal.

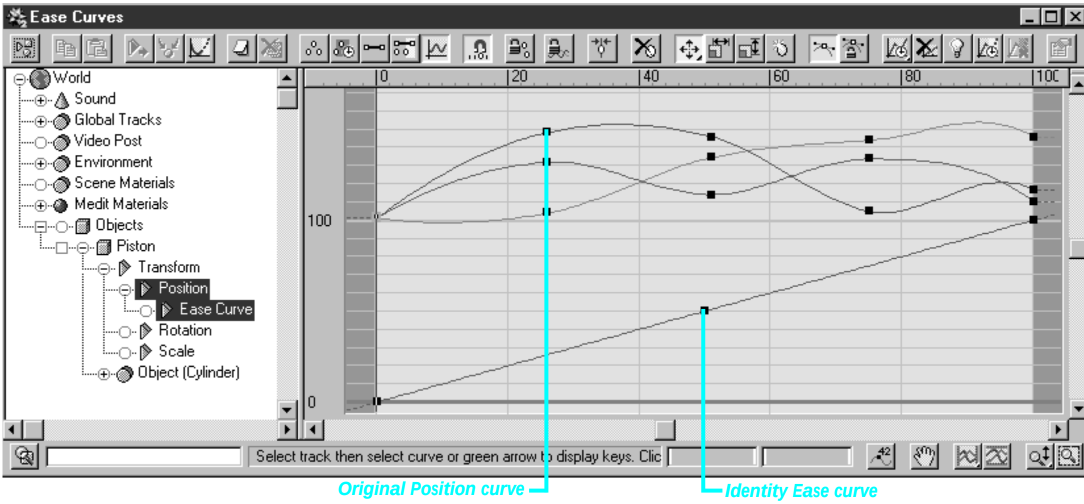
The value of an Ease curve references the value of the affected function curve at a specific time. An Ease curve with a value of 55 at frame 40 causes the value of the affected function curve at frame 55 to occur on frame 40. The Ease Curve has compressed, or advanced, time because the value that was originally set to occur at frame 55 now occurs at frame 40.

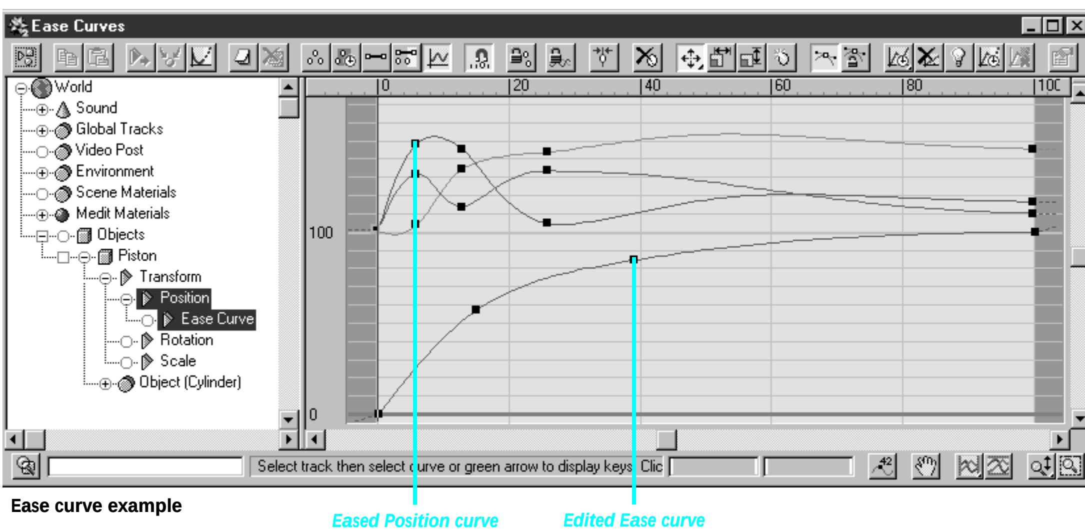
Ease curve value has the following meaning:

- Values above the identity curve advance time. Values on the original function curve occur earlier than their original time.
- Values equal to the identity curve do not affect time. Values on the original function curve occur at their original time.
- Values below the identity curve delay time. Values on the original function curve occur after their original time.

Ease curve slope has the following meaning:

- When the slope is pointing up, time moves forward.
- When the slope is horizontal, time pauses and the animation stops.
- When the slope is pointing down, time moves backwards.
- When the slope is steeper than the identity curve, time is compressed and the animation moves quickly.
- When the slope is equal to the identity curve, time is normal and the animation moves at its normal rate.
- When the slope is flatter than the identity curve, time is expanded and the animation moves slowly.





Ease curve example

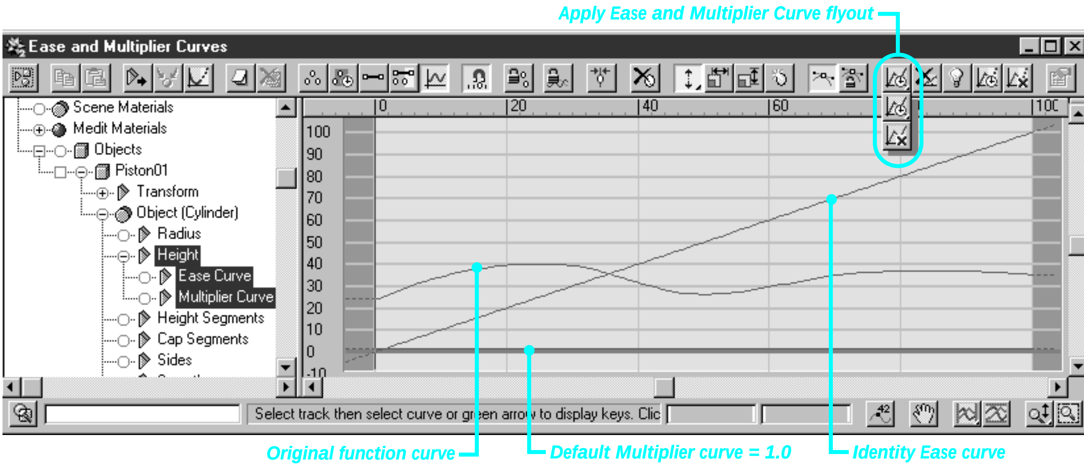
What Multiplier Curves Do

Multiplier curves are much easier to understand than Ease curves. The value of a multiplier curve is a scale factor applied to the value of its superior function curve.

The default value of a Multiplier curve is a horizontal line with a value of 1.0.

- Values greater than 1.0 increase the value of the function curve.
- Values below 1.0 decrease the value of the function curve.
- Values less than 0.0 negatively scale the value of the function curve.

Applying Ease or Multiplier Curves



You apply Ease and Multiplier curves in Function Curves mode. Once you apply an Ease or Multiplier curve, you can edit its keys, ranges, and properties like any other track.

 Click Apply Ease Curve or Apply Multiplier Curve to apply these curves to controller items selected in the Hierarchy List.

Not all controller items can receive Ease or Multiplier curves. For example, the Path position controller cannot receive an Ease or Multiplier curve, but its subordinate Percent controller can. Apply Ease Curve or Apply Multiplier Curve are unavailable if the selected controller cannot accept Ease or Multiplier curves.

You can apply ease and multiplier curves to most controllers in 3D Studio MAX. This includes applying ease or multiplier curves to other ease or multiplier curves. Experiment with this technique to produce complex effects.

Applying Ease or Multiplier Curves to Multiple Items

You can apply an instanced Ease or Multiplier curve by selecting multiple controller items and then clicking Apply Ease Curve or Apply Multiplier Curve. All of the applied Ease or Multiplier curves are instances and changing one curve changes all of the curves.

Use this technique when you want to apply the same Ease or Multiplier curve to multiple items.

For example, imagine you have animated three bouncing balls. You can adjust the timing of all three balls by applying an instanced Ease curve. You select the position track for each ball and then click Apply Ease Curve. When you adjust the Ease curve for any ball, all three balls are affected.

 Making Instanced Ease or Multiplier Curves Unique

You can convert an instanced Ease or Multiplier curve into an independent curve by clicking Make Unique.

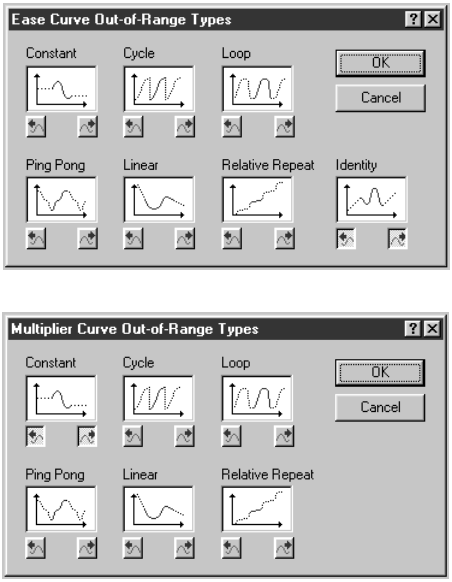
Deleting an Ease or Multiplier Curve

You can delete an Ease or Multiplier curve by clicking Delete Ease/Multiplier Curve.

Enabling and Disabling Ease and Multiplier Curves

You can toggle an Ease or Multiplier curve on and off by clicking Ease/Multiplier Curve Enable/Disable. Use this when you want to work with the original function curve rather than a curve altered by Ease and Multiplier curves

Choosing Ease or Multiplier Curve Out-of-Range Types



Ease Curve and Multiplier Curve Out-of-Range Type dialog

Ease and Multiplier curves use the same Out-of-Range types as other function curves, with two exceptions:

- Out-of-Range types for Ease and Multiplier curves are applied with their own buttons.
- Ease curves have one extra Out-of-Range type called the Identity type.

You usually want to use the same Out-of-Range type for Ease and Multiplier curves that you use for their affected function curve. See “Applying Out-of-Range Types.”

 Click Ease Curve Out-Of-Range Types to display the Ease Curve Out-Of-Range Types dialog for selected Ease curves.

 Click Multiplier Curve Out-Of-Range Types to display the Multiplier Curve Out-Of-Range Types dialog for selected Multiplier curves.

Out-of-Range Types

Identity—Projects the Ease curve along a line with a slope of 1.0. This causes time to flow at the normal rate outside the range of the Ease curve. Use identity when you want the Ease curve to be a localized, non-repeating effect.

Identity for an Ease curve produces an effect similar to constant for other types of curves.

Constant—Holds the value at the ends of the range for all frames.

Constant for an Ease curve stops time outside of the Ease curve’s range. The time value at the end of the range is held for all frames.

Linear—Projects the animation value along a line tangent to the Ease or Multiplier curve at the end of the range. Use linear when you want the animation to enter and leave the range at a constant velocity.

Cycle—Repeats the animation within the range. Use cycle when you want an animation to repeat exactly.

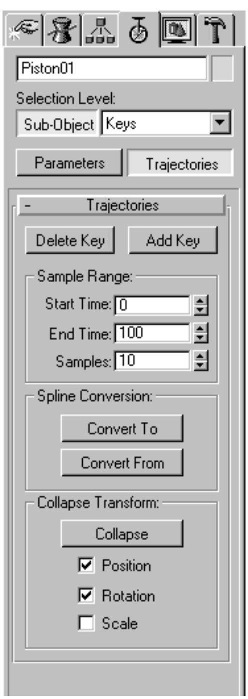
Loop—Repeats the animation within the range but interpolates between the last key and the first key of the range to create a smooth loop. Use loop with an extended range bar to produce smoothly repeating animation.

Ping Pong—Alternates between a forward and backward repeat of the animation within the range. Use ping pong when you want your animation to alternate back and forth.

Relative Repeat—Repeats the same animation as within the range but offsets each repetition by the value at the end of the range. Use relative repeat to create animations that build on each other as they repeat.

Editing Trajectories

An object and its trajectory



You click Trajectories on the Motion panel to display the path an object travels over time. Think of a trajectory as a three-dimensional function curve for an object's Position track.

When you view the Position track using the Track View Function Curves mode, you see three curves: one for each component of the *XYZ* position coordinate as the coordinates change over time. A trajectory displays the same information as a single 3D curve in your scene.

Using trajectories, you can do the following:

- Display the 3D path for selected objects' position tracks.
- Add and Delete keys from the path.
- Move, rotate, and scale keys on the path.
- Convert the path to a Spline object.
- Derive a new path from a Spline object.

Displaying Trajectories

You display trajectories for selected objects using two methods:

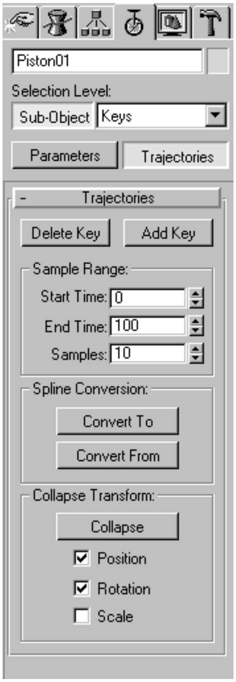
- Click Trajectories on the Motion panel to temporarily display trajectories while you edit them. The trajectories remain visible only as long as objects are selected and you remain in Trajectories area on the Motion panel.
- Select the Trajectory option on the Display Optimizations rollout on the Display panel to permanently display trajectories of selected objects. The trajectories remain visible until you turn them off.

If an object does not display a trajectory, make sure that its Position track contains animation. Trajectories only display the animated motion of the Position track.

Object trajectories appear with the following properties:

- The trajectory curve is drawn in blue.
- Frame increments display as yellow dots on the curve.
- Position keys display as white boxes surrounding the appropriate frame dot on the curve.

Editing Keys on a Trajectory



You edit keys on a trajectory with the same result as editing keys in the Position track in Track View. You edit trajectory keys by clicking Sub-Object mode on the Motion panel.

To edit keys on a trajectory:

1. Select objects.

2. Click Trajectories in the Motion panel.

3. 

Click Sub-Object to turn it on.

Adding Keys

You add keys on a trajectory by clicking Add Key on the Trajectories rollout on the Motion panel and then clicking one of the yellow frame dots on the trajectory.

Deleting Keys

You delete keys on a trajectory by selecting keys and then clicking Delete Key on the Trajectories rollout on the Motion panel.

Transforming Keys

You can click Move, Rotate, or Scale to change the location of keys displayed on a trajectory. This is the same as changing the values of a key in Track View.

Keys have no inherent size or orientation, so rotating or scaling a key is just another way of changing the key's position.

- Transforms change only the key's location. Timing or tangent type are not affected.
- Transforming keys cannot be animated.
- Rotating or scaling a key about its pivot point has no meaning. Use Selection Center or Transform Coordinate Center when rotating or scaling keys.

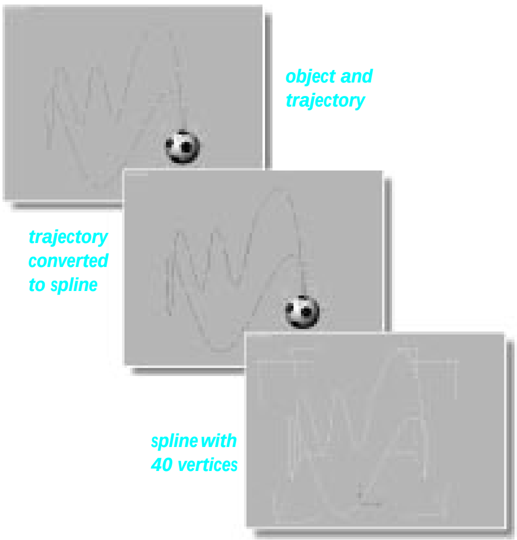
Collapsing Transforms

Click Collapse on the Trajectories rollout to create keys from the transform controllers of the selected object. The main purpose of Collapse is to convert a parametric transform controller—such as a Path or Noise controller—into transform keys you can edit.

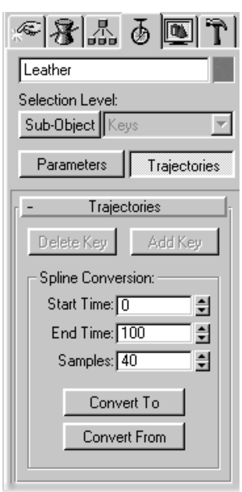
Imagine an object assigned a Path Position controller with Follow and Bank turned on. The Path controller handles both position and rotation of the object. If you collapse the Position and Rotation transforms, the Path controller is replaced with standard Position and Rotation keys that duplicate the original motion. You can then edit those keys in the Track View.

If you collapse only the Rotation transform of a Path controller, you'll get double rotation because the Path controller, though assigned to Position only, generates rotation when Follow or Bank are on. Collapse both Position and Rotation when collapsing a Path Position controller.

Using Splines as Trajectories



Converting a trajectory to a spline



You can gain great control over an object's trajectory by converting it to a spline, editing the spline, and then converting it back to a trajectory. You do this using tools in the Spline Conversion area of the Trajectories rollout.

Setting the Conversion Range and Samples

You set the Start Time and End TIme values to define the range over which the spline conversion takes place.

- When converting from a trajectory to a spline, the values define the range of frames that will be converted into a spline.
- When converting from a spline to a trajectory, the values define the range of frames over which new Position keys will be created.

Set the Samples parameters to define how many spline vertices or Position keys to create.

Converting To and From Splines

Once you have set the Start Time, End Time, and Samples parameters you are ready to perform the conversion.

- Click Convert To to create a spline from the selected object's trajectory. If multiple objects are selected, one shape is created containing multiple splines. You will usually want to use Convert To with a single selected object.
- Click Convert From, and then click a spline to create a trajectory from the spline for the selected object's. If there are multiple splines in the shape, the first spline created is used as the trajectory.

You can also make an object travel a spline path by using a Path Position controller. See chapter 27, "Working with Controllers" and the Online Reference.

Following and Banking Around Curves

You can use the Follow/Bank utility to add Rotation keys to an object with a motion trajectory to simulate turning and banking as the object travels around curves.

Following a Trajectory

An object follows its trajectory by rotating to satisfy the following conditions:

- The positive local X axis points in the direction of travel. Think of the axis as indicating the front of the object.
- The local Z axis points up. Think of the axis as indicating the top of the object.

You control which is the front of an object by rotating the object's pivot until its local X axis points in the direction you want. See chapter 23, "Building Hierarchies" for information on rotating an object's pivot.

You control which way is up by setting the following properties:

- Rotate the object's pivot until its local Z axis points to the top of the object.
- Clear Allow Upside Down on the Follow/Bank rollout to make the object's local Z axis point in the direction of the World Z axis. This option works best for most trajectories, except those which point straight up or down on the World Z axis.
- Select Allow Upside Down on the Follow/Bank rollout to make the object's local Z axis align with the plane of the trajectory. The plane of the trajectory is the plane which best fits the trajectory keys. This option works best with trajectories which lie on a plane.

If, after applying Follow/Bank, your object appears to roll or flip, try changing the setting of Allow Upside Down and apply Follow/Bank again.

Banking Around Curves

An object banks around a corner by rolling about its local X axis as it enters a curve and then rolling back to its original orientation after it leaves the curve.

Select Bank on the Follow/Bank rollout and set the Bank Amount and Smoothness parameters to control banking.

The Bank Amount parameter sets the maximum amount of banking allowed on the tightest curve of the trajectory. After applying Follow/Bank, check the motion of the object at the tightest curve. Change the Bank Amount parameter if the motion does not look right and reapply Follow/Bank.

Applying Follow/Bank

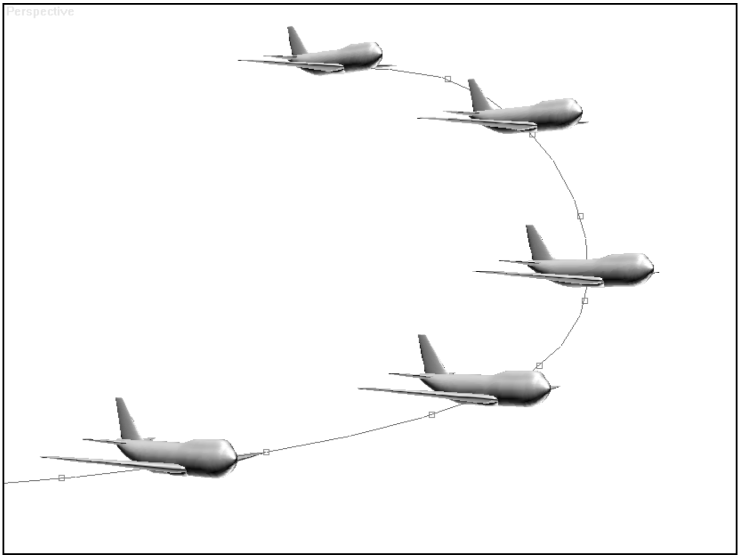
Once you have set the Follow and Bank parameters, you are ready to apply Follow/Bank to create rotation keys.

- Set the Start and End parameters to define the range of frames over which to apply Follow/Bank.
- Set Samples to define the number of Rotation keys to create.
- Click Apply Follow to create the keys.

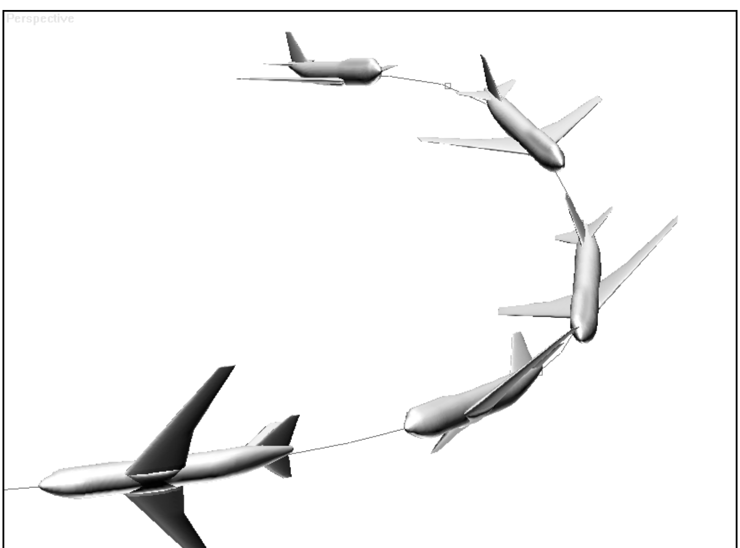
Follow/Bank is not dynamic. You have to click Apply Follow each time you change a parameter to see the effect of the change.

Choose Hold on the Edit menu, or save your file before applying.

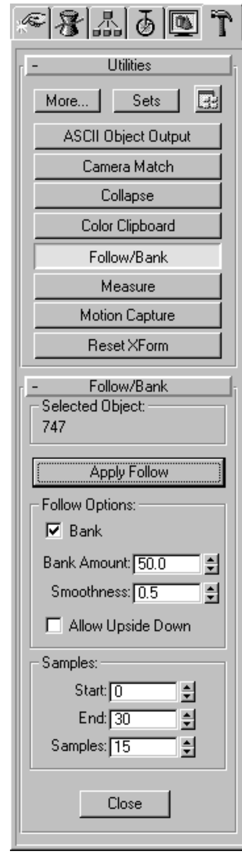
You can also make an object follow and bank around a spline path by using a Path Position controller. See chapter 27, "Working with Controllers" and the Online Reference.



Plane traveling on a trajectory



Plane after applying Follow/Bank utility



Chapter

27

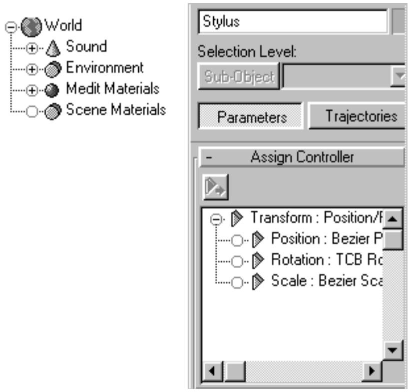
Working with Controllers

Everything you animate in 3D Studio MAX is handled by a controller. A controller is a plug-in that handles the storage and interpolation of all animated values.

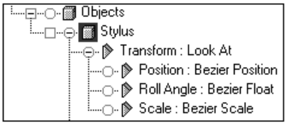
Although 3D Studio MAX has many different types of controllers, most animation is handled by the Bezier controller. Other controllers exist to provide you with animation options and special animation techniques. You can also add to the available controllers by acquiring new controller plug-ins, extending the animation capabilities of 3D Studio MAX.

This chapter explains techniques of working with controllers. See the Online Reference for details about the parameters and use of each controller type.

Understanding Controllers



Controllers in Track View and the Motion panel



Viewing controllers in Track View

Controllers are plug-ins that handle all of the animation tasks in 3D Studio MAX, including:

- Storing animation key values.
- Storing procedural animation settings.
- Interpolating between animation key values.

Most parameters don't receive a controller until you animate them. As soon as you change a parameter with the Animate button on, or click in its track using Track View Add Keys, a default controller is assigned to the parameter.

Accessing Controllers

There are two places in 3DS MAX where you work directly with controllers:

Track View—Controllers are indicated in the Hierarchy list by the green arrow icon. Using Track View you can view and work with the controllers for all objects and all parameters.

Motion panel—Contains special tools for working with transform controllers. The Motion panel contains many of the same controller functions as Track View, plus controls necessary for working with special transform controllers like the Look At and Path controllers. Using the Motion panel you can view and work with the transform controllers of a single selected object.

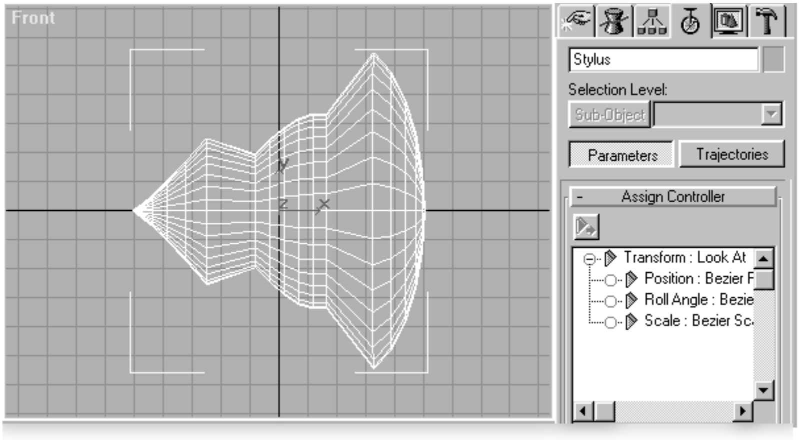
Categories of Controllers

There are two main categories of controllers in 3D Studio MAX. These categories are easy to identify when looking at the Track View Hierarchy list.

Single Parameter Controllers—Control animation values of a single 3DS MAX parameter. Regardless of whether the parameter has a single component, such as the number of sides of a cylinder, or multiple components, such as the RGB values of a color, the controller is handling a single parameter.

Compound Controllers—Combine or manage multiple controllers. Compound controllers in 3DS MAX include superior-level Transform controllers, such as Look At and PRS, the Euler XYZ Rotation controller, and the List controller.

Compound controllers appear in the Hierarchy list as a controller icon with subordinate-level branches of other controllers.



Viewing transform controllers in the Motion panel

Viewing Controller Types

You can view the controller type assigned to a parameter in both Track View and in the Motion panel. Before you can view the controller types in Track View, you must do two things:

- Select Controller Type display on the Track View Filters dialog.
- Assign controllers to parameters.

The Parameters mode of the Motion panel always displays the transform controller types for the selected object.

Reading Controller Types

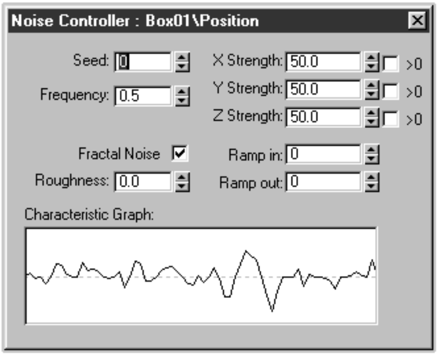
You can tell a lot about how a parameter is animated by looking at the controller type.

Parameter Name—Is always visible and is always to the right of the controller icon. It tells you what is animated.

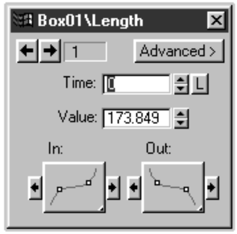
Interpolation Type—Usually follows the parameter name. It tells how animation values are calculated. The label Diffuse: Bezier Color indicates a Diffuse color parameter using Bezier interpolation with Color data.

Data Type—Usually follows the interpolation type. It tells what type of data is used. The label Height: Linear Float indicates a Height parameter using Linear interpolation with a floating point value.

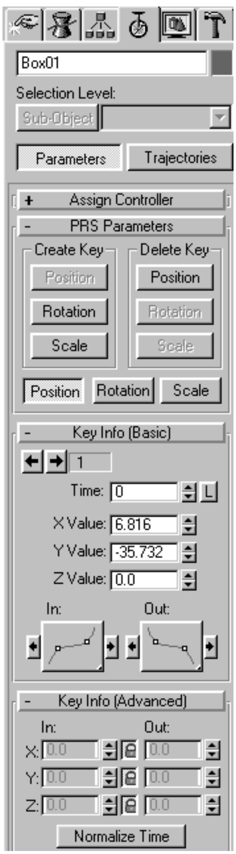
Changing Controller Properties



Global properties for the Noise controller



Key Info dialog in Track View



Key Info rollout
on the Motion panel

You can analyze and change your animation by editing controller properties. Most controllers display a properties dialog where you can change animation values. Whether or not a controller displays a properties dialog, and the type of information displayed, depends on the controller type.

Using Track View you can view controller properties dialogs for multiple tracks simultaneously. The following rules govern viewing multiple controller properties dialogs:

- Each track displays only one properties dialog in each Track View window.
- When properties dialogs for multiple tracks are visible, only one dialog can be active.
- Properties dialogs for tracks that use keys are disabled unless keys are selected.

Changing Controller Properties

Some controllers do not use keys, using instead a properties dialog that affects the entire animation. Such controllers are usually parametric controllers like Noise, or compound controllers like List.

To view controller properties in Track View:

1. Click in an empty area of the edit window to make sure no keys are selected.

2. Right-click the label for a parametric or compound controller, then choose Properties from the shortcut menu to display the properties dialog.

The dialog title identifies the controller type, the item or object name, and the parameter. For example: Noise Controller: Box01\Position

You can also view global properties for some transform controllers in the Motion panel.

Changing Controller Key Information

Some controllers interpolate between keys that you set on specific frames. Such controllers are always single parameter controllers such as a Bezier Float controller for Height or TCB for Rotation. These controllers use a Key Info dialog that contains settings for one of more selected keys.

To view key information in Track View:

- Right-click a selected key to display the Key Info dialog.

If more than one key is selected, Key Info displays common information for all of the selected keys. Settings that contain values indicate values common to all of the selected keys. Settings that are blank indicate values that change from key to key.

Viewing key information in the Motion panel always displays the settings for a single transform key.

To view key information for transform controllers in the Motion panel:

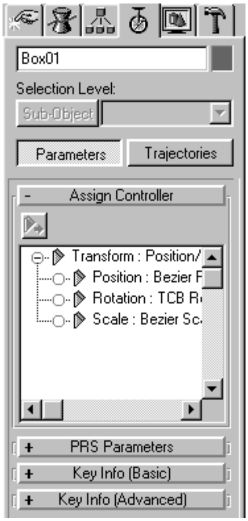
1. Select an object.
2. On the Motion panel, click Parameters.
3. Click Position, Rotation, or Scale on the Parameters rollout.

If the transform controller uses keys, Key Info rollouts appear below the Parameters rollout.

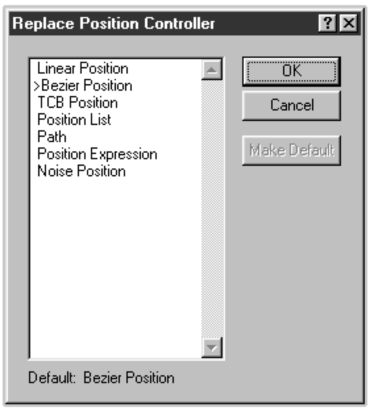
Changing Controllers That Do Not Display Properties

Some controller types do not display any properties in Track View or the Motion panel. You change the animation values of these controllers using the Animate button in the viewports and using the Key, Range, Time, and Function Curve tools in Track View.

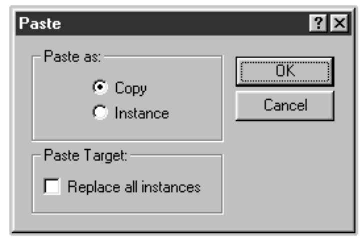
Assigning Controllers



Assign Controller rollout on Motion panel



Replace Controller dialog



Paste dialog

Every parameter has a default controller type that gets assigned the moment the parameter is animated. You can choose from multiple controller types for any parameter and change controllers after the parameter is animated.

Assigning Controllers in Track View

You can assign controllers to any parameter in Track View by selecting controller items and clicking the Assign Controller button in the Track View toolbar.

You can also assign the same controller type to a selection of multiple parameters as long as all the selected parameters can use the same type of controller. For example, you could select the Length, Width, and Height parameters for multiple Box objects and assign the same controller type to all of them. This is because they all use controllers that work on floating point data.

If a parameter has already been animated, then assigning a new controller has one of the following effects:

- The existing animation values are recalculated to produce a similar animation with the new controller. For example, replacing TCB Position with Bezier Position closely preserves the animation.
- The existing animation values are discarded. For example, replacing Smooth Rotation with Noise Rotation discards the Smooth Rotation animation values.

Assigning Controllers in the Motion Panel

You can assign controllers on the Motion panel by selecting an object and then, on the Assign Controller rollout, choosing a transform controller and clicking Assign Controller. You can only change the transform controllers of a single selected object.

Copying and Pasting Controllers

  Click Copy and Paste on the Track View toolbar to copy and paste controllers. For a general discussion of using Copy and Paste, see chapter 25, “Basic Track View Use.”

Rules for using Copy and Paste are as follows:

- You can copy only single controllers. Compound controllers like List or PRS Transform controllers are considered single controllers for Copy and Paste operations.
- You can paste a copied controller into one or more controllers using the same data type.
- You can choose to make an instance or a copy of the pasted controller.
- You can choose to convert other controller instances in the scene automatically.

Clicking Paste displays the Paste dialog, with three controls for determining how the Paste operation is carried out.

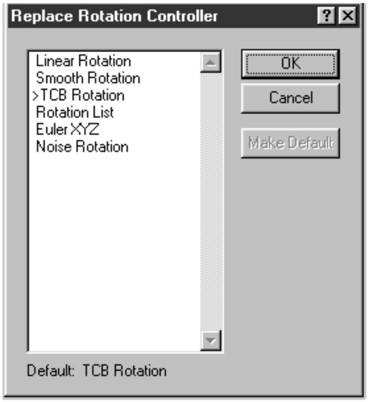
Copy—Pastes the controller as copy.

Instance—Pastes the controller as an instance of the source controller. Any change you make to either controller will affect the other.

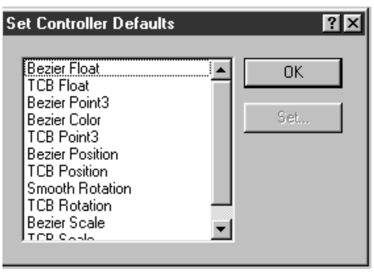
For example, you can paste a box’s Length controller as an instance into its Height and Width parameters. This makes the box a cube. Changing either of the Length, Width, or Height parameters changes the other two.

Replace All Instances—When selected, all instances of the target controller receive the paste controller, whether or not they are selected. This keeps all instances of the target controller as instances. When clear, the target controller is made unique and the remaining instances are unchanged.

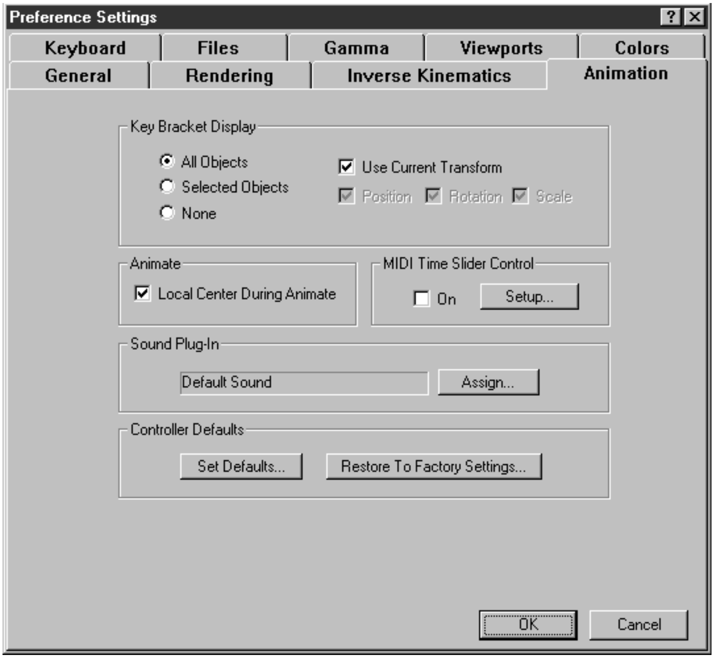
Specifying Default Controllers



Replace Controller dialog



Setting controller defaults



Controller default buttons in the Preference Settings dialog

You can specify the permanent defaults for controller types and controller settings to match the way you prefer to work. The following defaults are written to the *3dsmax.ini* file:

- The default controller used for each data type
- The default controller settings

Specifying Default Controllers

You specify default controllers by clicking Assign Controller on the Track View toolbar (see “Assigning Controllers”). When you choose a controller on the Replace Controller dialog you have the option to click Make Default before clicking OK.

Clicking Make Default assigns the chosen controller as the default for all parameters using that data type. It has the following effect:

- The default controller is listed at the bottom of the Replace Controller dialog.
- Many different parameters might share the same data type. For example, selecting the Length parameter of a Box and specifying Linear Float as the default sets the default controller for all parameters that use the Float data type. This includes, Width, Camera FOV, and Scale Deformation Curves.
- Default controller choices are written to the *3dsmax.ini* file and become the default for all new scenes.
- Previously assigned controllers are not affected.

Specifying Default Controller Values

You can specify the default settings for many controller types, or reset controllers to their factory settings on the Preference Settings dialog.

To specify default controller settings:

1. Choose File > Preferences.
2. On the Animation panel of the Preference Settings dialog, click Set Defaults of the to display the Set Controller Defaults dialog.
3. From the list of available controllers, choose a controller type and click the Set button to display the default settings supported by the selected controller. For example, the In and Out tangents for a Bezier controller.

Once you click OK, the controller defaults are changed. Changes to the controller default settings are written to your *3dsmax.ini* file and become the defaults for all newly assigned controllers and all new scenes.

You can also revert to the original 3D Studio MAX defaults for all controllers by clicking Restore to Factory Settings on the Animation panel of the Preference Settings dialog.

General Purpose Controllers

The following controllers are general purpose in that they are applied to parameters of different data types, yet they behave in essentially the same way for those different parameters.

Within certain general purpose controllers there may be variations according to the data type used by a parameter.

See the Online Reference for detailed descriptions of controller properties.

Bezier Controllers

Bezier controllers interpolate between keys using an adjustable spline curve; they are the default controller for most parameters.

Use Bezier controllers to provide fully adjustable interpolation between keys. Bezier is the only controller that supports the following:

- Adjustable tangent handles.
- Step tangents for abrupt changes from one key to the next.
- Constant velocity control.

TCB Controllers

TCB controllers produce curve-based animation like Bezier controllers. However, TCB controllers use fields to adjust the Tension, Continuity, and Bias of the animation.

Use TCB controllers when you want adjustable, curved interpolation between keys, and you want to use TCB style controls.

Linear Controllers

Linear controllers interpolate between animation keys by evenly dividing the change from one key value to the next by the amount of time between the keys.

Use Linear controllers when you want motion to have a rigid, mechanical look.

Noise Controllers

Noise controllers produce random, fractal-based animation over a range of frames. Noise controllers are parametric; they do not use keys.

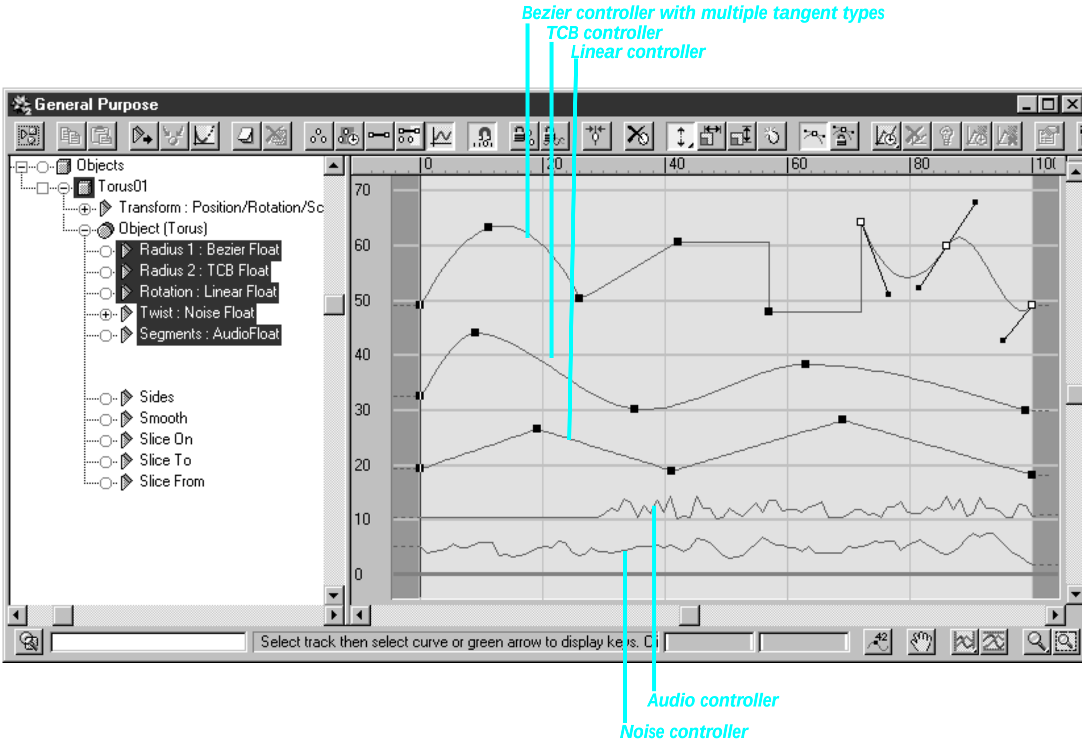
Noise controllers have many possible uses; as in the following examples:

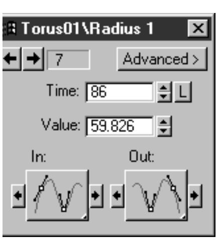
- Use Noise whenever you need completely random animation around a given value. For example, use a Noise Rotation controller when you want an object to wobble in place.
- Use Noise in a List controller to apply variations to the result of another controller. For example, use a List controller to combine Noise Position with Bezier Position. The Bezier controller moves the object while the Noise controller makes the object shake and stray a little from the trajectory.

Audio Controllers

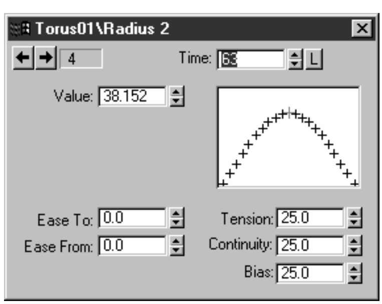
Converts the amplitude of a recorded sound file or real-time sound wave into values that can be used by an animated parameter.

Use the Audio controller to synchronize parameter values with a sound file. For example, use an Audio controller for a Multiplier Curve to scale a parameter in sync with a sound.

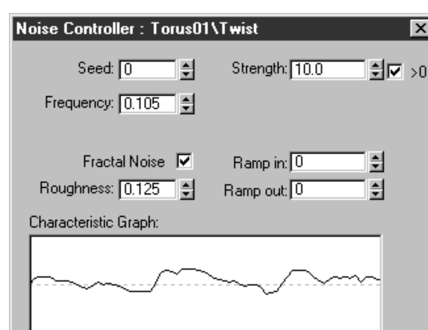




Bezier properties



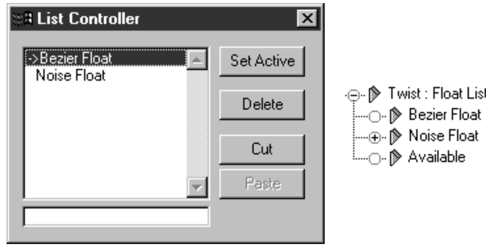
TCB Properties



Noise properties

Special Purpose Controllers

The controllers described in this topic are also applied to parameters of different data types, but they are used for special purposes. See the Online Reference for detailed descriptions of controller properties.



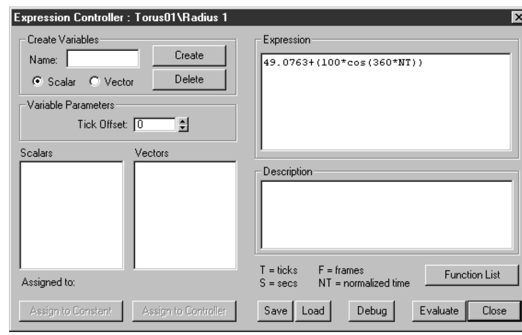
List controller

List Controllers

List controllers combine multiple controllers into a single effect. It is a compound controller with tools for managing the order in which its internal controllers are calculated.

Use List controllers to combine controllers as in the following examples:

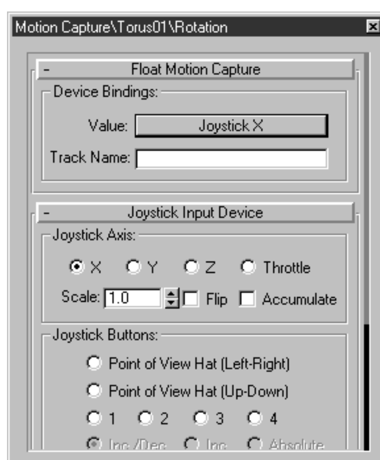
- Combine Noise Rotation and TCB Rotation controllers to introduce random orbital motion as an object rotates.
- Combine Bezier Position and Path Position to make an object follow a path with manually keyframed variation away from the path.



Expression controller

Expression/Script Controllers

You write custom code for Expression and Script controllers in an Expression or Script controller dialog. You specify parameters using mathematical expressions, functions, and variables. The code can include values based on the controllers of other objects in the animation.



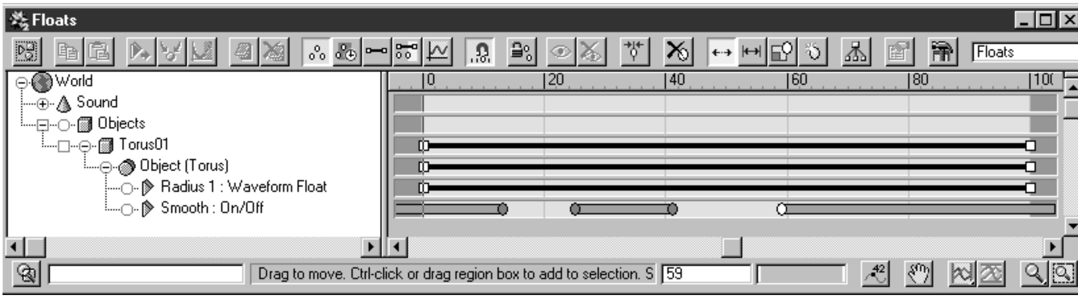
Motion Capture controller

Motion Capture Controllers

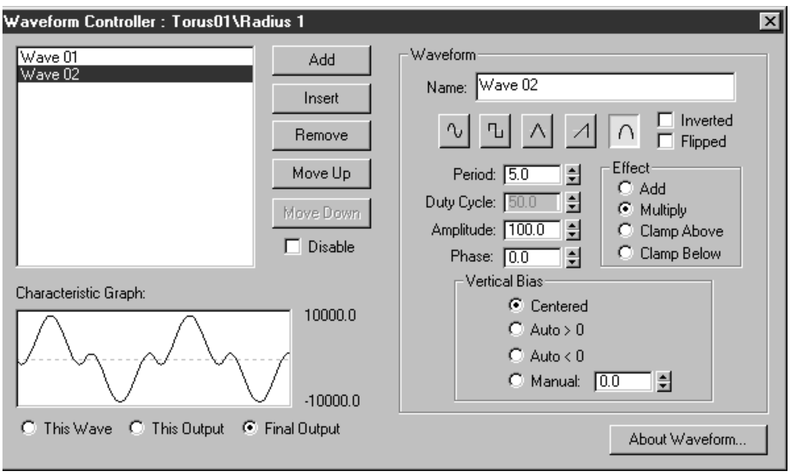
A Motion Capture controller controls parameters in real-time from the input of external devices. Currently supported devices are mouse, keyboard, midi device and joystick. Each device has specific properties that must be set:

- After assigning the motion-capture controller, open the controller Properties dialog and bind the type of peripheral device and set device parameters.
- Access the Motion Capture utility to test and record captured motion.

Float Controllers



On/Off controller



Waveform controller properties

The following controllers are available only for parameters using the Float data type.

On/Off Controller

An On/Off controls parameters using binary on and off values. The On/Off track displays a solid blue color in frames that are On, and the background in frames that are Off. The On/Off state of the parameter switches every time you add a key.

Use for binary parameters such as the Smooth object parameter, or for Visibility tracks.

Waveform Controller

A Waveform controller generates values by combining periodic waveforms. You can specify different types of waveforms and add waveforms together to create complex animation.

Waveform was originally created to control blinking lights. Use it for any value that you want to vary in a regular, oscillating pattern.

Controlling Transforms

Transform controllers are compound controllers. They set the type and behavior of the controllers used for Position, Rotation, and Scale.

You assign Transform controllers using either Track View or the Motion panel, but compound Transform controllers do not display properties in Track View. You access the parameters of Transform controllers only in the Motion panel.

Position/Rotation/Scale Controller

The Position/Rotation/Scale (PRS) controller is the default Transform controller for most objects. The PRS Transform controller sets up default Bezier Position, TCB Rotation, and Bezier Scale controllers.

Use a PRS controller whenever you want a standard transform set up or when you want complete control over the Position, Rotation, and Scale controllers.

Look At Controller

You assign the Look At Transform controller to make a specified local axis of any object point at the pivot of a target object. This controller uses standard Position and Scale controllers but the Rotation controller is restricted to Roll Angle only.

The use of a Look At controller is the default arrangement to make a camera or spotlight look at its target. You can also use a Look At controller for the following:

- Use a Look At controller to make any object track another object in the scene. For example, you can make a sign always point at another moving object.
- Change the Look At controller for cameras or spotlights by replacing their target with another object. This forces the camera or spotlight to track the new target object.

You pick another object to "look at" on the Look At properties rollout of the Motion panel. When you pick the target, the object using the Look At controller rotates to look at the target.

The orientation of an object using a Look At controller is undefined whenever its specified local axis is vertical in the World Coordinate System. This means that objects using a Look At controller have a tendency to flip when the look at target is directly above or directly below the object's pivot point.

Link Controller

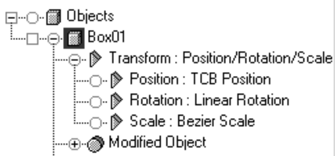
You assign a Link controller to an object to animate links from one parent to another. An example of using a link controller is to pass a ball from one hand to another. See chapter 23, "Building Hierarchies."

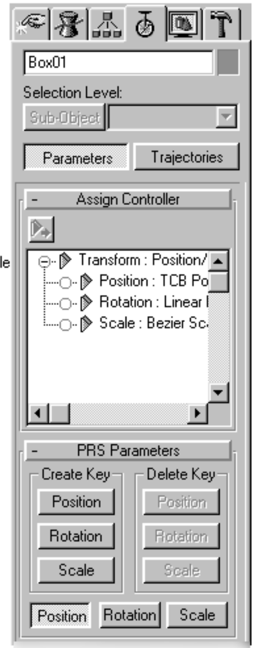
The Link controller keeps the object's original Transform controller as a subordinate track and adds the transforms of a selected parent object.

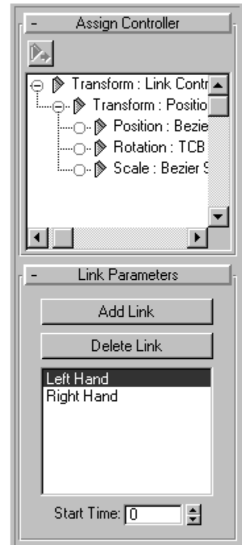
You add and delete links by clicking Add Link and Delete Link on the Link Parameters rollout on the Motion panel.

The Link controller includes these properties:

- The Link controller respects the link inheritance settings applied to the child object.
- The object using a Link controller is not a true child object. It does not appear in the subtree of any linked parent objects.
- Objects with Link controllers do not participate in IK solutions.

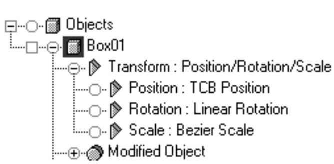


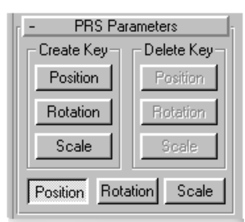




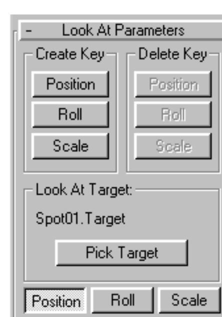
Link Controller Parameters rollout

Transform controllers in Track View and Transform controller properties in the Motion Panel



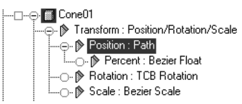


PSR Parameters rollout

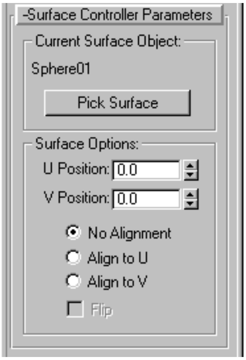


Look At Parameters rollout

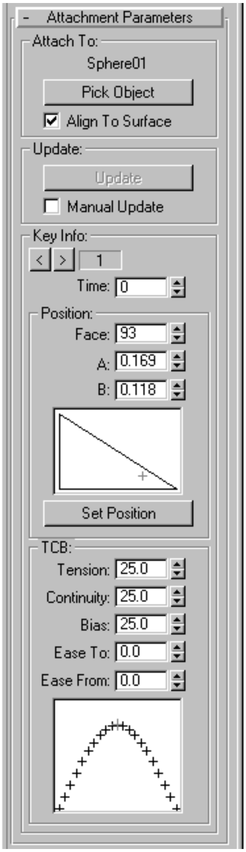
Controlling Position



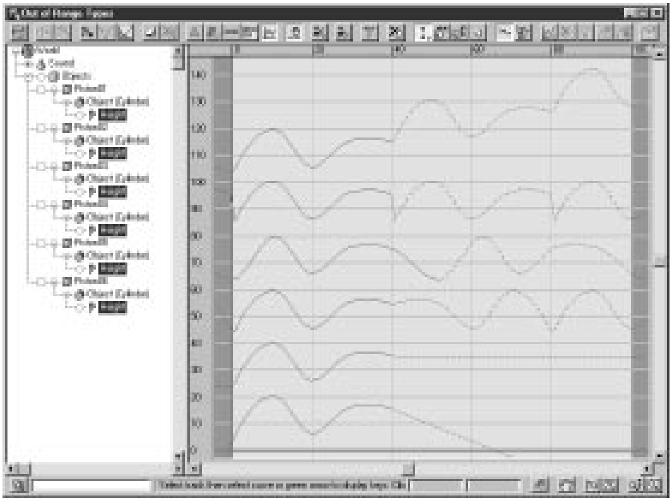
Path Parameters rollout



Surface Controller Parameters rollout



Attachment Parameters rollout



XYZ Position rollout

Below the Transform controller is the Position controller. Position is a data type that can use most of the standard controllers such as Bezier, TCB, and Noise.

The controllers described in this topic can only be used as position controllers.

XYZ Controller

An XYZ controller breaks a Position controller into three separate Bezier Float controllers. Each of the X, Y, and Z components of position receive their own track.

Use the XYZ controller when you want to have separate key patterns or controller types for each position component.

Path Controller

A Path controller locks object motion to a shape. You can edit and animate the path shape, and the object continues to travel along the shape.

- Use with Inverse Kinematics to define a path joint such as a key ring or slotted arc.
- Use to lock an object to a spline used as a loft path. The object travels along the length of the loft and if you change the loft path the object location also changes.

Path controller parameters display on the Path Parameters rollout of the Motion panel. You use these parameters to pick a Path shape and to set options for how the object follows and banks along the path.

Surface Controller

A Surface controller to locks object motion to the surface of a special parametric object. You can change and animate parameters of the surface, and the object continues to travel along the surface.

The surface object must be one of the following object types: Sphere, Cone, Cylinder, Torus, Single Quad Patch, Loft, or NURBS surface. The surface object cannot have any modifiers that convert to a mesh in its modifier stack.

- Use with Inverse Kinematics to define a surface joint. The surface must be set to No Alignment.
- Use to lock an object to a surface. The object travels across the surface and if you change the surface the object location also changes.

Surface controller parameters display in the Surface Controller Parameters rollout of the Motion panel. You use these parameters to pick a Surface object and to set options for how the object aligns to the surface.

Attachment Controller

Assign an Attachment controller to cause an object to hold a position on the surface of another object. The Attachment controller has the effect of gluing an object to the surface of another object.

Use the Attachment controller to attach a “source” object to the face of a “target” object so that the source object acts as if it's glued to the target object, no matter how the surface of the target object is deformed. You can animate the Attachment parameters so that the source object moves over the surface of the target object.

The Attachment and Surface controllers have two important differences:

- Only the Attachment controller works with deformed mesh objects.
- Only the Surface controller works with IK.

Controlling Rotation

Below the Transform controller is the Rotation controller. Rotation is a data type that can use most of the standard controllers such as TCB, Linear, and Noise.

Rotations in 3D are very complex. Even the standard controller types behave a differently when used for rotation. The most common way of calculating rotations in 3D animation uses four components to define rotation about an arbitrary axis. This is the *quaternion* method.

The benefits of using quaternion rotation include:

- They produce a direct one-to-one relationship between the quaternion values and how objects interactively rotate in the scene.
- They produce smoother rotation than other methods.

The drawbacks of using quaternion rotation include:

- Rotation values in key information can be difficult to understand.
- Quaternion rotation controllers do not display function curves in Track View.

The following controllers can only be used as rotation controllers.

Smooth Rotation

Use Smooth Rotation when you want rotation to have a smooth and natural look. Smooth rotation uses nonadjustable curved interpolation, and has the following characteristics:

- You can move keys in Track View to change timing.
- You can directly rotate objects in the viewports to change rotation values.
- You cannot display controller or key properties, or function curve.

Euler XYZ Rotation

Use the Euler XYZ rotation controller when you do not want to use the quaternion method. Euler XYZ is a compound controller that combines separate, single-value float controllers to specify an angle of rotation about each of the X, Y, and Z axes.

Euler XYZ is not as smooth as quaternion rotation, but it is the only rotation type that you can use to edit rotation function curves.

Euler angles are well suited for animating mechanical rotations. They can also be a good choice for an object in an Inverse Kinematic chain because IK Rotational joints are defined as Euler angles.

Controlling Colors

You can animate colors in 3D Studio MAX just like most other parameters. There are two data types used for controlling colors.

Point3 —A general purpose, three-component data type that works with RGB color values. It uses most of the standard controllers.

Color—A special data type designed specifically for working with RGB and HSV color values. Color uses the Bezier and RGB controllers.

Color Point3 Controller

When using Point3 controllers for color parameters, an important issue is behavior of the Key Value fields, labeled X, Y, Z. They store color values using only the RGB color model.

- The X value field stores Red color values.
- The Y value field stores Green color values.
- The Z value field stores Blue color values.

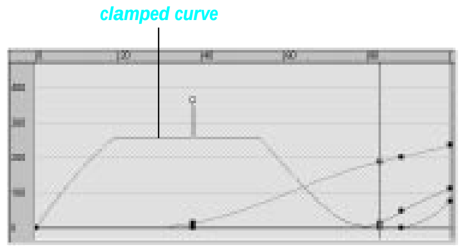
The Point3 Key Value fields do not clamp at the valid color range of 0 to 255. Values that drop below 0 or exceed 255 are ignored by the color parameter but are still displayed in Track View.

Bezier Color Controller

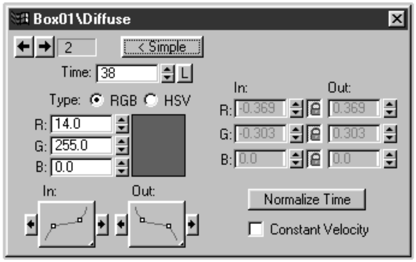
The Bezier Color controller is a data type using Bezier key interpolation. You can choose to work with RGB or HSV color models with the Bezier Color controller.

The choice of color model is global for all keys used by the controller. You can switch between color models anytime and color key values are correctly converted.

The Color Value fields are limited to a range of 0 to 255. You can drag the color keys above 255 in Track View Function Curves mode, but the values are clamped at 255.



Clamping Color Value fields



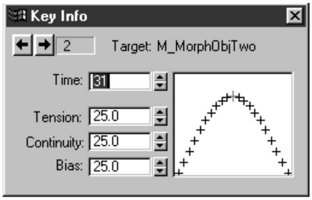
Color controller Key Info dialog Morph Controllers

Color RGB Controller

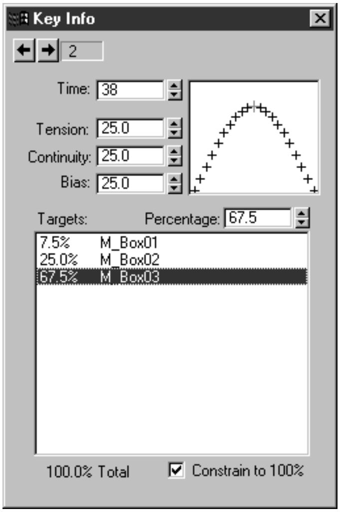
Assign a Color RGB controller to break a Color controller into three separate Bezier Float controllers. Each of the R, G, and B components of color use their own track.

Use the Color RGB controller when you want to have separate key patterns or controller types for each color component.

Morph Controllers



Cubic Morph Controller Properties



Barycentric Morph Controller properties

You can choose from two morph controllers. These controllers manage how morph targets blend from one target to another.

Cubic Morph Controller

A Cubic Morph controller is a TCB-style controller. It uses Tension, Continuity, and Bias controls much like the generic TCB controller.

The Cubic Morph controller manages only the interpolation from one morph target to the next. If you want to add Morph keys or change the morph target, you must use the Pick Targets and Current Targets rollouts in the Modify panel.

Barycentric Morph Controller

The Barycentric Morph controller is also a TCB controller like the Cubic Morph controller, except that instead of each key representing a single target, each key represents a series of weights for all targets. A Barycentric Morph key represents a new object which is a blending of all targets.

You can adjust each morph key to various percentages of the available morph targets, creating subtle adjustments in the animation.

You can also click Add Keys in Track View to create new Barycentric Morph keys. The added keys contain interpolated values for all targets.

Chapter

28

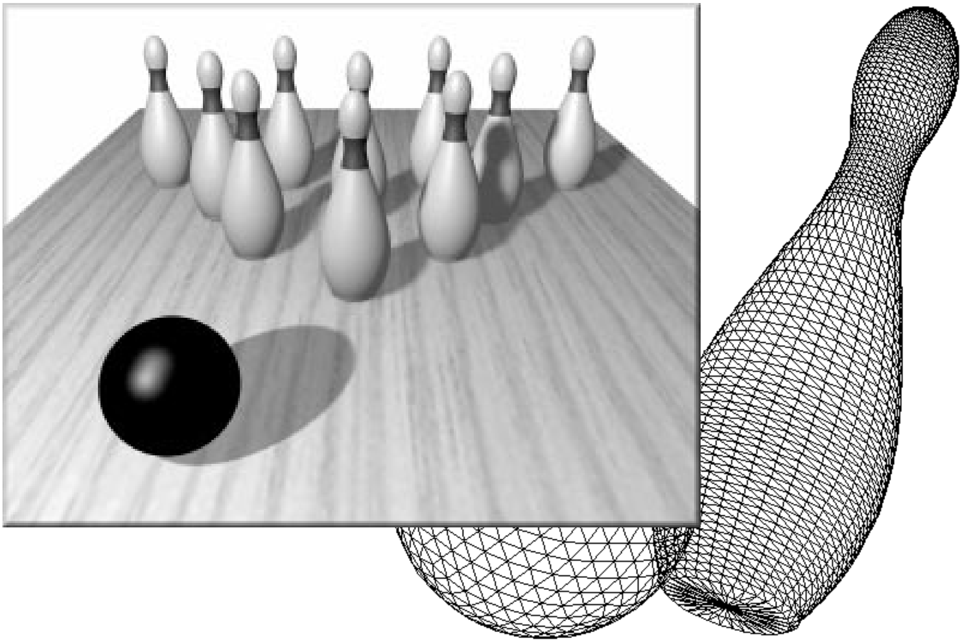
Animating Dynamic Simulations

3D Studio MAX has a Dynamics utility that you can use to simulate real-world physics in your animation. If you use manual animation techniques, for example, to animate a bowling ball falling down stairs, you have to decide how the ball falls and set all the keys. If you use the Dynamics utility instead, you apply physical properties to the ball and stairs, add gravity, and let 3DS MAX do the calculations.

You have the greatest amount of control and artistic freedom with the manual method, but it takes a lot of work. A good compromise is to use the Dynamics utility to solve the tedious work of animating the basic motion and then manually edit the resulting keys to provide extra effects.

See the Online Reference for details about Dynamics properties.

How Dynamics Works



The Dynamics utility works by calculating all of the forces applied to each object in the simulation, checking for collisions with other objects, and then assigning Position and Rotation keys to objects as indicated by the calculations. This process is repeated for every frame of the range you specify for the dynamic simulation.

3D Studio MAX employs a method called rigid body dynamics to create a simulation in which objects do not deform as a result of any force or collision. You can add deformation to your animation later by manually editing keys after the dynamic simulation or by using a soft body deformation plug-in such as Hypermatter.

Components of a Simulation

You specify the following components to control the results of a dynamics simulation:

Physical properties—Object properties such as bounce, friction, and mass. A rubber ball bounces high and has a "sticky" surface. A bowling ball bounces low and has a slippery surface.

External effects—Forces, such as gravity or torque, that act on objects in a simulation. You create space warps and assign them to a simulation to generate effects.

Collisions—Reactions between objects that touch during a simulation. Collision results depend on the speed and direction of objects and their physical properties. You specify which objects are considered for collision calculations.

For a bouncing ball simulation you specify that the ball collides with the floor.

Basic Process of Using Dynamics

Once you have constructed a scene, you generate a dynamics simulation by performing the following steps:

1. Create space warps to generate external forces for the simulation.
2. Name a new dynamics simulation and specify which objects participate in the simulation.
3. Assign physical properties to objects in the simulation. You assign properties for materials on the Dynamics Properties rollout of the Material Editor, or for entire objects on the Dynamics rollout of the Utility panel.
4. On the Link Info rollout of the Hierarchy panel, set Move and Rotate locks to limit the motion and rotation of linked objects.
5. On the Dynamics rollout, specify which effects influence which objects, and which objects collide with other objects.
6. Specify the range of frames covered by the simulation, set calculation parameters, and click Solve to calculate the solution and generate animation keys.
7. Play the animation to see if the effect is acceptable. Repeat steps 3 through 7, to fine-tune the simulation if one or more objects fly off into space, drift through objects, or otherwise move incorrectly.

Caveat: Your Results May Vary

No Dynamics simulation is ever completely stable. There are too many possible conditions and some problem conditions cannot be solved or detected in advance. This chapter describes how to increase your chances for success by avoiding certain error situations.

Setting Up a Dynamic Simulation

There are few basic parameters for defining a dynamic simulation. Most of your work involves defining the physical properties of objects in the simulation.

Creating a New Simulation

Your first task is to create a new named simulation to store in your 3DS MAX scene. A scene can contain any number of named simulations.

To create a new named simulation:

1. On the Utility panel, click More and then choose Dynamics.
2. On the Dynamics rollout, click New and then enter a name for the simulation on the Simulation Name list.

All of the simulations you create in a scene can be selected from the Simulation Name list.

Simulations are stored in your scene until you remove them by clicking Remove. When you remove a simulation, all of the parameters are deleted, but any keys that were generated by the simulation remain.

Adding Objects to the Simulation

You add objects to a named simulation by clicking Edit Object List on the Dynamics rollout. The Edit Object List dialog works much the same way as the Light Include/Exclude dialogs. You select from a list of objects in your scene and specify whether the selected objects should be included or excluded from the simulation.

Assigning Effects to the Simulation

You can choose one of two methods for assigning and space warp effects to a simulation. Only one method can be active for a simulation.

Effects By Object—Use when you want to apply space warp effects to some, but not all, of the objects in your scene. You must specifically state which effects apply to which objects.

Global Effects—Use when you want all space warp effects in the simulation to apply to all objects.

See “Preparing Dynamic Effects” for information on assigning effects.

Assigning Collisions to the Simulation

You can choose one of two methods for assigning and calculating collisions in a simulation. Only one method can be active for a simulation.

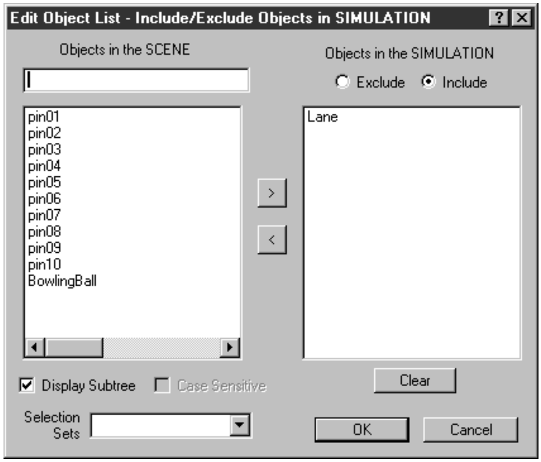
Collisions By Object—Use when you want to explicitly define which objects can collide with which other objects. Using this method, you have greater control of collisions.

Global Collisions—Use when you want all selected objects to collide with each other.

See “Defining Object Collision Parameters” for information on collisions.

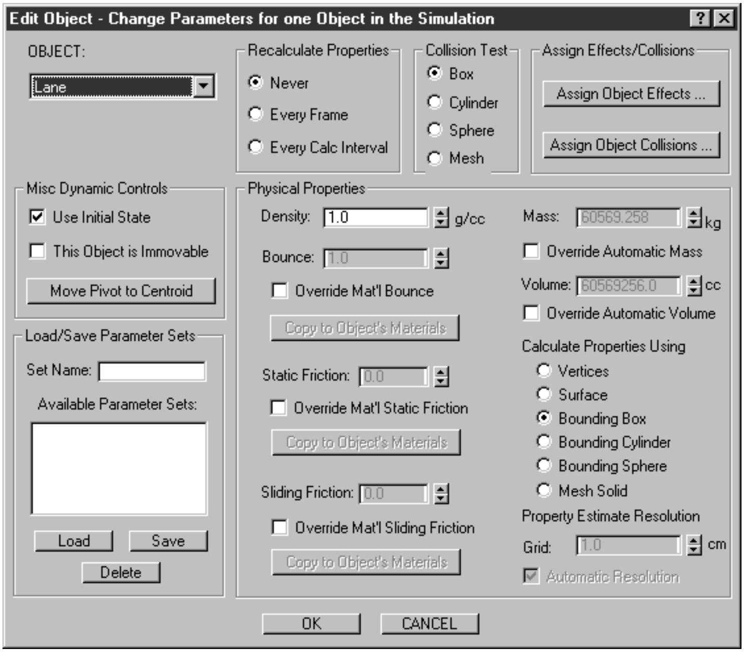
Defining Air Resistance

You set the Density percentage value to specify the amount of air resistance in the simulation.



Selecting objects to include in a simulation

Preparing Objects for a Dynamic Simulation



When you prepare objects for a dynamic simulation, you set different kinds of properties.

- Some properties are controlled from the Dynamics rollout on the Utility panel.
- Other properties are general conditions of your scene.

This topic, and following topics, describe how to prepare objects for a dynamic simulation.

Setting Dynamic Properties

You set many object properties on the Edit Object dialog after the objects have been added to the dynamic simulation.

To set dynamic properties for an object:

1. On the Dynamics rollout, click Edit Object to display the Edit Object dialog.

2. Select the name of an object from the Object List, then set properties for that object.

Setting Object Pivots

Dynamic simulations do not solve correctly for objects that have a rotated pivot. If any objects in a dynamic simulation have rotated pivots you must reset the pivot transform using Reset Transform on the Hierarchy panel.

Moving a pivot will not affect the solution of a dynamic simulation. However, when the pivot is moved away from an object's center of mass, it slows the solution of the dynamic simulation.

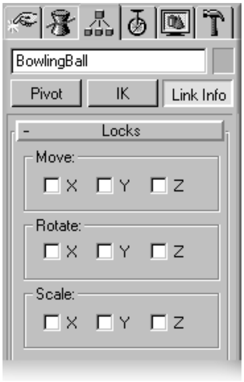
See chapter 23, "Building Hierarchies," for information about adjusting pivots.

Aligning Objects

Objects that have their edges perfectly aligned set up conditions that may be impossible to solve. Such conditions include a straight row of dominoes, or stacked, identical boxes with aligned edges.

If you have objects with aligned edges, and the simulation does not solve correctly, you should move or rotate the objects slightly out of alignment. A small offset in alignment allows the simulation to solve correctly.

Linking Objects



If you include linked hierarchies in a simulation, you must set Lock options to confine the motion of child objects. You set these options on the Lock rollout of the Hierarchy panel.

The Dynamics utility uses Locks for linked hierarchies to provide the constraints similar to active and inactive axes in IK Joint Parameters. See, “Defining Object Joint Parameters.”

Binding Objects to Space Warps

Whether or not you should bind a space warp to objects in a dynamic simulation depends on the type of space warp:

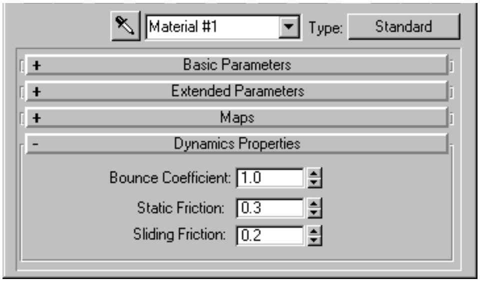
- Bind Geometric/Deformable and Modifier-Based space warps to deform objects. Objects deformed by space warps may behave unexpectedly because the dynamics solution is applied before the space warp deformation.
- Do not bind Dynamics Effects space warps. Assign Effects on the Dynamics rollout.

Setting the Initial State of an Object

If you have manually animated objects that participate in a simulation, you must decide how to interpret the initial state of the object.

- Select Use Initial State on the Edit Object dialog to solve the motion and inertia of an object at the start of a simulation.
- Clear Use Initial State if the object is assumed to be sitting motionless at the start of a simulation.

Defining Object Material Properties



Dynamics Properties on the Material Editor

You set properties to define how much friction exists between two objects and how much energy is transmitted they collide. The dynamic effect of friction and bounce is a function of the values assigned to both of the colliding objects.

Defining Properties by Material

You can set friction and bounce values as material properties on the Dynamics Properties rollout on the Material Editor. You then assign those properties to an object in a dynamic simulation by assigning the material to the object.

Because Dynamics Properties are part of a standard material type, you can set different properties for each submaterial of compound materials like Multi/Sub-Object. Use a Multi/Sub-Object material to assign separate friction and bounce values to different faces of an object. For example, you could define a soft rubber ball with a hard steel ring at its equator.

The default values define a surface similar to Teflon-coated hardened steel.

Defining Properties by Object

To override the friction and bounce properties of a material, click Edit Object on the Dynamics rollout on the Utility panel and then set friction and bounce values on the Edit Object dialog. These values override the material values and always apply to the entire object.

Setting Static Friction

You set Static Friction to define how difficult it is for an object to start sliding across a surface. High Static Friction values make it more difficult for an object to start moving. If a box weighs ten pounds and sits on Teflon (a static friction of near 0.0), it takes almost no force to slide the box. If the box sits on sandpaper (a high static friction of 0.5 to 0.8), it takes much more force to slide the box.

Setting Sliding Friction

You set Sliding Friction to define how difficult it is for an object to continue sliding across a surface. High Sliding Friction values make it more difficult for a sliding object to continue sliding.

Once objects begin to move, sliding friction replaces static friction. Sliding friction is usually much lower than static friction. For some materials, such as brake linings, sliding friction is just as high as static friction.

Setting Bounce

You set Bounce Coefficient to define how far an object bounces after hitting another object. High Bounce values cause objects to bounce farther. Bounce defines how much energy is transmitted when objects collide, a value of 1.0 represents a collision in which all energy is transmitted between colliding objects.

If you've seen the toy with four ball bearings swinging back and forth hitting one another, you've seen an example that comes close to a bounce coefficient of 1.0. Hardened steel or a super ball has a bounce near 1.0, while lead has a bounce near 0.0.

If you want to completely eliminate bouncing, both objects in the collision must have Bounce Coefficients of 0.0.

Defining Object Joint Parameters

When you manually animate linked objects, motion of a parent affects all descendents but motion of a child has no effect on its ancestors. If you want a hierarchy to represent a jointed structure, you constrain the motion of child objects in relation to their parent.

Defining Joint Types With Locks

To constrain the motion of selected objects, click Link Info on the Hierarchy panel and select options on the Lock rollout. You can constrain moving, rotating, and scaling on each of the object's local axes. Dynamics ignores Scale locks and uses the Move and Rotate locks to define joint constraints.

- When a lock is selected, a child object cannot move or rotate on the locked axis with respect to its parent. Locked axes transmit force or torque from child to parent.
- When a lock is clear, a child object moves or rotates freely on the clear axis. Cleared axes do not transmit force or torque.

The following list describes some useful combinations of Move and Rotate locks.

Fixed Axle / Hinge—All Move locks selected and two Rotate locks selected. This joint does not move and rotates on one axis. It transmits force on three axes and transmits torque on two axes perpendicular to the axis of rotation.

Ball joint—All Move locks selected and all Rotate locks clear. This joint does not move but rotates in any direction. It transmits force but does not transmit torque.

Sliding Track—Two Move locks selected and all Rotate locks selected. This joint slides in one direction, like a cabinet drawer, and does not rotate. It transmits force in two directions perpendicular to the slide axis and transmits torque on all axes. You can use this joint with the Push space warp to make a hydraulic cylinder.

Universal joint—All Move locks selected and one Rotate lock selected. This joint cannot move and rotates on two axes, like the joint between the rear axle and drive shaft of a car. It transmits force on three axes and torque on one axis.

Sliding Universal—Two Move locks selected and one Rotate lock selected on the axis that does not match the selected Move locks. This joint can slide in one direction and rotate on two axes, like the spline shaft between the transmission and drive shaft of a car. It can transmit torque on one axis and transmit force on two axes perpendicular to the axis of rotation.

Hockey Puck—One Move lock selected and two Rotate locks selected where neither Rotate lock matches the axis of the Move lock. This joint slides on the surface of a plane and it rotates about an axis perpendicular to the plane. It transmits force along the axis of rotation and transmits torque on the planar axes.

Sliding Axle—Two Move locks selected and the matching two Rotate locks selected. This joint slides in and out of a hole, and rotates on the axis of the hole. It transmits force and torque on the two axes perpendicular to the hole.

Welded—All Move and Rotate locks selected. This is a completely rigid joint that transmits force and torque on all axes.

Using IK Limits and Damping

You can have the Dynamics utility consider IK Joint Limits and Damping when solving a simulation. On the Simulation Controls area of the Dynamics rollout, select Use IK Joint Limits and Use IK Joint Damping.

See chapter 24, "Using Inverse Kinematics," for information on joint parameters.

Defining Object Mass

You define the mass of objects in your scene by setting Density value and a volume calculation method on the Edit Object dialog. Dynamics then calculates the mass of an object as density times volume. You can also set explicit volume or mass values to avoid the need for calculation.

To define object mass:

1. On the Dynamics rollout, click Edit Object.
2. On the Edit Object dialog, choose the object you want to edit from the Object list, then set the Density, Volume, or Mass parameters.

Mass affects how quickly an object reacts to force and how much force the object transmits when it collides with another object.

- Objects of low mass react quickly to small forces and do not transmit much force in a collision unless they are moving very fast.
- Objects of high mass react slowly to most forces and transmit great force in a collision.

Setting Density

You set the Density value in grams per cubic centimeter (g/cc). A setting of 1.0 g/cc is similar to the density of water and is a good choice for wood, plastic, or organic objects.

Defining Volume

You can choose from a number of methods for defining the approximate volume of an object.

Vertices—Assigns a volume of 1.0 cubic centimeter to each vertex of an object. Mass is based on the number of vertices. Mass distribution collects at the area with the greatest number of vertices. Use for objects with a uniform pattern of vertices or dense vertices near where you want to locate the center of mass.

Surface—Approximates volume by multiplying surface area by a thickness of 1.0 centimeter. Mass increases with the number of wrinkles and deformations on the surface. Mass distribution collects at the area with the greatest amount of surface deformation. Use for irregularly shaped objects with uniform surface deformation.

Bounding Box/Cylinder/Sphere—Approximates volume as a primitive surrounding the object extents. Mass is based on overall size. Mass distribution is uniform throughout the bounding primitive. Use for objects that match the shape of the bounding primitive.

Mesh Solid—Approximates volume by calculating object geometry. Mass and Mass distribution are calculated accurately. Use when you need the greatest accuracy.

Note: Mesh Solid cannot calculate objects with holes in their surface. Use the other volume choices for such objects.

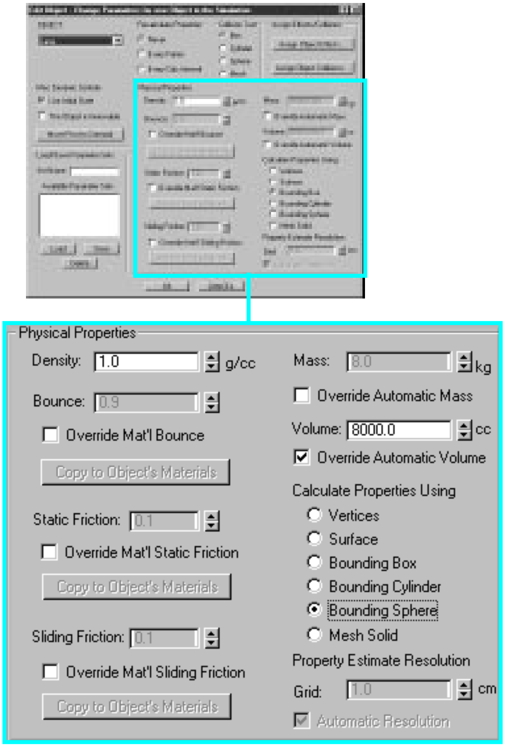
Your choice for defining volume affects mass in the following ways:

Mass Value—Mass is the result of density times volume. Your choice for defining volume affects the accuracy of mass value calculation.

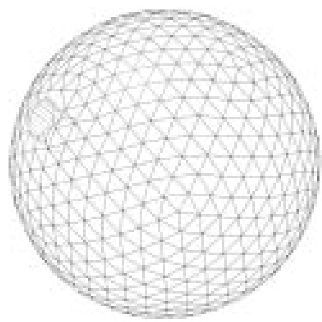
- Vertices is the most abstract definition.
- Mesh Solid is the most accurate.

Mass Distribution—The distribution of mass determines the center of rotation of an object—that is, its center of mass—and affects how the object reacts to torque. Your choice for defining volume affects the distribution of mass.

- Vertices and Surface distribute mass based on object topology.
- Bounding options distribute mass uniformly within the bounding shape.
- Mesh Solid accurately distributes mass based on the true form of an object.



Physical Properties area in Edit Object



Setting Frequency of Mass Calculation

You specify how often to calculate mass and mass distribution by setting options in the Recalculate Properties area on the Edit Object dialog.

Recalculating properties accounts for changes in mass and mass distribution caused when an object changes shape. You do not need to recalculate properties if objects do not change shape during the simulation.

Overriding Volume

You can override volume calculation and specify an explicit volume for an object. Select Override Automatic Volume on the Edit Object dialog and enter a Volume value.

Using this option ignores the recalculation setting because volume and mass never change.

Overriding Mass

You can override mass calculation and specify an explicit mass value for an object. Select Override Automatic Mass on the Edit Object dialog and enter a Mass value.

You can also select This Object Is Immovable on the Edit Object dialog. The object cannot be moved by any effect or collision.

Using either option ignores recalculation, volume, and density settings.

Defining Object Collision Parameters

Collision parameters control which objects collide with other objects and how Dynamics detects collisions. Most motion created by a simulation results from detecting and calculating collisions.

Setting Collision Type

You select an option in the Collision Test area on the Edit Object dialog to set how Dynamics detects collisions.

Box/Cylinder/Sphere—Detects collisions against a primitive surrounding the extents of an object. Select these options for objects that roughly match the bounding primitive shapes. Bounding primitive calculations are fast.

Mesh—Detects collisions against the true surface of the object. This option is the most accurate, but takes longer to calculate. Select this option when your object is too complex to work with the bounding options.

Note: Mesh is the only option that works with bounce and friction properties assigned through Multi/Sub-Object materials.

Setting Collisions with Other Objects

To identify which objects in a simulation an object can collide with, click Assign Object Collisions on the Edit Object dialog. The Assign Object Collisions dialog is an Include/Exclude dialog where you choose the names of objects the current object can collide with.

- An object will pass through any other object not included in its collision list.
- Sliding and resting is a form of collision. If you want a box to sit or slide on top of a table, you must include the table in the collision list for the box. The surfaces of sliding or resting objects must be aligned for the simulation to solve correctly.
- Hierarchical linking greatly increases the complexity of collision calculations. Linked objects involved in a collision may destabilize and break their links.
- Allowing collisions between a parent object and any of its ancestors in a hierarchy can create unsolvable conditions for dynamics.

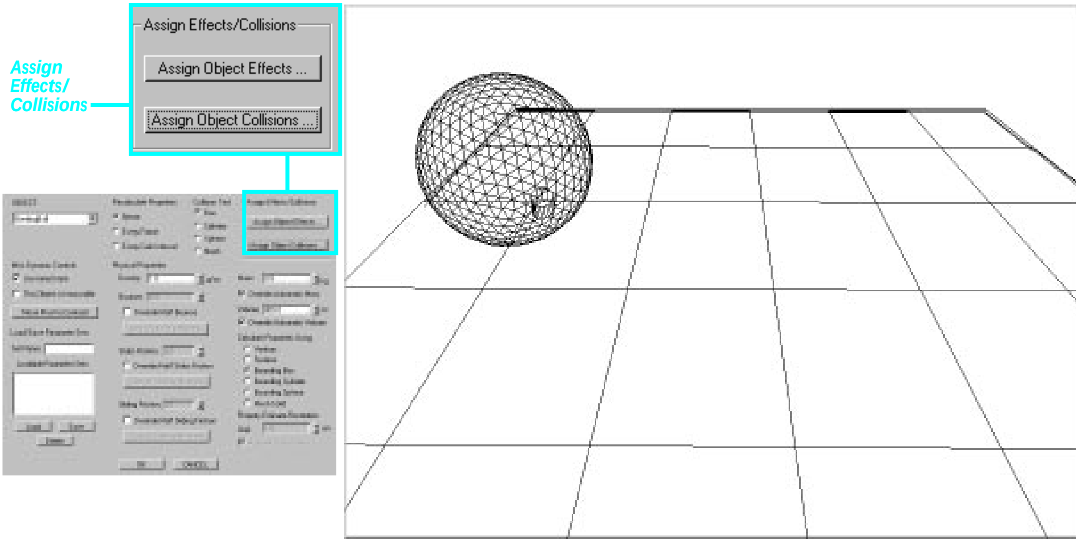
To use object collisions, select Collision By Object on the Dynamics rollout.

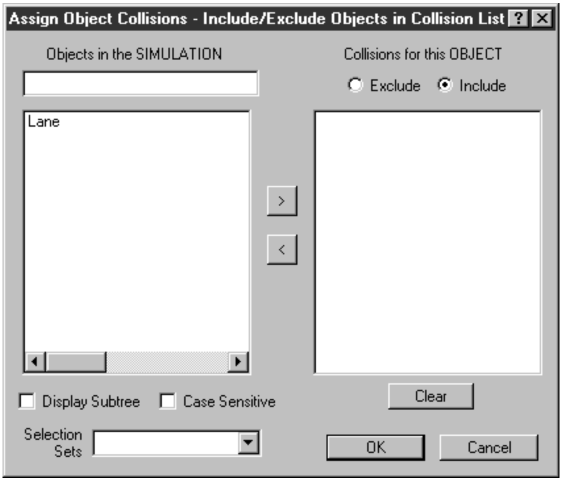
Setting Global Collisions

You can assign global collisions as an alternative to assigning collisions for each object. Click Assign Global Collisions on the Dynamics rollout to display the Assign Global Collisions dialog. This is an Include/Exclude dialog where you choose objects for the global collision list.

- Objects on the global collision list can collide with all other objects on the list.
- Global collisions can greatly increase the number of collision calculations required. Use this method only with simple simulations or simulations where most objects will collide with all other objects—for example, ball bearings poured into a jar.

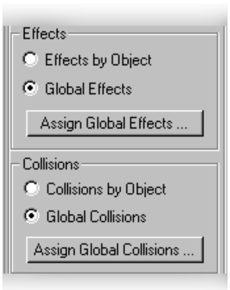
To use global collisions, select Global Collision on the Dynamics rollout.





Assign Object Collisions Dialog

Preparing Dynamic Effects



You create effects such as gravity, wind, or torque by creating space warps and assigning them to a dynamic simulation. You can only assign space warps that support Dynamic Effects. The default space warps that support dynamic effects are Gravity, Wind, Push, Motor, and Particle Bomb.

Like collisions, you can assign effects individually by object, or globally for all objects.

Assigning Effects to Objects

You identify which effects can act on an object by clicking Assign Object Effects on the Edit Object dialog. The Assign Object Effects dialog is an Include/Exclude dialog where you choose the names of effects which act on the current object.

Use this method whenever you have effects that you want to limit to a few objects in the simulation. To use object effects, select the Effects By Object option on the Dynamics rollout.

Assigning Effects Globally

You can assign global effects as an alternative to assigning effects for each object. Click Assign Global Effects on the Dynamics rollout to display the Assign Global Effects dialog. This is an Include/Exclude dialog where you choose the names of effects which act on all objects in the simulation.

Use this method when you want the effects in your simulation applied to all objects. To use global effects, select Global Effects on the Dynamics rollout.

Creating Natural Effects

You create the effects of natural forces using Gravity and Wind space warps. See the Online Reference for detailed descriptions of Gravity and Wind parameters.

Assigning Gravity

You create a Gravity space warp to generate constant acceleration. The effect of Gravity is uniform on all objects regardless of size, mass, or mass distribution.

If you create a Gravity space warp using default settings you get an effect like Earth gravity pulling objects to the bottom of your scene.

- Create a planar Gravity space warp to generate a uniform directional force. If you are standing on Earth, gravity seems planar and pulls everything down towards the ground.
- Create a spherical Gravity space warp to generate a uniform radial force. If you are out in space, gravity seems spherical and pulls everything towards the center of a planet.
- Rotate a planar Gravity space warp to simulate other forces such as magnetism.

Assigning Wind

You create a Wind space warp to generate a turbulent force. The effect of Wind varies for objects depending on surface area, mass, and mass distribution.

- Create a planar Wind space warp to generate a turbulent directional force. This effect is similar to wind blowing across the surface of the Earth.
- Create a spherical Wind space warp to generate a turbulent radial force. This effect is similar to air blowing out of a vent.
- Create Wind space warps to simulate other turbulent forces such as flowing water.

Creating Mechanical Effects

You create mechanical effects such as push rods and torque using Push and Motor space warps. See Online Reference for detailed descriptions of Push and Motor parameters.

Pushing Objects

You create a Push space warp to apply a point force to an object. The effect of Push varies for objects depending on mass and mass distribution.

You can use Push as follows:

- Assign Push to a hydraulic cylinder to raise a platform.
- Assign Push as a thruster to move a jet plane or speed boat.

You usually position a Push space warp with respect to a single object and then link it to that object for predictable results.

- If Push is aligned with an object's center of mass it, applies pure force, moving the object in the direction of the Push space warp.
- If Push is offset from an object's center of mass, it applies a combination of force and torque moving and rotating the object.
- If the Push force is not great enough to overcome mass and friction, the object will not move.

You can also set Feedback and Period Variation parameters to vary the Push force:

- Select Feedback to limit the speed of pushed objects. Push force can either taper off of reverse as the object reaches the target speed.
- Select Period Variation to randomly vary the Push force. Use it to generate vibration effects.

Rotating Objects

You create a Motor space warp to apply rotational force—torque—to an object. The effect of Motor varies for objects depending on their mass and mass distribution.

- Motor always rotates an object about its center of mass regardless of where the Motor space warp is located.
- The axis of rotation is aligned with the local Z axis of the Motor space warp.
- If the Motor torque is not great enough to overcome mass and friction, the object will not rotate.

You can also use Feedback and Period Variation parameters to vary the rotational force:

- Select Feedback to limit the revolutions of motored objects. Motor torque can either taper off or reverse as the object reaches the target revolutions.
- Select Period Variation to randomly vary the Motor torque.

Creating Explosions

You create a Particle Bomb to generate explosive effects in a dynamic simulation. Particle Bomb works on geometric objects when assigned to a dynamic simulation. See the Online Reference for detailed descriptions of Particle Bomb parameters.

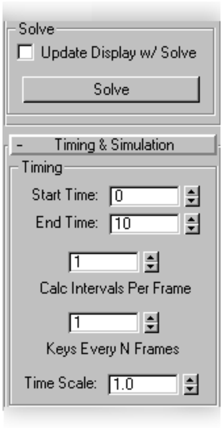
The effect of Particle Bomb varies for objects based on their mass. If the strength of the Particle Bomb is not sufficient to overcome an object's mass, nothing happens.

- Create a spherical Particle Bomb to generate a spherical shock wave radiating in all directions.
- Create a cylindrical Particle Bomb to generate a ring shock wave radiating perpendicular to the local Z axis of the Particle Bomb.
- Create a planar Particle Bomb to generate a directional shock wave traveling along the local Z axis of the Particle Bomb.
- Set Strength to a negative value to create an implosion shock wave.

You can set other parameters of a Particle Bomb to control the following properties:

- Set Start Time and Duration to control when the blast occurs and how long it lasts. Typical blast effects use short Durations of 0.0 to 3.0.
- Choose range options to limit the distance the blast travels.
- Select Chaos to randomly vary the force of the blast for each object. Chaos is only effective if Duration is set to 0.0.

Solving a Dynamic Solution



You solve a dynamic simulation by setting the timing parameters and then clicking Solve. Once calculations are complete you can play the animation to check the result.

Defining Range and Speed

You set Start Time and End Time parameters to limit the simulation to a range of frames.

- Animation keys for objects before the start time are either ignored or applied to the start of the animation based on the Use Initial State property for each object.
- Animation keys for objects within the range of the simulation are replaced.
- Animation keys for objects after the end time of the simulation are not changed.

You can make adjustments to the timing of the simulation by changing the Time Scale parameter. The simulation still covers the same range of frames but the rate at which effects occur is changed.

- Values less than 1.0 reduce speed.
- Values greater than 1.0 increase speed.

Setting Frequency of Calculation

You set Calc Intervals Per Frame to specify how many calculations are performed for each frame of the simulation. The faster objects move, the higher you should set this value.

Increase Calc Intervals Per Frame if objects fail to collide properly and instead pass through each other.

Solving the Simulation

Once all of the object properties, effects, and timing parameters are set, click Solve and then play the animation to check results.

Because the Dynamics utility can create many keys for every object in the simulation you may want to consider the following:

- Save your scene before solving a dynamic simulation.
- On the Edit menu, click Hold before solving a simulation.
- On the Edit menu, click Fetch to restore the scene if a solution is not to your satisfaction.

You can use the Hold and Fetch menu commands to quickly restore your scene while you experiment with dynamic properties.

Reducing Keys After Solving a Simulation

When you solve a dynamics simulation, Position and Rotation keys are generated at every frame of the specified range for every object affected in the simulation. Not only does this result in an excess of keys, but it can increase the size of scene file.

Consider using the Reduce Keys command in Edit Time mode of Track View after you successfully solve a dynamic simulation.

To reduce the keys:

1. On the Dynamics rollout, click Select Objects In Sim.
2. Open a Track View, and set Track View filters to show Animated Tracks Only and Selected Objects Only.
3. Right-click the Objects item in the hierarchy list, and choose Expand All.
4. Click Edit Time, then right-click the hierarchy list, and choose Select All.
5. Double-click in any selected track to select all keys in all tracks.
6. Click Reduce Keys, set a Threshold value, then click OK.

The Position and Rotation keys for all objects in the simulation are reduced to satisfy the Threshold value that you set. Start by using a low threshold when reducing keys. Using a high threshold may allow objects to pass through each other, destroying the effect of the simulation.

See chapter 25, “Basic Track View Use” for more information about Reducing Keys.

Chapter

29

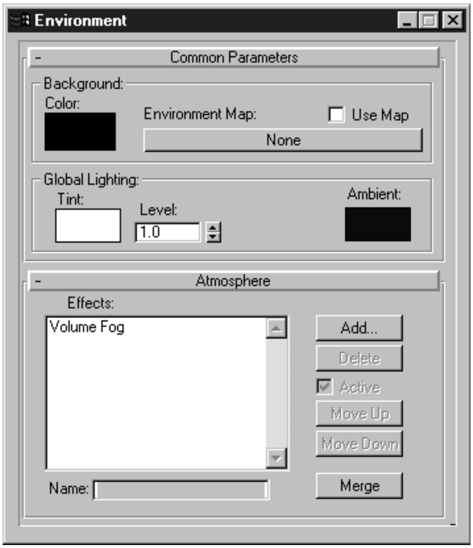
Creating Environments and Atmospheres

Environments are backgrounds used when you render. The background can be a solid color or a map projected in a number of different ways. It can be a 2D animation, as well.

Atmospheres are lighting effects created at rendering time. They can enhance realism or the dramatic effect of a scene. 3D Studio MAX provides these atmospheric effects:

- 2D fog, standard and layered
- 3D volumetric fog
- 3D volumetric light and shadows
- 3D combustion and explosions

Atmospheres are implemented as plug-ins, so more atmospheres than those described in this chapter might be available in your configuration of 3D Studio MAX.



3D Studio MAX provides two kinds of environments:

Background Color—Sets a uniform color that appears behind objects when you render.

Environment Map—Renders a map as the background. Because the map is a standalone map that uses the map tree resources of the Material Editor, this second kind of background is extremely versatile.

Environment maps can be procedural images, bitmaps, 2D animations, or some combination of these.

The Material Editor provides four options, discussed later in this chapter, for mapping background environments:

Spherical—Surrounds the scene completely.

Cylindrical—Surrounds the scene with vertical sides.

Shrink-wrapped—Surrounds the scene completely.

Screen—Appears behind the scene opposite the viewer, like a backdrop.

Note: You can also combine a scene with a background by using Video Post. See chapter 31, “Using Video Post.”

These options affect the finished environment of a scene:

Global Lighting—Sets a uniform tint and intensity for all lights in a scene except ambient light.

Ambient—Sets the color of ambient light. See chapter 17, “Lighting Your Scene.”

Setting Environment Parameters

Background color, environmental mapping, and global and ambient lighting are all set from the Environment dialog.

To set environment parameters:

- Choose Rendering > Environment to display the Environment dialog.

  Environment settings are on the Common Parameters rollout. The Atmospheres rollout, also part of this dialog, is discussed later in this chapter.

To set background color:

1. On the Common Parameters rollout, in the Background area, click the color swatch to display a Color Selector dialog.

2. Use the Color Selector to change the background color.

   The Renderer now uses this color as a background. There is no change in the viewports.

Setting Global Lighting

Ambient light simulates background radiosity in your scene. Tint and Level affect all lights in your scene equally, *except* ambient light.

These options are on the Common Parameters rollout, in the Global Lighting area.

To set global lighting:

- For Ambient and Tint, click the corresponding color swatch to display a Color Selector dialog. Choose a color and click Close. Ambient is black by default, Tint is white.
- For Level, set a value above or below the default of 1.0. Higher numbers raise the lighting level, negative numbers decrease it.

You see the effects of these settings when you render your scene. The settings can be animated.

Assigning and Adjusting an Environment Map

The general procedure is to first assign an environment map, and then adjust it in the Material Editor to the correct environmental coordinates. This topic introduces these general procedures.

Assigning an Environment Map

You have a number of ways to assign an environment map for the background of your scene. Because you can drag maps freely between the different dialogs, the choices are not exclusive of one another.

To use the Environment dialog:

1. On the Common Parameters rollout, in the Background area, click the long Environment Map button. This displays a Materials/Map browser similar to the one in the Materials Editor.
2. Use the controls to locate the map you want to use.
3. Click OK to assign the map.

Using the Material Editor

You typically use the Material Editor to create a *standalone* map not associated with a material. You then assign this map as an environment map. See chapter 21, "Designing Mapped Materials," and online reference for more details.

To use the Material Editor:

1. Load the map you want to use into one of the sample slots of the Material Editor.
2. Drag the map from the slot to the Environment Map button on the Environment dialog.

   A popup dialog asks if you want to make the map an Instance or a Copy. You usually want to choose Instance, the default, so you can make adjustments later.

Using the Asset Manager

The Asset Manager is a useful browsing utility for searching and displaying different file types. You can drag a map directly from the Assets Manager to either the Environment dialog or the Materials Editor.

To use the Asset Manager:

1.  Open the Utilities panel.
2. On the Utilities rollout, click More, select Asset Manager, then click OK.
3. Using the navigation column on the left of this dialog, go to a maps directory. Maps are displayed in easy-to-read thumbnails.
4. Drag a map directly from the Assets Manager to the Environment Map button.
5. When asked to do so, type in a unique name and click OK.

Adjusting an Environment Map

Once you've assigned a map, you typically want to check its mapping coordinates—whether these are set to Spherical, Cylindrical, Shrink-Wrapped, or Screen in the Material Editor.

- If you assigned the map from the Material Editor and made the map an instance, you can adjust the original map in the sample slot to update the environment map.
- If you assigned the map from the Asset Manager or the Environment browser, drag the map from the Environment Map button to a slot in the Material Editor.

  On the Coordinates rollout, the Environ toggle sets automatically.

You have these choices on the Mapping list:

Spherical Environment—Surrounds the scene with a sphere. Bitmaps and some other patterns show a seam where the map's left and right edges meet.

Cylindrical Environment—Surrounds the scene with a cylinder. That is, the world XY plane is surrounded, but the sides of the environment are parallel to the world Z axis. There is no environment looking directly up or down. Bitmaps and some other patterns show a seam where the map's left and right edges meet.

Shrink-wrapped Environment—Surrounds the scene completely. It is like spherical mapping, except that the maps corners are truncated and the edges meet in one singularity—there is no seam even for bitmapped environments.

Screen—Appears behind the scene opposite the viewer, like a backdrop. In other words, screen mapping is a flat plane that is always normal to the viewer.

- Screen is usually the best option for a stationary camera (or other view).
- Circular environments are good when the camera moves through a scene.

Note: You cannot animate the type of environment mapping.

Other Mapping Controls

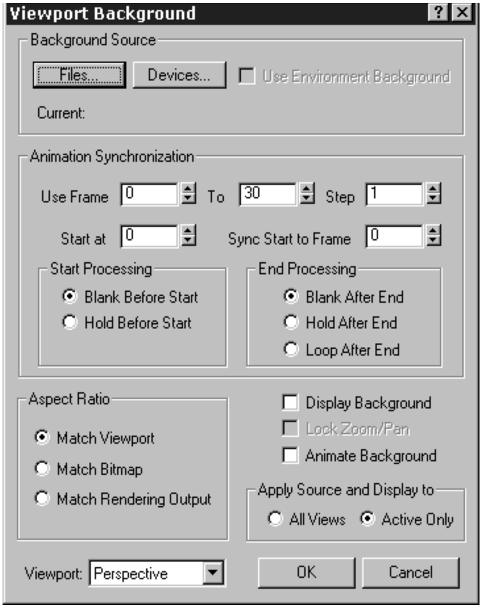
The other mapping controls in the Coordinates rollout—offset, tiling, mirroring, and so on—behave for environment maps as they do for materials. See chapter 21, "Designing Mapped Materials."

If your environment map is a bitmap that has the same resolution as your rendered output, you can save rendering time by turning off bitmap filtering.

To turn off bitmap filtering:

- On the Bitmap Parameters rollout, in the Filtering area, choose None.

Previewing the Environment



Once you've assigned an environment map, you can view the map in any viewport, and match the viewport with the rendered image.

Using Background Image

Background Image is a dialog that loads a bit-mapped or device-generated image into the active viewport. Each viewport can display a different image, and the viewport background can be animated.

Background Image displays an image based on the setting for environmental mapping (Screen, Spherical, and so on) assigned to that image in the Material Editor.

To use a background image:

1. Choose a viewport and make it active.
2. Choose Views > Background Image from the menu bar to display the Viewport Background dialog.

To see the current environment map in the viewport:

1. Set Use Environment Background. Most of the dialog grays out.
2. Set Display Background. This setting is cleared by default—be sure it's set.
3. Click OK to display the environment map.

To see another map in the viewport:

1. Clear Use Environment Background if it is set.
2. Click Files to display a Select Background Image dialog. Select a file.

   The Device options are controlled by plug-in components, and require the plug-in interface, the device driver, and the device itself to be installed on your system.
3. Be sure Display Background is set.
4. Set the aspect ratio, as described below.
5. Set any other parameters and click OK. The image appears in the viewport.

Switch to another viewport and repeat the process with the same or a different image. See online reference for parameter settings.

Matching Viewport and Rendered Backgrounds

This procedure lets you see the exact extent of the background image when it is rendered.

To match a viewport and rendered background:

1. Right-click the viewport label and set Show Safe Frames in the viewport you plan to render.
2. Select the instanced environment map in the Material Editor and set its mapping to Screen.
3. Click Files on the Viewport Background dialog and choose the same map for viewport display.
4. Set the Aspect Ratio options to either Match Viewport or Match Rendering Output. Click OK.

When you render the viewport, the background of the rendered scene should exactly match the background displayed in the Live area of the safe frames.

Aspect Ratio Controls

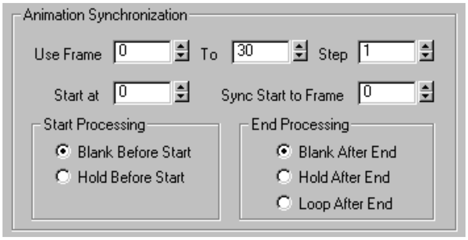
The Viewport Background dialog has three options for how the background fits in the viewport.

Match Viewport—Fits the background to the viewport. This can distort the bitmap.

Match Bitmap—Leaves the bitmap's aspect ratio unchanged. A border might appear in the viewport.

Match Rendering Output—Fits the background to the aspect ratio currently set for rendered frames. This can distort the bitmap, and a border might appear in the viewport.

Using an Animated Environment Map



A bitmapped environment map can be an animation file.

You use the animation bitmap's time controls—on the Time rollout in the Material Editor—to synchronize it with your animated scene.

See online reference for specific procedures.

Note: You can also combine a scene with an animation by using Video Post. See chapter 31, "Using Video Post."

Previewing an Animated Background

There are two reasons to display an animated background in a viewport:

- To preview the effect of an animated background environment.
- To make object movement match the animation or video.

  This technique is called *rotoscoping*. You can rotoscope to simulate motion, without planning to render the background. You can also rotoscope in conjunction with mattes so that scene objects throw shadows on the background. See the next topic, "Creating Mattes with Matte/Shadow Material."

The animation appears in the viewport only one frame at a time.

The 3D Studio MAX Play button does not play the animation unless you set the Update Background While Playing preference for viewports (File > Preferences).

To display an animation in a viewport:

1. Choose Views > Background Image.
2. On the Viewport Background dialog, click Files to choose an animation file.
3. Set Display Background to make sure the background is displayed.
4. Set Animate Background to make sure the background is animated.
5. Set the aspect ratio, as described in the previous topic, and then click OK.

Synchronization Controls

The controls in the Animation Synchronization area let you choose how the animation is played back in the viewport.

Use Frame—Animation frame number where playback begins.

To—Animation frame number where playback ends.

Step—Plays every *nth* frame.

Start At—Scene frame number where the animation begins to play.

Sync Start to Frame—Animation frame at which playback starts in the scene. Defaults to the value of Use Frame.

Setting Sync Start to be greater than Use Frame starts the animation at the later frame (and plays back fewer frames of the sequence).

Blank Before Start—Doesn't show a background before playback begins.

Hold Before Start—Shows the animation's first frame (Sync Start) before playback begins.

Blank After End—Doesn't show a background after playback ends.

Hold After End—Shows the animation's last frame after playback ends.

Loop After End—After animation ends, replays the animation.

Creating Mattes with Matte/Shadow Material

In filmmaking, a *matte* is a compositing tool—it masks part of a scene, usually so the masked area can later be filled with a different background. For example, a singer filmed in a studio is shown singing on a mountain in the finished scene.

Matte/Shadow material performs a similar function. Faces with this material reveal the current environment map. When rendered, the faces are effectively invisible against the background, but they block geometry behind them. They can receive shadows from other objects and be affected by the current atmosphere.

Creating a Matte

You use mattes to make the environment appear to be part of the scene. Not all objects need shadows to be effective. For example, the ground and standing objects toward the front of the background usually receive shadows. The sky, and objects in the distant background, usually do not.

Displaying the background in the viewport can help you shape the object. See “Previewing the Environment.”

To create a matte:

1. Create the object to use as a matte.

If you want the matte to receive shadows, make the object shape similar to the shadow-receiving portion of the background.

2. Use the Material Editor to assign Matte/Shadow material to the object. You can also assign Matte/Shadow material to a sub-object selection.

3. Render the scene to see the finished effect.

In the viewport, an object assigned Matte/Shadow material appears as opaque gray.

Matte/Shadow Parameters

The parameters for Matte/Shadow material control how it interacts with the environment and atmosphere.

Opaque Alpha—When set, the matte is opaque. Clear before you composite the scene with another image, so objects with Matte/Shadow will be invisible. Default=set.

Apply Atmosphere—When set, fogging applies to matte objects. The depth parameters control how fog is applied. This parameter has no effect if you don’t use fog. Default=cleared.

At Background Depth—(Default) The renderer first fogs the scene, and then renders shadows.

Use this option to maintain the illusion that the environment is continuous with the scene.

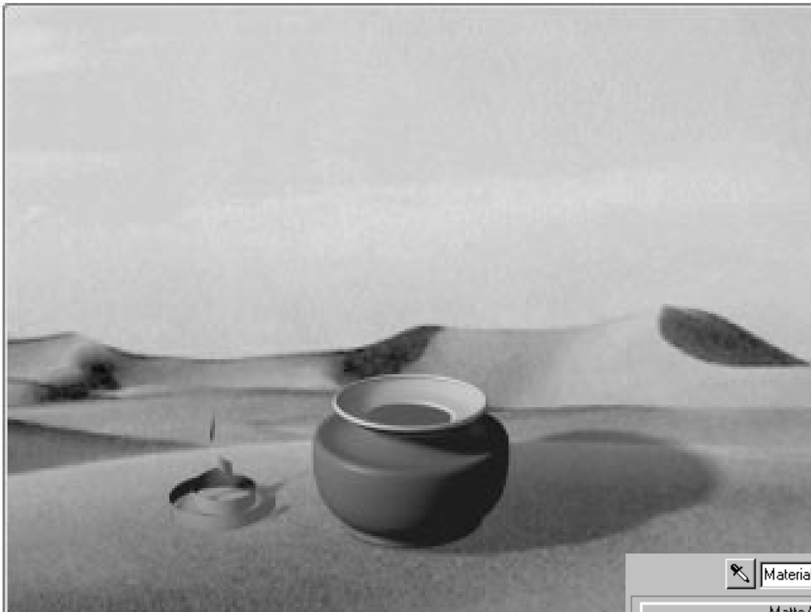
At Object Depth—The renderer first renders shadows, and then fogs the scene. This can create discontinuities between the matte and the background.

Use this setting when you render the scene *without* the background, in preparation for compositing the rendered scene against a separate rendering of the background.

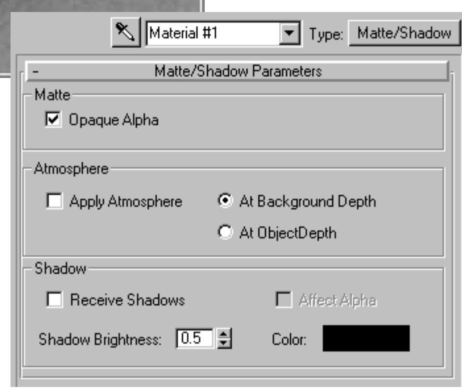
Receive Shadows—When set, shadows are rendered on matte surfaces. Default=cleared.

Shadow Brightness—Bases shadow brightness on the background color.

- At 0.0, full-strength shadows show no color—they are completely black.
- At 1.0, shadows are the same color as the background—in other words, they are invisible.
- At 0.5, shadow brightness is 50% of the background color. Default=0.5.



Scene with matte-created shadows



Creating Atmospheres

Atmospheres are generated when you render a scene. You do not see them in the viewport. 3D Studio MAX provides four kinds of atmospheres:

Fog—Two kinds can be created.

- Standard Fog applies a mist across the entire scene.
- Layered Fog creates fog in a horizontal layer, like ground fog or a cloud cover.

Volume Fog—Creates fog of variable density, which you can animate for wind-blown cloud effects.

Volume Light—Creates visible, 3D light cones and shadows, as if the atmosphere were filled with smoke or dust. As with volume fog, you can animate the density of volume light.

Combustion—Creates fire, explosion, and smoke effects.

Adding an Atmosphere

This is the beginning procedure for all atmospheric effects.

To add an atmosphere:

1. Choose Rendering > Environment to display the Environment dialog.
2. On the Atmosphere rollout, click Add to display the Add Atmospheric Effect dialog.
3. Choose the kind of atmosphere you want to use, and then click OK. Atmospheres are applied in order, from top to bottom.

Top to bottom: Bitmapped background, layered fog added, color added to layered fog

Handling Multiple Atmospheric Effects

You can have more than one atmosphere in effect at the same time. Simply add additional atmospheres. To keep track of the different effects, rename them in the Name field.

The following controls help you manage the Effects list.

- Use Active to turn an atmosphere's effect off or back on.
- If there are more atmospheres than you can see in the window at one time, use the scroll bar to see the rest of the list.
- Use Move Up and Move Down to change an atmosphere's position in the list.
- Click Delete to delete an atmosphere.

Merge Option

Merge allows you to use an atmospheric effect from another 3D Studio MAX file. Clicking Merge on the Atmosphere rollout displays a Merge Atmospheric Effects dialog.

From this dialog, open a file with atmospheric effects. Another dialog lists the available effects in that file.

Select one or more of the effects, and click OK to merge them into the current scene.

Resolving Merge Conflicts

Any lights or gizmos that are part of the effect are merged along with the effect. If one of these incoming objects has the same name as an object in the scene, an alert appears giving you the following choices:

- Rename the incoming object by changing its name in the edit field.
- Merge the incoming object, resulting in two objects with the same name.
- Delete the existing object by clicking Delete Old.

You can also set Apply To All Duplicates. This applies the same resolution to all other matching objects in the two files.

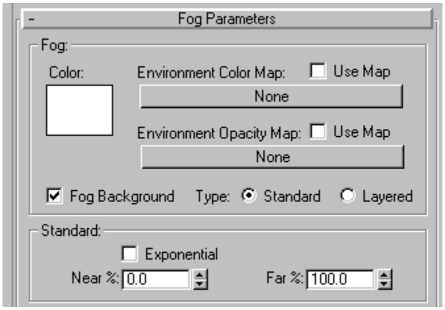
Exponential Option

Volume Fog and Volume Light have a control labeled Exponential.

- When Exponential is set, fog density increases exponentially with distance.
- When Exponential is cleared, density increases linearly with distance.
- Set Exponential only when you want to render transparent objects.

Exponential density is more accurate physically, but it renders more slowly and can cause banding artifacts in the rendering. Linear density renders more quickly and does not cause banding; however, it does not correctly render transparent objects.

Using Standard Fog



Standard Fog is one type of the atmosphere called Fog. The other type, Layered Fog, is discussed in the next topic.

Standard Fog requires a camera in the scene, because it uses the camera's Near and Far environment range values. The effect is best when rendered in camera viewports.

Setting up Standard Fog

Begin by creating a camera view of your scene. Standard fog is based on the camera's environment range values.

To set up standard fog:

1. In the camera's creation parameters, set Show to show the Environment Ranges.

2. Adjust Near Range and Far Range to include the objects you want to fog in your rendering.

As a general guideline, set Far Range just beyond the objects, and Near Range to intersect the object geometry closest to the camera.

3. When you choose Fog in the Add Atmospheric Effect dialog, the parameters for Fog appear. By default, Standard is the selected fog type.

4. Set parameters for Standard Fog and render the result.

Setting Near and Far Percentages

By default, Standard Fog fogs the scene 100% at the Far environment range, and 0% at the Near range.

In other words, beyond the Far range no object geometry is visible and before the Near range all object geometry is visible.

To change these values, adjust Near and Far in the Standard Fog parameters.

Exponential—When set, density increases exponentially with distance. When cleared, density increases linearly with distance. Default=cleared.

Set this only when you want to render transparent objects in volume fog.

Fog Color and Mapping

The color and mapping controls for Fog are similar to those for maps.

Color—Click the Color swatch and then use the Color Selector to change fog color.

Black fog makes the object seem to emerge from a black background—this effect is also known as *distance cueing*.

Environment Color Map—Changes fog from a uniform color to a pattern derived from the map.

Environment Opacity Map—Where the opacity map is opaque, the Renderer fogs the scene; where the map is transparent, no fog appears.

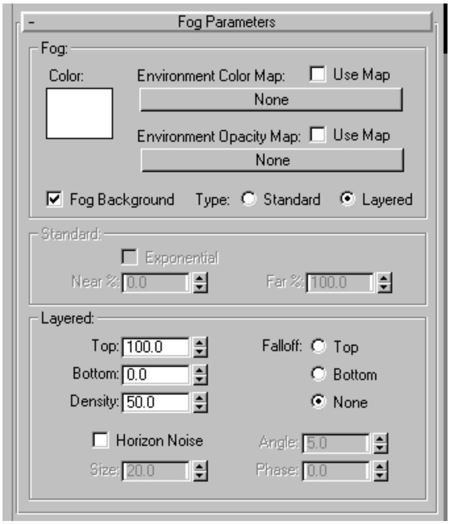
Volume Fog is another way to create fog of irregular density. See "Using Volume Fog."

You assign Fog maps and adjust them in the Material Editor just the way you assign and adjust environment background maps. See "Assigning and Adjusting an Environment Map."

Fog Background—When set, fog is rendered over the background as well as over objects. Default=set.

Clear this parameter only when you are rendering objects *without* a background, in preparation for compositing the rendered objects over another image.

Using Layered Fog



Layered Fog is one type of the atmosphere called Fog. The other type, Standard Fog, is discussed in the previous topic.

Layered Fog requires you to render a camera or perspective view. Orthographic views are not fogged when you render them with Layered Fog turned on.

Setting up Layered Fog

Begin by creating a camera or perspective view of your scene.

To set up layered fog:

1. When you choose Fog in the Add Atmospheric Effect dialog, the parameters for Fog appear.

2. Choose Layered as the fog type.

3. Set parameters for Layered Fog and render the result.

You can have multiple fog layers in the scene by adding multiple Fog entries to the list and making the fog Layered.

Layered Controls

The fog layer is parallel to the ground plane. It can be at any height in the scene, simulating, for example, either a ground fog or a cloud layer. Fog layers are measured in active units from the ground plane.

Top—Top of the fog layer. Default=100.0 units above the ground plane.

Bottom—Bottom of the fog layer. Default=0.0 units.

Density—Layered fog has a uniform density except for its falloff. Default=50 (that is, 50% of the object color).

- You can simulate an uneven density for layered fog by assigning an opacity map.

Falloff—Causes fog density to taper from its Density setting to 0 at either the top or the bottom of the fog layer. Default=None.

Choose Top for the top of the layer to fall off, Bottom for the bottom to fall off, or None to render layered fog with no falloff.

Horizon Noise Controls

Layered Fog has a straight-edged horizon line. The effect is good if this horizon is outside the frame of your scene or is blocked by geometry, but it appears unrealistic if the horizon is visible.

Turn on horizon noise if you are going to render the fog horizon.

Horizon Noise—Set to turn on horizon noise. Default=cleared.

The noise effect is *mirrored* at the horizon line. This can result in strange effects if the height of the fog layer traverses the horizon. In a camera view, the camera's horizon line display can help you frame the scene so the fog layer is either above or below the horizon line.

Size—The scale of noise applied to the fog layer's horizon. Increase this value to increase the size of the fog's "tendrils." Decrease it to decrease tendril size. Default=20.0.

Angle—The angle between the camera and the horizon line at which noise is applied to the fog. Default=5.0.

Phase—Animate this parameter to animate fog noise. Default=0.0.

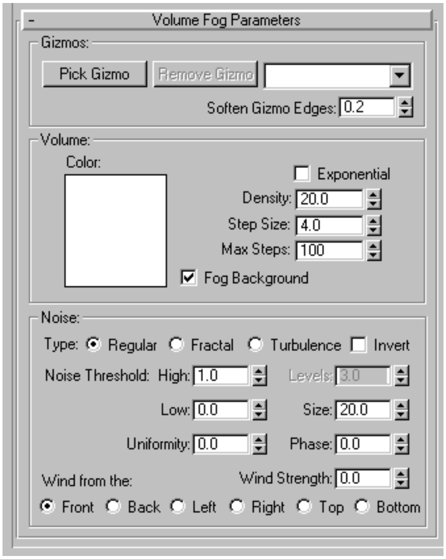
- If you increase Phase over time, then fog tendrils drift upwards and change shape as they do. Use this for a "ground fog" layer.
- If you decrease Phase over time, fog tendrils drift downward. Use this for a fog layer above the horizon.

Tip: The horizon noise is generated in 2D to simulate 3D fog (a "2 1/2 D" effect). This doesn't always work well with a moving camera. If you do move the camera with layered fog, use a small value for the horizon noise Angle, and don't move the camera too much. If camera movement is important to your scene, consider using Volume Fog.

Fog Color and Mapping

These controls are identical to those for Standard Fog, described in the previous topic.

Using Volume Fog



The Fog atmosphere, both standard and layered, is a 2D effect interposed between the scene and the viewer. Volume Fog is an actual 3D effect that creates fog with a density that varies in space and time.

Volume Fog requires you to render a camera or perspective view. Orthographic views are not fogged when you render them with Volume Fog turned on.

Setting Up Volume Fog

By default, Volume Fog fills the entire scene with fog. You can also confine the fog effect by using an atmospheric apparatus.

Using the Default

Begin by creating a camera or perspective view of your scene.

To set up volume fog:

1. Choose Volume Fog in the Add Atmospheric Effect dialog.

2. Set parameters for Volume Fog and render the result.

Note: If there are no objects in your scene, rendering shows only a solid fog color. By default, Volume Fog obscures the background. You can change this by clearing Fog Background.

Using an Atmospheric Apparatus

You can create three types of *atmospheric apparatus* or *gizmo*—a sphere, a box, or a cylinder. These gizmos act to limit the spread of fog in your scene.

You use the same three gizmos with the Combustion effect to shape a fire or explosion. See “Using Combustion.”

As with the default method for Volume Fog, you begin by creating a camera or perspective view of your scene. The apparatus needs to be in the scene and bound to the effect before setting parameters.

To use an atmospheric apparatus:

1. On the Create panel, click Helpers.

2. Open the subcategory list and choose Atmospheric Apparatus. Click one of the three types—BoxGizmo, SphereGizmo, or CylGizmo (cylinder).

3. Drag in a viewport to create the gizmo. The technique is the same as creating the corresponding Standard Primitive.

4. Set gizmo creation parameters as needed. SphereGizmo also has its own hemisphere setting.

Seed values generate different patterns. Clicking New Seed randomly generates a new Seed value.

5. Display the Add Atmospheric Effect dialog and choose Volume Fog.

6. On the Volume Fog Parameters dialog, click Pick Gizmo and select the gizmo in the viewport.

   This binds the gizmo to the Volume Fog effect. The name of the gizmo appears in the list.

7. Set parameters and render the result. The fog is confined to the space occupied by the gizmo.

Gizmo Controls

You can transform a bound gizmo to alter the position, orientation, or scale of Volume Fog. This can be animated, as can the gizmo's creation parameters, causing the fog to grow, move, or shrink.

For Volume Fog, a field for Soften Gizmo Edges has the effect of feathering the outer edges of the fog. The range is 0 to 1.0. In most cases you should use a value greater than 0. You can animate this setting. Default=0.2.

You can bind more than one gizmo to a Volume Fog effect. All use the same setting for Soften Gizmo Edges.

To unbind a gizmo from an effect:

- Choose the gizmo from the list and click Remove. The gizmo remains in the scene.

Volume Fog Controls

These controls change the color and behavior of volume fog.

Color—Click the Color swatch and then use the Color Selector to change volume fog color.

Exponential—When set, density increases exponentially with distance. When cleared, density increases linearly with distance. Default=cleared.

Set the Exponential control only when you want to render transparent objects in volume fog.

Tip: If you set Exponential, increase the Step Size value to avoid banding.

Density—Maximum fog density. Values greater than 20 tend to obscure the scene. Default=20.0.

Step Size—Controls the size of sampling steps used to generate volume fog. Keep this value in the 3 to 5 range unless fog scale is low, in which case Step Size should be in the 1 to 3 range. Default=4.0.

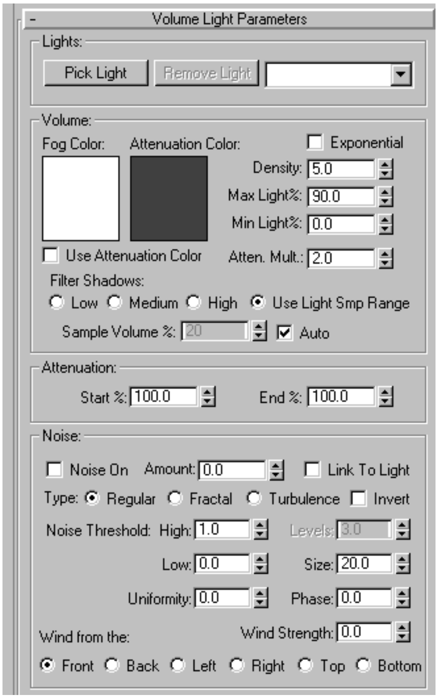
Max Steps—Number of steps taken for fog sampling. The lower the value, the quicker the fog is generated but the greater the danger the fog will render poorly. Default=100.

You can chose a lower Max Steps when using a gizmo. However, if the setting goes too low, aliasing results.

Fog Background—Same as Standard Fog.

See online reference for Noise settings.

Using Volumetric Light and Shadow



Volumetric lights and shadows are created by the Volume Light atmosphere. This behaves as if the scene were filled with a thin fog (or smoke or dust) visible only when light shines through it. Volumetric shadows are also visible when a cone or sphere of light is obstructed by a shadow-casting object.

Volume Light requires you to render a camera or perspective view. Orthographic views are not fogged when you render them with volume light turned on.

Tip: Ray-traced shadows do not appear in volumetric light beams. (They do appear on objects within the beam.) To have the shadows appear within the beam as well as on objects, use shadow-mapped shadows.

Setting Up Volume Light

Begin by creating a scene with lights.

To set up volume lights:

1. Create a camera or perspective view of your scene. Avoid making the view axis parallel to the cone of a spotlight. This tends to create only a washed-out scene, possibly with rendering artifacts.

2. When you choose Volume Light in the Add Atmospheric Effect dialog, the parameters for Volume Light appear.

3. Click Pick Light and then click a light in a viewport to add the light to the list of volume lights. Repeat this step for each light you want to use as a volume light.

4. Set parameters for Volume Light and render to see the result.

Managing Volume Lights

To make a light render with a visible volume, you add it to the list of volume lights.

- To add a light to the list, click Pick Light, and then click the light in a viewport.

- To remove a light from the list, choose the light and click Remove Light.

Tip: Omni lights cast a radial glow. The effect is best if you make the light's attenuation end within the frame of the scene.

Volume Light Controls

These controls change the color and behavior of volume lights.

Tip: To vary these controls on a per-light basis, you can add multiple Volume Light atmospheres to the scene and associate each with a particular light.

Fog Color—Click the Color swatch and then use the Color Selector to change the color of the fog or mist.

This color *combines* with the color of a volume light; it's best to leave it at its default of white and set light color individually, as you would for a scene without volume light.

Attenuation Color—Set like Fog Color. Ramps the color of the fog within the Attenuation Near and Far range of the volume light.

- Use in conjunction with Attenuation Multiplier. Default=2.0.
- Set Use Attenuation Color to turn this effect on.

See online reference for more details.

Exponential—Set this only when you want to render transparent objects in volume light.

Density—Sets the density of the fog. The greater the value, the more light is reflected within a lit volume. Values from 2 to 6 usually work best. Default=5.0.

Max Light %—Maximum glow of a volume light. Default=90.0.

Reduce this value to limit the brightness of a light and prevent it from "whiting out."

Min Light %—Minimum glow of a volume light. This is similar to an ambient light setting—if this value is greater than 0, areas outside light volumes glow with the fog color. If the areas are unbounded by geometry, the fog color fills them as it does for volume fog. Default=0.0.

Set this value greater than 0 only if your scene is completely enclosed by geometry.

Filter Shadows—Sets the level of shadow filtering to improve shadow quality. Choose Medium or High to avoid banding artifacts. The higher the level, the slower the scene renders.

The default is Use Light Smp Range, which blurs the shadows based on the Smp Range value in the light's parameters.

Sample Volume—Number of times a light volume is sampled. Default: Auto=set.

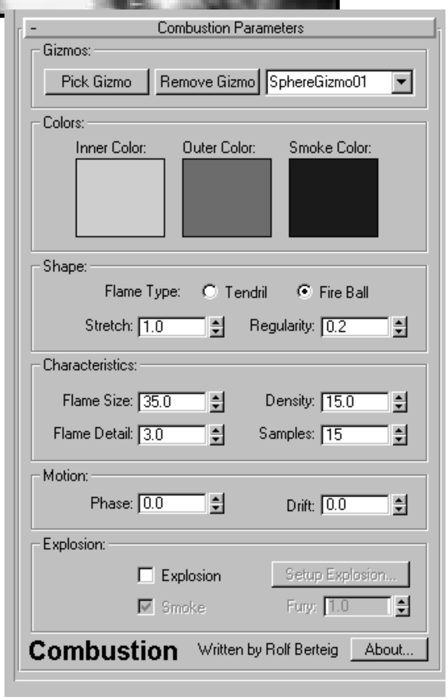
- If Auto is set, this value is set automatically.
- If Auto is cleared, you can set Sample Volume to be from 1 to 100. When using Smp Range, try a setting of 4 to speed rendering.

Attenuation—Attenuation for the volume light. These are percentages of the light's attenuation range. Reducing Start moves the light volume closer to the light object; reducing End extends the length of light casting. Default: Start=100%; End=100%.

- For spotlights, you need to set Use in the Attenuation area in the light's General Parameters.
- For omni lights, Use attenuation can be set or not.

Controls in the Noise area let you animate the fog used by volume light, much as you animate volume fog. See online reference for Noise settings.

Using Combustion



Combustion produces animated fire, smoke, and explosion effects. These effects are shaped by the same gizmos used with Volume Fog, discussed earlier in this chapter.

Combustion effects use no geometry, and therefore tend to render faster than particle systems.

Setting Up Combustion

You can set up Combustion in any viewport. Usually a perspective or camera viewport works best.

You can create three types of *atmospheric apparatus* or *gizmo*—a sphere, a box, or a cylinder. These gizmos control the location of the effect in your scene. The apparatus needs to be in the scene and bound to the effect before setting parameters.

To set up combustion:

1. Open the Create panel and click Helpers.
2. Open the subcategory list and choose Atmospheric Apparatus. Click one of the three types—BoxGizmo, SphereGizmo, or CylGizmo (cylinder).
3. Drag in a viewport to create the gizmo. The technique is the same as creating the corresponding Standard Primitive.
4. Set gizmo creation parameters as needed. SphereGizmo also has its own hemisphere setting.

   Seed values generate different patterns. Clicking New Seed randomly generates a new Seed value.
5. In the Add Atmospheric Effect dialog, choose Combustion.
6. On the Combustion Parameters dialog, click Pick Gizmo and select the gizmo in the viewport.

   This binds the gizmo to the Combustion effect. The name of the gizmo appears in the list.
7. Set parameters and render the result. The fire or explosion is confined to the space occupied by the gizmo.

Setting Combustion Parameters

This is a partial listing of Combustion parameters. See online reference for the details of the complete set.

Color

Two colors define the color of the fire effect, with an optional third color for the smoke that follows the fire in an explosion.

Shape

The two Flame Types are Tendril and Fire Ball (the default).

Tendril—Produces pointed flames with veins along their centers.

Fire Ball—Produces rounded flames, suitable for use with explosions.

These values affect the flame type:

Stretch—Scales the flame effect along the Z axis of the gizmo, with values above 1.0 making the flame longer. Default=1.0.

Regularity—Affects the overall shape of the fire. At 1.0, the shape is close to that of the gizmo. At 0, the overall shape is irregular. Flame Type and Flame Size affect Regularity. Default=0.2.

Characteristics

These parameters set the characteristics of the flame.

Flame Size—Specifies the size of the individual flames or tendrils, not the overall effect. Large-scale scenes require a larger flame size than a small-scale scene.

Flame Detail—Specifies the amount of detail within the individual flames. Larger values provide more detail and increase rendering time.

Density—Specifies the overall strength of the effect. Smaller values make the fire more transparent and less noticeable, while larger values make the fire brighter and more opaque.

Sample—Specifies the rate at which the volume is sampled. Higher values increase precision and generally produce better results while increasing rendering time. In most cases the default works well. Default=15.

Motion

These motion parameters are the animation controls.

Phase—Animates the fire. The faster this value changes during animation, the faster the fire appears to burn.

Drift—Moves the fire along the Z axis of the gizmo, making the flames appear to rise.

Explosion

These parameters set the Combustion effect for an explosion and smoke.

Explosion—Automatically animates Flame Size, Density, and Colors as the Phase parameter changes from 0 to 300. Default=cleared.

Smoke—Automatically animates the transition from flame colors to smoke color as the Phase parameter changes from 100 to 200. Default=set when Explosion is set.

Setup Explosion—Displays a dialog that automates the setup for a typical explosion curve.

Fury—Controls the rate at which the flames "churn" independent of the explosion itself. Increase to 2 or 3 to make the fire burn more furiously. Default=1.0.

Chapter

30

Rendering Scenes and Animations

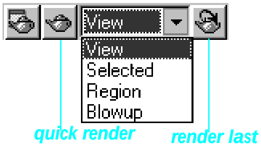
The ultimate purpose of modeling with 3D Studio MAX is to create still images or animations. You do this by *rendering* your scene.

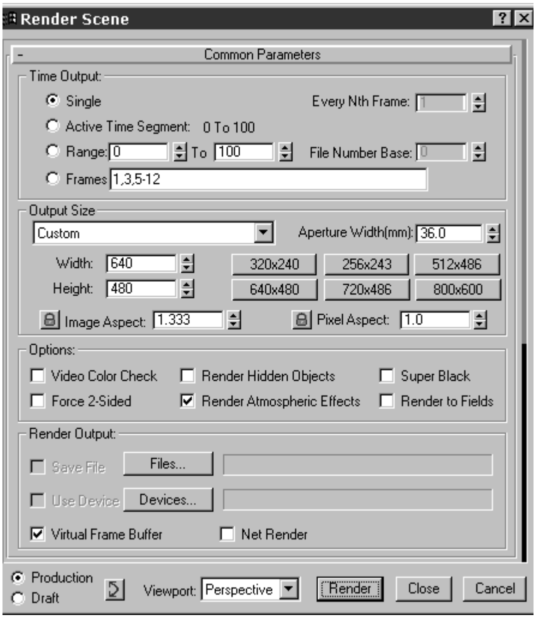
Rendering "colors in" a scene. It shades the scene's geometry using the lighting you've set up, the materials you've applied, and environment settings such as background and atmosphere.

Rendering is multi-threaded and multi-processed on multiple-processor configurations. A two-processor Windows NT system can render in nearly half the time a single-processor system can.

Rendering can also take place on multiple systems by using a network. See the online network rendering documentation.

Using the Rendering Interface





The toolbar gives you the main commands for initiating a rendering in 3D Studio MAX. You set up rendering parameters on the Render Scene dialog. You view the progress of the rendering in a "virtual frame buffer" window.

Using the Toolbar

The rendering buttons and a list are clustered at the right end of the toolbar

 **Render Scene—**This button displays the Render Scene dialog so you can set up the conditions of the current rendering and subsequent renderings as well. Render Scene is equivalent to choosing Rendering > Render from the menu bar.

Render Scene *always* renders the active viewport. If necessary, you can set the active viewport from the dialog just before you render.

Quick Render—These flyout buttons let you choose between Production and Draft rendering, discussed later. Quick Render is the same as Render Scene, except that it doesn't display the Render Scene dialog.

Use Quick Render when you don't need to change any rendering parameters.

 **Render Last—**This button renders the last view you rendered, ignoring which viewport is active. Like Quick Render, it renders the scene immediately, using the previous Render Scene settings. Render Last doesn't save the rendering to disk and doesn't allow network rendering.

Use Render Last when you simply want to view your rendering on the screen.

If you click Render Last when you haven't yet used Render Scene or Quick Render, it renders the active viewport.

Choosing What to Render

You use the render list to choose what the rendering will include. These options work with either Render Scene or Quick Render.

In general, you use View to render completed renderings and the other options to render portions of the view while you are working on a scene.

View—Renders the active viewport.

Selected—Renders only selected objects in the active viewport.

Region—Renders a region, or window, within the active viewport.

Use this option for speed when you need to render only a portion of the scene.

Blowup—The same as Region, except that the window is constrained to the aspect ratio of the output size specified in the Render Scene dialog, and window contents are blown up to fill this size.

To render with Region or Blowup:

1. Choose the Region or Blowup option, then click Quick Render or Render Scene. A dotted window appears in the active viewport.
2. Adjust the window location and size:

- Drag in the middle of the window to move it.
- Drag the window's handles to adjust its size.

3. Click an OK button at the lower right of the viewport to proceed with the rendering.

   Quick Render renders immediately, while Render Scene opens the Render Scene dialog.

Using the Render Scene Dialog

The Render Scene dialog appears when you click Render Scene or choose Rendering > Render. There are two rollouts in the Render Scene dialog:

Common Parameters—Applies to all renderers. This is the one shown above. Unless otherwise noted, parameters in this chapter refer to the Common Parameters rollout.

- Collapse Common Parameters to see the rollout beneath it.

MAX Default Scanline A-Buffer—Default rollout for the renderer provided with 3D Studio MAX. The VUE File Renderer is the other renderer supplied. See online reference for details.

This rollout is specific to the current renderer. Other renderers might be available as plug-in components in your configuration. See "Using Render-Specific Options."

Choosing Production or Draft Rendering

Options at the bottom of Render Scene toggle between Production and Draft rendering. You can also make this switch with the Quick Render flyouts. By default, Production and Draft have exactly the same settings.

As an example of use, you might set up Production parameters for a specific video resolution and output file. You could then switch to Draft and set parameters for quick previews at a smaller resolution.

If you have a second renderer configured in your system, you can assign it to Production, Draft, or both.

To assign a renderer:

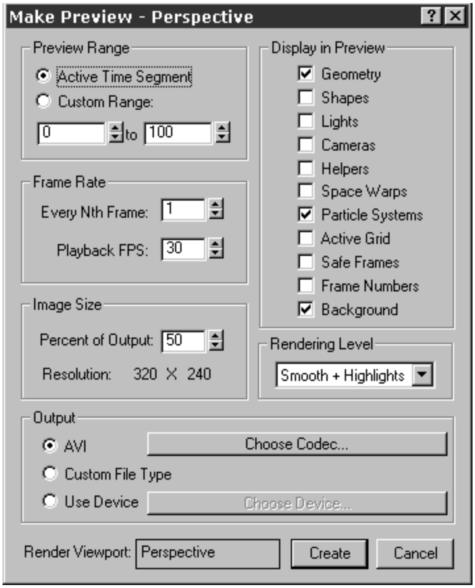
1. Choose File > Preferences and open the Rendering panel of the Preference Settings dialog.
2. In the Current Renderer area, click Assign for Production or Draft and select another renderer.

Viewing the Rendering

By default, when 3DS MAX begins to render, a Virtual Frame Buffer (VFB) window shows the rendering, and a dialog shows the progress of rendering and the rendering parameter settings. See "Using the VFB Render Window."

- Before rendering, you can turn off the VFB window by clearing its option in the Render Output area.
- To stop rendering at any time, click Cancel in this dialog, or press ESC.

Creating a Preview Animation



Before final rendering, you can create a special animation to preview the result. Preview animations have a number of distinct features:

- Non-renderable icons for lights, cameras, space warps, and other kinds of non-geometric objects, all render.
- Environment background and atmospheric effects do not render.
- Only one material map renders per object, as is true for viewports.

To create and view a preview:

1. Choose Rendering > Make Preview to display the Make Preview dialog. The active view is listed in the title bar.
2. Change the preview parameters or accept the defaults, and then click OK.

   3D Studio MAX renders the preview and saves it in a file called *_scene.avi*.

   Immediately after rendering the preview, 3DS MAX invokes the Media Player for this animation.
3.  View the preview by clicking Play in the Media Player.

To view the preview at a later time:

- Choose Rendering > View Preview. This invokes the Media Player with *_scene.avi* loaded again.

You can rename the *_scene.avi* preview file so it won't be overwritten the next time you make a preview.

To save the preview under a different name:

- Choose Rendering > Rename Preview, and give the preview another name in the File Save dialog that appears.

Tip: If you clear AutoPlay Preview File in the General panel of the Preference Settings dialog (File > Preferences), the Media Player is not automatically called every time you make a preview.

Preview Options

The Make Preview dialog gives you a choice of how to render the preview animation.

Preview Range—Choose either the active time segment or specify a range by entering From and To frame numbers. Default=Active Time Segment.

Every Nth Frame—Skips frames as the Render Scene dialog does. Default=1.

Playback FPS —Specifies at how many frames per second the preview will play back. Default=30.

Image Size—Changes Percent of Output to change the size of preview frames. The Resolution label updates to show your changes. Default=50% of the size set in the Render Scene dialog.

Display in Preview—Selects which kinds of objects or viewport aids you want to display in the preview. Objects that don't render, such as lights, appear in the preview as they do in viewports. Defaults=Geometry, Particle Systems, Backgrounds.

Rendering Level—Lets you choose the rendering level of the preview. The lower the level, the quicker frames will render but the less detail will appear. Default=Smooth+Highlights.

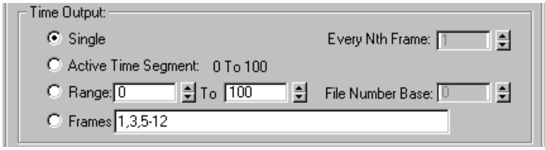
AVI—(Default) Renders to *_scene.avi*, as previously described. Activates Choose Codec, which displays another dialog where you specify the type of video compression you want to use.

Note: You can set the frequency of AVI video-compression keyframes for AVI preview files in Rendering preferences (File > Preferences).

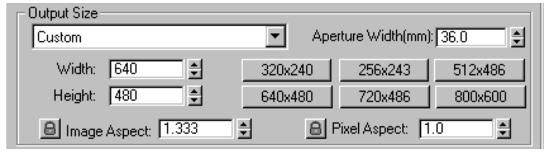
Custom File Type—Displays a Make Preview dialog as you create the preview. Choose from available file types.

Use Device—Enables use of an output device, if one is configured for your system. See "Rendering to a Device."

Setting Output for Still and Animated Images



Time Output area of Render Scene dialog



Output Size area

Rendering depends on setting the proper parameters to achieve the output you want. This topic covers choices for creating still and animated images, and the size and format of those images. Following topics discuss the destination of this output in more detail.

Setting Frame Ranges

Rendering a still image is the default setting for the Render Scene command. However, a still image is only a special case of animation—a single frame. The process is essentially the same for both types of output.

To set render output:

1. Begin by making the viewport you want to render active.

2.  Click Render Scene to display the Render Scene dialog. Options for setting the number of frames are in the Time Output area.

- For a still image, make sure Single is selected. For an animation, set a number of frames using the other parameters, defined below.
- For either stills or animation, set the output image size as described in the next heading.
- For animation, you also need to set a render output, typically to a file. You can do this for still images as well. See following topics.

3. Click Render. Once parameters are set, you can render a view—without using the dialog—by clicking Quick Render. Do not use Render Last if you are saving to file.

Warning: If you set a time range but don't specify a file to save to, the animation is rendered only to the window. If you intended to save the animation or view it at full playback speed, this can be a time-consuming mistake. See "Rendering to a File."

Time Output Parameters

Single—(Default) Renders only the current frame.

Active Time Segment—Renders all of the active time segment. This range is shown to the right of the label.

The active time segment is the range of the time slider, which you set by using the Time Configuration dialog. See chapter 22, "Animation Concepts and Methods."

Range—Renders a range you specify, which can be other than the active time segment. You choose the range by setting frame numbers in the (unlabeled) From and To fields.

Frames—Renders selected frames, which you specify by entering a list of frame numbers in the edit field.

Separate individual frames or sub-ranges by a comma. For example, **1,3,5**. To specify a sub-range, use a hyphen. For example, **1–5**.

The Active Time Segment and Range options, as well as sub-ranges in the Frames option, specify starting and ending frames. By default, the Renderer renders all frames in the range, but you can choose to have it skip frames in a range using Every Nth Frame.

Every Nth Frame—Set to a value greater than 1 to skip frames in the range. For example, 3 renders every third frame; 10 renders every tenth frame. Default=1.

Skipping frames creates a faster animation that you can use as a preview or "profile" of the finished animation.

Use Every Nth Frame if you want to see rendered detail such as materials in the preview.

File Number Base—Used with Every Nth Frame to specify the starting point of incremental file names. For example, if Every Nth Frame is set to 10 over a range of 30, and File Number Base is 23, the rendered files (if named *file*) would be numbered *file0023*, *file0033*, *file0043*, *file0053*.

Setting Image Format and Resolution

Options for setting the size and format of the output image are in the Output Size area.

Choosing a Format

You choose a format from the list. There are over a dozen formats available, from 35mm formats to HDTV (video).

The default format is Custom, which allows you to change any of the settings at will.

When you choose any of the standard formats, the aperture width and the image and pixel aspect ratios are locked to that format. However, you can change the resolution.

See online reference for further details.

Setting Resolution

You can set pixel resolution manually using the Width and Height fields. For all formats but Custom, changing one field changes the other to maintain the image aspect of that format.

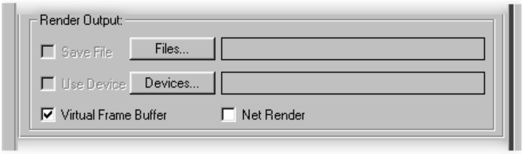
You can also choose resolution by clicking any of the six preset buttons. The resolutions listed on these buttons vary with format. The Preset buttons can be customized.

To customize preset buttons:

- Right-click any button to display a Configure Preset dialog where you can vary the Width and Height for that button. For standard formats, image aspect is maintained.

Tip: Smaller images render much more quickly. Use 320 × 240, for example, to render draft images, then change to a larger size for your final work.

Choosing the Destination of Rendered Output



By default, 3D Studio MAX renders to a window. In many cases, you want to render a still image or an animation to a file as well.

Two additional options are to render to a graphics device such as a digital recorder, and to use a network when rendering.

Controls are on the Render Scene dialog in the Render Output area.

Save File—Select to output a rendering to file. This setting remains grayed out until you enter a valid file name.

To set file output:

- Click Files to display a standard file dialog where you choose the output file type and give the file a name and location on your system. See the next topic, “Rendering to a File.”

After you have specified a file, clearing Save File lets you render only to the screen or a device, without “losing” the file name, which is saved with the scene.

Use Device—Select to output a rendering to an output device. This setting remains grayed out until you choose a device attached to your system.

To set device output:

- Click Devices to display a dialog where you choose a device and its setup options. See “Rendering to a Device” later in this chapter for more details.

After you have specified a device, clearing Use Device lets you render only to the screen or a file, without “losing” the device name or settings, which are saved with the scene.

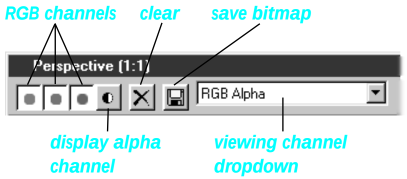
Virtual Frame Buffer—Select to render to a window, regardless of whether it is also rendering to a file or a device. Default=selected.

See the next topic, “Using the VFB Render Window,” for a description of window controls.

Net Render—Select to output a rendering to a network. When selected, Virtual Frame Buffer is automatically disabled.

See online reference for details on how to set up and administer a network rendering system.

Using the VFB Render Window



The window that displays renderings on the screen—the render window—is also called the Virtual Frame Buffer (VFB). *Virtual frame buffer* refers to the memory used to store the rendering that is displayed.

You also access a version of the VFB when you view files. See "Using the View File Window" following.

Render Window Controls

The toolbar of the render window lets you control rendering display, save the rendering to a file, or make a copy of the VFB itself.

Title bar—Displays the name of the viewport, the frame number, and the zoom ratio. See details on zooming below.

Enable R, G, B—Three buttons let you view RGB channels together, individually, or in combinations of two colors.

Display Alpha Channel—Turns off RGB display and displays the alpha channel.

Monochrome—Displays the image in gray scale.

Clear—Clears the render window.

Save Bitmap—Saves the rendering. A file dialog appears, where you can choose the bitmap file type and enter a name for the file.

Clone Virtual Frame Buffer—Creates a copy of the current render window. The copy is independent of the original and can be saved and closed. The original window is used for subsequent renderings. You can continue to clone the original to make comparison copies.

Viewing Channel list—Shows only "RGB Alpha" unless you rendered to a format with additional graphics buffer (*G-Buffer)* channels. See online reference for details.

Pixel Color—Displays the current pixel color as you right-click and drag an eyedropper across the image. The color is saved in this swatch. You can copy or swap this color to another color swatch by dragging. Right-clicking the swatch displays a color selector, which can remain open while you select a pixel color.

Zooming the VFB

You can zoom and pan the image in the VFB window, even while rendering is in progress. The zoom level updates in the title bar.

To zoom and pan the VFB window:

- To zoom in, press CTRL and click.
- To zoom out, press CTRL and right-click.
- To pan, press SHIFT and drag.

Using the View File Window

When you use File > View File to display a saved image, a variant of the render window appears without Save or Clone controls.

When you view a sequentially numbered file or an *.ifl* file in this window, frame-step arrows appear at the right end of the toolbar.

To view sequential files:

- Click arrows to move forward or backward through the sequence of files.
- Press CTRL and click arrows to move to the first or last image.

Rendering to a File

When you render to a file, you save either a single-frame still image or an animation.

Saving Still Images

You can save still images to a variety of image file formats, such as *.bmp, .jpg, .tga*.

In addition to the usual red, green, blue, and alpha channels. the *.rla* still-image format can save a number of channels with information about the scene.

Saving Animations

You can save animations to the *.avi* or *.flc* file formats. You can play back *.avi* files using the Files > View File command, which invokes the Media Player.

Depending on which file format you choose, the file dialog's Setup button might display a dialog that presents options for that format.

Image file formats are supported by plug-ins. Check the online reference or documentation from your plug-in developer for details about supported file formats.

Saving Animations as Still Images

You can also save animations as a sequence of still image files.

To render a sequence of stills:

1. Specify a time range for rendering.

2. Specify a still-image file format.

3D Studio MAX renders to a sequence of still-image files, appending a sequence number to the name of each file.

Note: In the Rendering preferences (File > Preferences), you can change the sequence number convention with the Nth Serial Numbering control.

Setting Output Gamma Correction

For some file formats and devices, the gamma correction required differs from the output gamma correction set globally for 3D Studio MAX. You can set a special output value when you specify the output file type or device type.

To set gamma correction for rendered output:

1. In the file or device dialog, click Gamma to display an Output Gamma Settings dialog.

2. Click Override to override the global output gamma setting.

3. Set the gamma correction value, and then click OK.

If you leave the gamma option set to Use System Default Gamma, 3DS MAX uses the global setting.

Setting Global Gamma Correction

You set overall gamma correction for 3DS MAX in the Gamma panel of the Preferences Settings dialog (File > Preferences). The Output Gamma field under Bitmap Files is the global gamma correction for rendering to files and devices.

For guidelines on how to use gamma correction, see the online reference for this dialog, and in particular, the online definition of the term *gamma*.

Rendering to a Device

To use device output, the device and its driver must be installed on your system.

Devices are supported by plug-ins. Check online reference or documentation from your plug-in developer for details about supported output devices.

To set up a device for rendering:

1. Click Devices in the Render Scene dialog or the Make Preview dialog to display the Select Image Output Device dialog.

2. Choose the device you want to use from the list of devices.

3. Click Setup. Depending on which device you choose, the device dialog's Setup button might display a dialog that presents options for that device.

Adjusting Aspect Ratio

For video monitors and file formats, pixels are square—their aspect ratio is 1:1, also expressed as 1.0—but some devices use non-square pixels.

The Render Scene dialog's Pixel Aspect Ratio control is provided primarily for use when rendering to a device.

To adjust aspect ratio:

- On the Render Scene dialog, in the Output Size area, adjust the Pixel Aspect Ratio to fit the requirements of your output device.

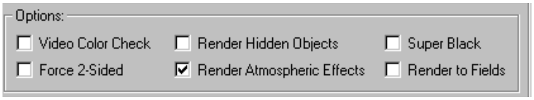
  The Image Aspect Ratio field updates to show what the aspect of the rendered image will be.

If you alter pixel aspect ratio but also render to a window or a file, the rendered image appears distorted.

Gamma Correction

The gamma correction for devices is the same as for files. See the previous topic, "Rendering to a File."

Using Render Control Options



The Options area of the Common Parameters rollout in the Render Scene dialog contains several miscellaneous options.

Video Color Check—Renders “illegal” pixels—pixels whose color is beyond the safe NTSC or PAL video threshold—as black. Computers can output color values beyond the capabilities of video systems. These so-called “hot” colors (pure reds, for example) appear blurry or fuzzy in video. Default=cleared.

Use this option to check renderings before you transfer them to video.

If Video Color Check detects illegal colors, you can either return to the Material Editor and reduce the saturation of the material colors in question, or adjust the lighting on the illegal material to reduce its rendered intensity.

In the Rendering preferences (File > Preferences), you can choose between NTSC and PAL checking, and also choose alternate ways to display illegal pixels.

Force 2-Sided—Renders all objects as two-sided, regardless of the material assigned to them.

Use this option to fix a rendering that appears with missing faces. Because twice as many faces must be rendered, this option significantly increases rendering time. Default=cleared.

Render Hidden Objects—Renders hidden objects. Hidden objects aren’t rendered when this option is cleared. Default=cleared.

Render Atmospheric Effects—Renders atmospheric effects, such as fog and volume light, that you specify in the Environment dialog. (The background you specify in the same dialog is rendered in either case.) Clear this option to speed up rendering time or to see objects the atmospheric effects might obscure. Default=selected.

Super Black—Limits the darkness of rendered geometry. When selected, dark shadows are rendered with an intensity slightly higher than the “super black” background. Default=cleared.

Use this option to prevent shadows from becoming invisible when you use luminance keying to composite renderings transferred to video.

In the Rendering preferences (File > Preferences), you can choose the threshold between super black and shadows.

Render to Fields—Renders to video fields rather than frames. Default=cleared.

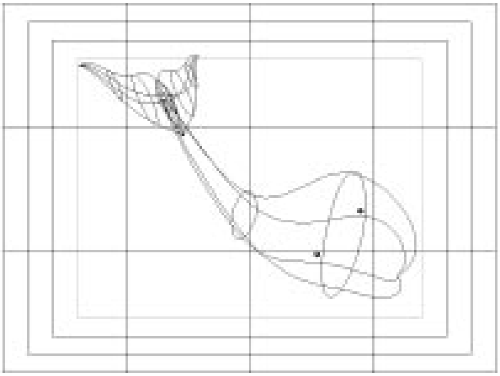
Use this option when you render animations to videotape.

Each frame of video consists of two interlaced *fields* of even and odd scan lines. When this option is selected, each frame of output combines the odd scanlines from one frame with the even scanlines from the next.

In the Render window, field rendering appears blurry because the lines of the alternate fields are offset, as video requires.

In the Rendering preferences (File > Preferences), you can choose whether to render odd or even fields first. Different video systems require different field ordering.

Using Safe Frame Display



Safe frames and 12-field grid

The Safe Frame display in a viewport helps you "frame" your rendering, depending on its output destination and how you want to use this. The display is based on the Output Size setting on the Render Scene dialog.

To turn on Safe Frame display:

- Right-click a viewport label and then choose Show Safe Frame from the viewport properties popup menu. As an alternative, use the default keyboard shortcut, SHIFT+F.

Using Safe Frame display, you can show any of four kinds of frames in the active viewport:

Live Area—The rendered area of the full scene, including its background. The aspect ratio of the Live Area is set by the Render Scene output size. The longer dimension of the output size fits just inside the viewport.

Action Safe—The area in which it is safe to show subject matter, including action, that you want the viewer to see.

The action safe area is smaller than the live area because image reproduction often clips the edges of a frame—broadcasting usually clips the edges in order to avoid a black border on the television screen; slide mounts sometimes cover part of the frame; and printed images are often clipped to fit them into page layout.

Title Safe—The area in which it is safe to show titles. This area is smaller than the action safe area, mainly because of the distortion that can occur in broadcast video.

User Safe—This area can be customized. On the Safe Frame panel, clear the lock and enter percentage reductions.

Safe Frame Settings

To display Safe Frame settings, choose Views > Viewport Configuration and open the Safe Frames panel. These settings are saved with each scene file.

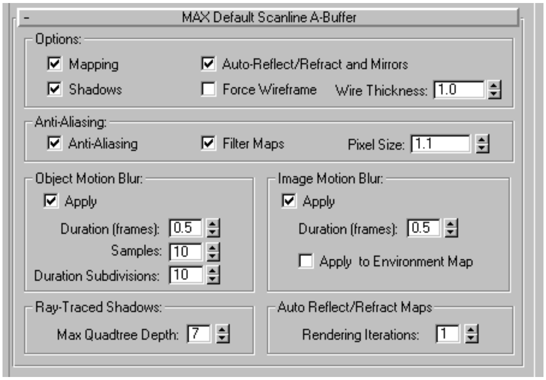
Setup is the main area of the panel where you can select or clear the display of any of the four frame types. See online reference for more details.

Percentage Reductions—Values are expressed as a percentage of the size of the full frame or Live Area. The Title Safe area is constrained to always be smaller than the Action Safe area.

- Select Lock to maintain the ratio for horizontal and vertical percentages. Allows both settings to be changed at once.
- Clear Lock to remove the ratio constraint. Allows horizontal and vertical settings to be changed separately.

12-Field Grid—Displays a grid in the viewport as an aid in describing specific areas within an image. The term "field" refers to the cells of the grid, not to scanline fields. A 4x3 field is the default, with an option for 12x9. Default=cleared.

Using Render-Specific Options



The MAX Default Scanline A-Buffer rollout in the Render Scene dialog contains controls that are specific to this renderer, which is the renderer shipped with 3D Studio MAX.

Settings for the Default Renderer

The settings described here are for the default renderer. Other renderers might be available as plug-in components in your configuration. You choose which renderer to use in the Rendering panel of the Preference Settings dialog (File > Preferences).

With a different renderer active, you see a different renderer-specific rollout in the Render Scene dialog.

Option Controls

The controls in the Options area are mainly for speeding up rendering when you are working with a draft of a scene or animation.

Mapping—Clear to turn off material mapping. Default=selected.

Shadows—Clear to turn off shadow generation. Default=selected.

Auto-Reflect/Refract and Mirrors—Clear to turn off reflection generation. Default=selected.

Force Wireframe—Select to force all materials to display as wireframes. Default=cleared.

Note: Anti-Aliasing must be selected for forced wireframes to render. With anti-aliasing cleared, Force Wireframe has no effect.

Wire Thickness—Selects the wire thickness for forced wireframe. Has no effect on wireframe materials when Force Wireframe is cleared. Default=1.0.

Increase this value if objects are hard to see in forced wireframe renderings.

Anti-Aliasing Controls

The controls in the Anti-Aliasing area are for anti-aliasing and map filtering.

Anti-Aliasing—Clear to turn off anti-aliasing. Use this to save time when rendering draft images. Default=selected.

Filter Maps—Clear to turn off map filtering. Use this to save memory when rendering draft images of materials with summed-area filtering, or when you need extremely sharp textures in distant parts of your scene. Default=selected.

See chapter 21, “Designing Mapped Materials,” for more about map filtering.

Pixel Size—Selects the pixel size used in averaging for anti-aliasing purposes. Range=1.0–1.5. Default=1.1.

Increase this value to improve anti-aliasing for nearly vertical and nearly horizontal edges, especially with high-contrast lighting.

In the Rendering preferences (File > Preferences), you can increase the Pixel Size limit up to 2.0. For high-resolution film and video output, increasing Pixel Size above 1.5 can improve smoothing at a cost of greater rendering time.

Pixel Sizes in the 1.5–2.0 range can occasionally cause artifacts such as black pixels along the edge.

Motion Blur Controls

The Motion Blur area is for blurring motion on a per-object basis. See the next topic, “Adding Object and Image Motion Blur.”

Ray-Traced Shadows Control

The Ray-Traced Shadows area contains a global control for ray-traced shadows.

Max Quadtree Depth—Limits the depth of the tree used when generating ray-traced shadows. Reduce this value to save time and memory while rendering. Range=0–10. Default=7.

Increase this value if ray-traced shadows appear to be inaccurate.

Auto Reflect/Refract Maps Control

The Auto Reflect/Refract Maps area contains a control that affects reflective objects reflecting other reflective objects. In real life, this creates a virtually infinite number of interreflections. In 3D Studio MAX, you set the number of interreflections with the Rendering Iterations parameter.

Rendering Iterations—Selects the number of interreflections when automatically reflective materials (Reflect/Refract or Flat Mirror) reflect each other. Range=1–10. Default=1.

Increase the value to increase the number of interreflections. This increases rendering time, as well.

Adding Object and Image Motion Blur

Motion blur settings make rendered movement appear to move more smoothly. There are two choices for motion blur at the object level:

Object Motion Blur—Applied *during* the scanline rendering process. This is the object motion blur used in 3D Studio MAX Release 1.

Image Motion Blur—Applied *after* the frame is rendered.

Object and Image motion blur differ from scene motion blur, which you can create by using Video Post. Scene motion blur applies blur at the scene level. It takes camera movement into account, which object motion blur does not.

You can use all three kinds of blur in the finished animation. These are general guidelines:

- Use Object motion blur to smooth the animated rendering of rapidly moving objects.
- Use Image motion blur to blur both objects and the environment background. When blurring the environment, you must use Spherical, Cylindrical, or Shrink-wrapped mapping coordinates. Screen does not work.
- Use scene motion blur if you want blurred movement or "trails" to be noticeable as a special effect in your animation.

How Motion Blur Works

For Object motion blur, the renderer generates motion blur by simulating a camera with an open shutter. It interpolates and then renders movement *within* a frame, generating a series of images of the moving object, instead of the default single image.

For Image motion blur, the renderer creates a smearing effect for each object. This produces a smoother blur than the time-slice blur effect produced by either Object or Scene motion blur. Generally, Image motion blur provides a better effect.

Setting Up and Rendering Motion Blur

The procedures for setting up and rendering Object and Image motion blur are generally the same.

To set up motion blur:

1. Begin by selecting the object to blur.
2. Right-click the object, and then choose Properties from the object properties popup menu.
3. In the Object Properties dialog, set Motion Blur to Object or Image, and then click OK.

- Image has an additional setting for Multiplier to increase or decrease the length of the blur "streak."

4. Repeat these steps for each object you want to blur in your scene.

To render motion blur:

1.  Click Render Scene to display the Render Scene dialog and open the MAX Default Scanline A-Buffer rollout.
2. Apply either Object Motion Blur or Image Motion Blur. Apply both if you have objects set to both types of blur.
3. Set Duration, Samples, and Duration Subdivisions. See "Motion Blur Parameters," following.
4. Set other rendering parameters, and then click Render.

Motion Blur Parameters

Duration (Frames)—For both Object and Image motion blur, sets the number of frames during which the simulated “camera shutter” is open. Range=0.01–99.0. Default=0.5.

Increase this value to exaggerate the motion blur effect. Decrease it to make the blur more subtle.

Apply To Environment Map—For Image motion blur only. Default=cleared.

Samples—For Object motion blur, sets the number of sample slices used in the final rendering. Range=1–16 (must be less than or equal to the value of Duration Subdivisions). Default=10.

The higher this number, the smoother the effect, and the longer the rendering time. Low settings cause a “sketchy” or dithered blur.

Duration Subdivisions—For Object motion blur, sets the number of sub-frame “slices” that are interpolated and rendered within the Duration period before sampling. Range=1–16. Default=10.

If Samples is less than Duration Subdivisions, the slices used are selected randomly, giving a grainy look to the blur.

If Samples equals Duration Subdivisions, the blur is smooth. The smoothest blur results from larger, equal values of these two parameters—but be aware that this can slow down rendering by a factor of three to four.

Chapter

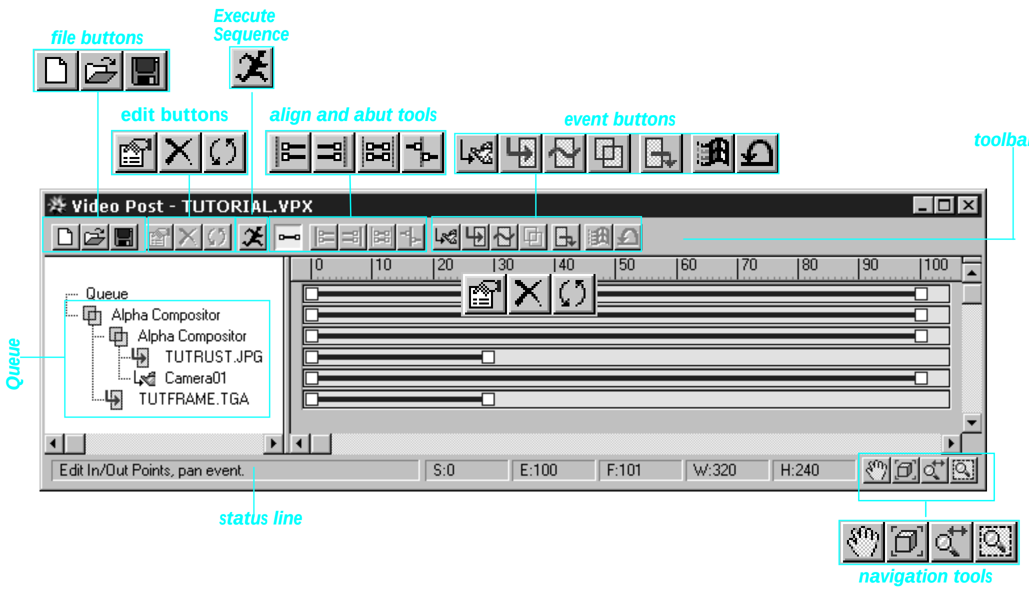
31

Using Video Post

Film and video creation includes a *production* phase, when footage is shot and animations are drawn or generated, and a *post-production* phase. Typical post-production activities are editing to combine raw footage into sequences, adding special effects including layered composite shots, and printing titles for the film or video.

The 3D Studio MAX Video Post dialog helps you accomplish these post-production activities. You can combine your animated scene with other images, which can be animations rendered by different 3DS MAX scenes, still images, captured video, and so on. You can organize the sequence of the finished video, and create transitions between scenes and sequences.

Using the Video Post Interface



Video Post is a self-contained, modeless dialog. Its appearance is similar to the Track View dialog.

Video Post manages the elements of the finished video by means of a *queue*. Items in the queue are called *events*. Events in the queue can be sequential, such as one scene that cuts to another, or they can be simultaneous, such as a composite that combines a still background, a 3D animation, and a title in the foreground.

Video Post data is saved with a scene. You can also save it separately to an independent file.

To use Video Post:

1. Choose Rendering > Video Post to display the Video Post dialog.

2. Create a new Video Post sequence by adding events to the queue, or open an existing Video Post file in order to edit it.

The Video Post dialog has these main elements:

Toolbar—Provides Video Post commands.

Queue—Shows the sequence of post-production events.

Edit window—Controls when each event occurs in the finished video.

Each event is associated with a track that has a range bar, similar to the animation ranges in Track View.

Navigation tools—Let you zoom or pan in the edit window.

Status line—Shows information about the active Video Post controls.

- The first field is a prompt for the active command mode.
- The S: and E: fields show the start and end frame numbers of a range.

  If no ranges are selected, this is the range of the entire output video. If one range bar is selected, this is the range of the selected event. If more than one range bar is selected, then S: shows the earliest frame of all the bars, and E: shows the last frame of all the bars.

- The F: field shows the total number of frames in the output video.
- The W: and H: fields show the width and height of a frame in the output video.

Video Post Toolbar

The Video Post toolbar is organized into groups of related buttons.

- The file buttons are for creating a new Video Post file, opening an existing file, or saving the current data.
- The edit buttons are for editing the queue.
- The Execute Sequence button creates a new animation by executing the events in the queue.
- The Edit Range Bar button lets you edit range bars in the edit window by dragging a range bar to move it or dragging its endpoints to change its length.
- The align and abut tools adjust range bars to control the sequence of the final video.
- The event buttons add input and output events to the queue.

Navigating the Edit Window

The navigation buttons in the lower-right corner of the Video Post window help you move around in video sequences, especially long ones.

Zoom Region—Drags a region in the edit window to scale that region to fill the window. Zoom Region remains active until you right-click to cancel or click another button.

Pan—Drags your view of the edit window. Drag horizontally to slide the view forward and backward in time. To move vertically in the edit window, use the scroll bar at the right. Pan remains active until you right-click to cancel or click another button.

Zoom Time—Drags horizontally to zoom your view of time. Drag left to zoom out and see more time. Drag right to zoom in and see less time. Zoom Time remains active until you right-click to cancel or click another button.

Zoom Extents—Horizontally scales your view to display the full length of all range bars. Click Zoom Extents to zoom. Unlike the other Video Post navigation buttons, Zoom Extents is not modal.

Managing Video Post Files

All Video Post settings are saved with your *.max* scene file. You can also save and load Video Post data from a file independently of your scene. Video Post files are saved with a *.vpx* filename extension.

- Click Save Sequence to save the active Video Post data. A file selection dialog appears to let you enter a name for the new Video Post file.

  By default, Save Sequence saves to the Vpost subdirectory in your 3DS MAX directory. You can change the default path to another directory by using File > Configure Paths.

-  Click Open Sequence to open an existing Video Post file. Use the file selection dialog that appears to choose the video post (*.vpx)* file you want to edit.

  You can also use Open Sequence to open the queue from a 3DS MAX scene (*.max* file). Video Post loads only the Video Post data from the *.max* file, leaving the current scene unchanged.

-  Click New Sequence to create a new Video Post file. This erases all the current Video Post data.

Managing the Queue

In the Video Post dialog, you always have at least one item in the queue, and that is a placeholder labeled Queue. This placeholder is always the first item in the queue; it is the queue's parent event. The queue is illustrated in the previous topic.

The queue can be linear, but some kinds of events, such as Image Layer, combine other events and become the parent event to those events. In other words, the queue can also be a hierarchy. The level immediately below Queue is the queue's top level.

To add an event to the queue:

- Click an event button. The event displays a dialog so you can specify particulars about the event. The dialog depends on the type of event; some events have different kinds of subtypes.

In general, the new event appears at the end of the queue—but some kinds of events require that you first select one or more events in the queue. An event button is grayed out if the selection in the queue (or the absence of one) is not legal input to the button's type of event. See the event type descriptions after "Adding Events" later in this chapter.

To highlight an event already in the queue:

- Click its icon, label, or range-bar area.

To delete events from the queue:

1. Highlight the events you want to delete.
2. Click Delete.

To swap two events in the queue:

1. Highlight both events.
2. Click Swap.

   As with adding an event, this operation might not be allowed if the result would be impossible to execute. At the top level of the queue, you can almost always swap events; at lower levels, an event's output must be legal input to its parent event.

Editing an Event in the Queue

Editing an event re-displays the control dialog for the event type, letting you change the event parameters.

To edit an event in the queue:

1. Highlight a *single* event.
2. Click Edit Current Event. The dialog that controls that kind of event appears.
3. Either double-click the event's icon or name in the queue, or double-click the event's range-bar area in the edit window.

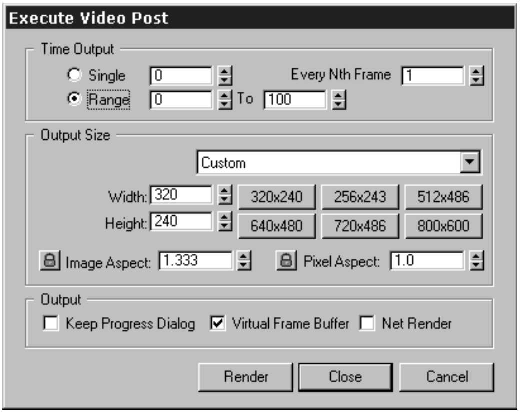
Double-clicking is a shortcut for highlighting and then clicking the Edit Current Event button.

Executing the Queue Sequence

Executing the queue is the final event in creating a post-produced video. The term *execute* is used for this command instead of "render." This is because Video Post output involves rendering only if a Scene event is part of the queue—you can also use Video Post to composite images and animations without including the current 3D Studio MAX scene.

To execute the queue sequence:

1. Click Execute Sequence to execute the queue. An Execute Video Post dialog appears.



2. Set the time range and output size, and then click Render to create the video.

While executing, Video Post minimizes the 3DS MAX window and displays a progress dialog.

3. When execution is done, click Close to dismiss the Video Post progress dialog.

On the Execute Video Post dialog are these areas:

Time Output—Lets you create a single frame, the entire range of frames in Video Post, or selected frames of the video.

Output Size—Lets you specify the size of frames in the final video.

Both the Time Output and Output Size controls are similar to controls in the Render Scene dialog, as are the Output destination controls. See online reference for details.

Note: Although the Execute Video Post controls are similar to those in the Render Scene dialog, its settings are completely independent. Changing a setting in Execute Video Post has no effect on Render Scene, and vice versa.

Virtual Frame Buffer—Renders to the virtual frame buffer window. Default=set.

Net Render—Uses a network for rendering. See the online network rendering documentation for details.

Warning: Video Post output is saved in a file or directed to a device only if you put an appropriate Image Output event in the queue. See “Image Output Events.”

Adding Events

Events in the Video Post queue fall into two overall categories.

- *Scene*, *Image Input*, and *Image Output* events directly handle images and animations; their results are comparable to results you can obtain from 3D Studio MAX for simpler animations, without using Video Post.
- *Image Filter*, *Image Layer*, *External*, and *Loop* events control the composition of the final video, and are more comparable to traditional post-processing work.

Types of Events

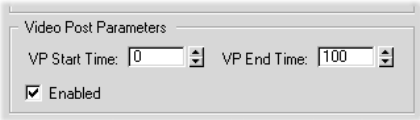
These are the seven types of events:

- *Scene* events include a view of the current 3DS MAX scene or animation in the final video.
- *Image Input* events add still images or animations to the final video. You might have previously created the images by rendering other 3DS MAX scenes.
- *Image Filter* events process the image given to them. The processing can be to change the appearance or use of the input image, or it can create a *transition*, such as a fade, that involves a single Image Input or Scene event.
- *Image Layer* events composite images to appear in the output simultaneously. For example, you can use an Alpha Compositor Layer event to superimpose a bitmap with alpha over the current scene. Or you can use Image Layer to create a transition, such as a crossfade, that involves two Image Input or Scene events.
- *Image Output* events send the result of executing preceding events in the queue to either a saved file or an external device. You must include an Image Output event in the queue if you want to save your work on either disk or video.
- *External* events add an external program to the queue. Typically this might be an image-processing utility that performs a paint operation on each frame before returning control to Video Post.
- *Loop* events cause another event—typically either the Scene or an Image Input event—to loop or ping-pong over the Loop event's range.

Some event types can encompass several individual kinds of events. For example, Image Input events handle a number of file types. There are different kinds of Image Filter events. The various kinds are supported by plug-in software, and your configuration might include more kinds of events than the ones that are supplied with 3D Studio MAX and described in this chapter.

The queue defines the sequence in which events are executed. It is executed from the top down. If an event is a parent to another event, the child event is executed first; if it is a parent to two other events, the two children are executed in top-down order. Changing the sequence of events usually changes the appearance of the final video.

Tip: It might be helpful to think of images that share the same time range as layers, comparable to matted film images in a compositor. The top of the queue is the bottom layer, and the final event in the queue is the top layer. Images that share a time range must be composited with an Image Layer event; otherwise, the second image in the queue "overwrites" the first.



General Video Post Parameters

The dialog for every type of event has an area titled Video Post Parameters. The parameters in this area apply to all kinds of events.

Video Post Start Time—Sets the Video Post frame where the event begins.

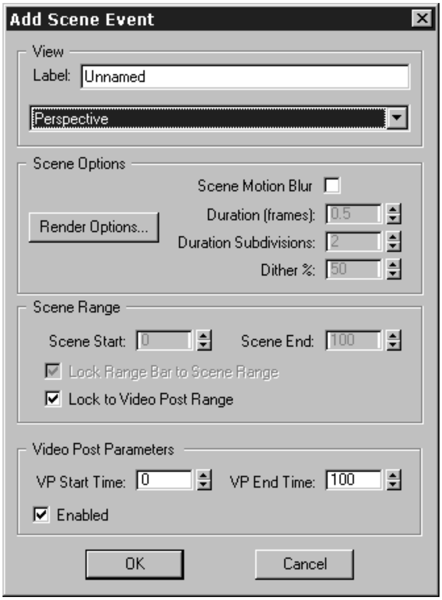
Video Post End Time—Sets the Video Post frame where the event ends.

Changing the value of these parameters is equivalent to dragging the end points of the event's range bar in the edit window, as described in "Arranging Events in Time."

Enabled—When cleared, the event and its range bar are grayed out in the Video Post window, and the event is not used when Execute Sequence creates the final video. Use this to disable the event while keeping it in the queue for possible use in the future. Default=set.

- You can adjust the Start Time and End Time parameters while the event is disabled. The range bars remain grayed out and the event remains disabled.
- The enabled/disabled state of a parent event does *not* enable or disable its children. If you disable a Filter event, for example, its child Image Input event is still enabled unless you explicitly disable it as well.

Scene Events



A scene event is a view of the current 3D Studio MAX scene. You can choose which view to display, and you have several options for how to synchronize the scene with the final video.

Scene events are similar to Image Input events, in the sense that they place an image in the queue. The difference is that the Scene event is the current 3DS MAX scene, and it must be rendered when you execute the Video Post queue. You can have more than one Scene event in the queue.

Tip: If you have more than one Scene event in the queue, and the Scene events occupy the same time range, you must composite them with an Image Layer event; otherwise, the second Scene event "overwrites" the first. However, you can use multiple Scene events to show two different views of the scene simultaneously, or to cut from one view to another.

Adding a Scene Event

Scene events are controlled by the Scene Event dialog.

To add a scene event:

1. First make sure no events are selected in the queue.
2.  Click Add Scene to display the Add Scene Event dialog. Choose which view to use from the View list.
3. Click Render Options to change rendering settings from the way you have set them in the Render Scene dialog.

   Note: Unlike settings in the Execute Video Post dialog, changes you make to the Scene event rendering options *do* change settings in the Render Scene dialog, and vice versa.
4. Set the Scene Range options, described below, and then click OK.

   The Scene event appears at the end of the queue.

Synchronizing the Scene with Video Post

The settings in the Scene Range area of the Edit Scene Event dialog control how scene time is synchronized with Video Post time. There are three overall approaches:

- Match the scene's frame numbers with Video Post frame numbers.
- Offset the scene animation in time, but play it at the same speed as the final video, frame for frame.
- Both offset the scene and change its playback rate, relative to the frames in Video Post.

Matching Frames

To match the scene's frames with Video Post frames, make sure Lock To Video Post Range is set. This is the default.

- Frames in the scene match Video Post frames and have the same frame number. That is, frame 0 in the scene is frame 0 in the Video Post dialog, frame 15 in the scene is frame 15 in Video Post, and so on.
- The range bar for the Scene event represents which portion of the scene is selected. If the range bar covers Video Post frames 25 to 35, executing the queue renders scene frames 25 to 35. Moving the range bar for the scene is like moving a time window within the scene.
- Other Scene Range options are disabled when Lock To Video Post Range is set.

Offsetting Scenes in Time

To offset the scene in time, set Lock Range Bar To Scene Range.

- The Scene Start control is enabled but the Scene End control remains disabled—synchronization is controlled by the Scene Start value and the length of the range bar.
- The Scene Start value is the scene frame number where playback begins. If Scene Start is 0, frame 0 of the scene is the first frame played back; if Scene Start is 12, frame 12 is the first frame to play, and so on.
- The range bar length determines how many frames of the scene to play. Dragging the end point of the range bar changes the length of the playback range. Although Scene End is grayed out, its value updates to show the frame number of the last scene frame that will be played.
- Dragging the range bar changes where the scene is played within the final video. For example, if you set Scene Start to 5 and move the range bar to begin at Video Post frame 20, frame 5 is played at frame 20 of the final video, and so on.

To offset the scene and change scene playback rate, clear Lock Range Bar To Scene Range.

- Both Scene Start and Scene End are now enabled. As before, Scene Start specifies the first scene frame to play. Scene End specifies the last scene frame to play, and the length of the range bar determines playback speed.
- If the range bar specifies the same number of Video Post frames as there are corresponding scene frames, then playback is at the scene's playback rate.
- If the range bar specifies fewer frames, the scene is sped up. If the range bar specifies more frames, the scene is slowed down. When it executes, Video Post automatically skips frames or adds frames to control the speed of scene playback.

  For example, if Scene Start is frame 5 and Scene End is frame 35, the range bar represents 30 frames overall.

  If the range bar covers only 10 Video Post frames, scene playback is sped up to fit 30 frames into 10 of the final video. If, however, the range bar covers 120 frames, scene playback is stretched to slow it down.

Preparing the Scene for Post-Processing

Image Filter and Image Layer events can use masks that are based on graphics buffer (*G-Buffer)* channels instead of the more widely used RGB and alpha channels. Also, some kinds of Filter and Layer events can post-process objects or materials designated by these channels.

You can set two kinds of these channels in the scene itself, to identify and group objects or materials for a particular post-processing effect.

- You set an object's G-Buffer Object Channel value to identify that object—or several objects—to receive a particular post-processing effect.
- You set a material's Material Effects Channel value to identify that material—or several materials—to receive a particular post-processing effect.

You create object-specific or material-specific post-processing by going through the following overall steps:

- Assign a particular Object Channel ID or Material Effects Channel ID to the objects or the materials you want to be post-processed.
- In the Image Filter or Image Layer event, choose the channel ID that associates the event with the ID value you assigned in the scene.

When you execute the Video Post queue, the Filter or Layer event singles out objects or materials that have the designated ID, and performs its post-processing only on those objects or materials.

Warning: Although Image Layer events can use G-buffer channels, not all kinds of Image Layer events *output* channel data. The result of an Image Layer might not preserve channel data for later use.

Setting G-Buffer ID

Begin by selecting the object.

To set an object's G-Buffer Object Channel ID:

1. Right-click the object and then choose Properties from the shortcut menu.
2. In the Object Properties dialog, set G-Buffer Object Channel to a non-zero value, and then click OK. The G-Buffer ID can be any positive integer.

If you give the same G-Buffer ID value to more than one object, all these objects will be post-processed.

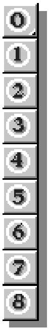
Setting Material Channel ID

To begin, go to the Material Editor and make the material you want to be post-processed the active material.

To set a material's Effects Channel ID:

- Choose a non-zero ID from the Material Effects Channel flyout. The Effects Channel ID can range from 1 to 8.

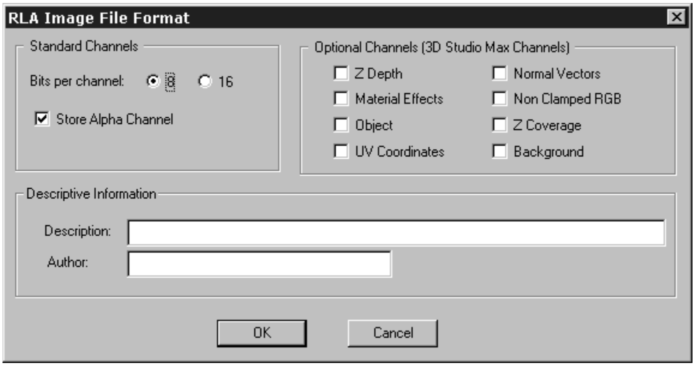
  If you give the same Effects Channel ID value to more than one material, all these materials will be post-processed.



Note: For Multi/Sub-Object materials, post-processing applies at the sub-object/sub-material level. The Effects Channel ID of the parent Multi/Sub-Object material is ignored.

Rendering the Scene event creates an Object or Material Effects channel if an object or material has a non-zero ID in that channel.

To save the channel data with the rendering, use the RLA (*.rla)* file format. (RLA files can save other kinds of graphics buffers as well.)



Setting Up an RLA File

You want to set up an RLA file so it saves Object and Material Effects channel data.

To set up an RLA file:

1. In the Image Output Event dialog, click Files.
2. Choose the *.rla* file type and a file name, and then click Setup.
3. In the RLA Image File Format dialog, set Object, Material Effects, or both, and then click OK.

When the RLA file has saved the Object and Material Effects channels, you can use the rendered scene file as an Image Input event or a Filter or Layer mask, and continue to use the Object or Material Effects Channel data.

Adding Scene Motion Blur

Scene motion blur is a Scene event setting that adds the special effect of blurring or “trails” to moving objects. Scene motion blur differs from object motion blur, which is an object property that you render. Scene motion blur applies blur at the scene level, taking camera movement into account, which object motion blur does not.

Use scene motion blur as follows:

- In general, use scene motion blur as a special effect to strongly emphasize motion.
- Use object motion blur to smooth the rendering of rapidly moving objects.

You can use both kinds of blur in the finished animation.

- To add scene motion blur, set Scene Motion Blur in the Scene Event dialog.
- Set the scene motion blur parameters.

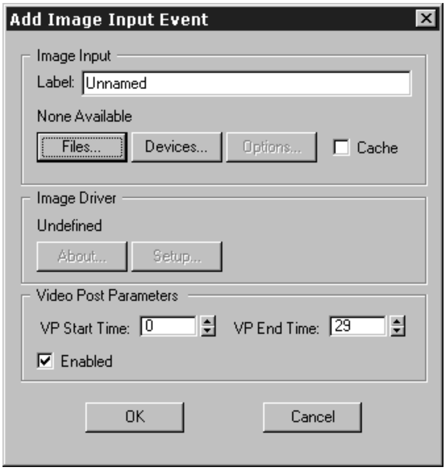
The Scene event generates motion blur by simulating a camera with an open shutter. It interpolates and then renders movement *within* a frame, to generate a series of images of the moving object, instead of the default single image.

Duration—Sets the number of frames the simulated “camera shutter” is open. Increase this value to exaggerate the motion blur effect. Decrease it to make the blur more subtle. Range=0.0 to 2.0. Default=0.5.

Duration Subdivisions—Sets the number of subframe “slices” that are interpolated and rendered within the Duration period. Range=2 to 160. Default=2.

Dither %—Displays the percentage of dithering used when rendering the motion-blur slices or “trails.” Default=50%

Image Input Events



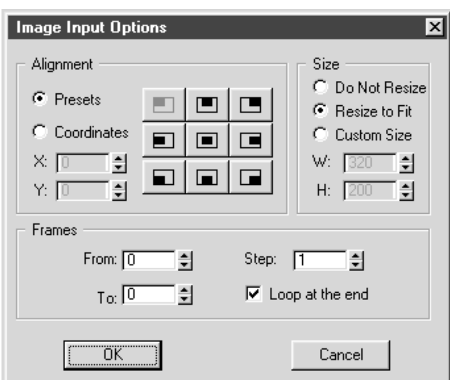


Image Input events add a still or moving image to the scene. Image Input events are similar to Scene events, in the sense that they place an image in the queue. The difference is that the image is either a file that was saved beforehand, or a device-generated image.

Image Input events are controlled by the Image Input Event dialog.

Adding an Image Input Event

First make sure no events are selected in the queue.

To add an image input event:

1.  Click Add Image Input Event to display an Add Image Input Event dialog.
2. Click Files to choose a bitmap or animation as the image, or click Devices to choose an image-generating device.

   If you click Files, a file dialog appears to let you choose the bitmap or animation file.

   If you choose Devices, a Select Image Input Device dialog appears. This dialog has a list of installed device options.
3. Click Options to choose the size and placement of the image in the final video frames.

   An Image Input Options dialog appears. See “Image Input Option Controls,” following.
4. Adjust other Image Input settings, and then click OK.

   The Image Input event appears at the end of the queue.

Tip: Think of images that share the same time range as layers, comparable to matted film images in a compositor. Images that share a time range must be composited with an Image Layer event; otherwise, the second image in the queue “overwrites” the first.

These are the other controls in the Image Input Event dialog.

Cache—Set this if you have chosen a single-image bitmap. Cached images are stored in RAM so Video Post doesn’t have to reload or scale them for each frame.

Image Driver—The two buttons in this area are enabled only when you have chosen a device as the image source. Click About to see information about the device's 3DS MAX plug-in. Click Setup to use the device-specific setup dialog.

Note: The device, its driver, and its 3DS MAX plug-in must all be installed on your system for you to use device input.

Image Input Option Controls

When you click Options, an Image Input Options dialog appears. You use this dialog to set the image's size and placement relative to the frames of video output. For animated input images, you also use it to synchronize the Image Input event with the frame sequence of video output.

Aligning the Input Image

To align the input image, do one of the following:

- Choose Presets and then click one of the preset alignment buttons.
- Choose Coordinates and then enter the X,Y coordinates for the image's location.

  The upper-left corner is (0,0) for both the input image and the output frame. Increasing X moves the image to the right, and increasing Y moves the image down. Negative values move the image in the opposite direction. X and Y values specify pixels.

Setting the Input Image Size

To set the input image size, do one of the following:

- Choose Do Not Resize to maintain the image's original resolution.
- Choose Resize to Fit to change the image size to match the output frame. This can change the image resolution, causing it to be rescaled for every frame.
- Choose Custom Size and then enter the width and height of the image in the output frame.

Controlling Playback

To control playback of an animated image:

- Set the input animation frame range and speed. See "Input Playback Controls," following.
- Set Loop At The End if you want the animation to repeat. Clear Loop At The End if you want the animation to stop after playback. This option applies only when the input animation is shorter than the final video.

Input Playback Controls

The controls in the Frames area of the Image Input Options dialog are for synchronizing an animated input image with the Video Post video.

From—Input animation frame number where playback begins.

To—Input animation frame number where playback ends.

Step—Plays every *nth* frame.

These controls specify only frame range and speed for the animated image—they do not specify where in the Video Post frame sequence the animation will appear. To do that, adjust the range bars in the edit window or use the Image Input Event dialog's Video Post parameters.

Image Filter Events

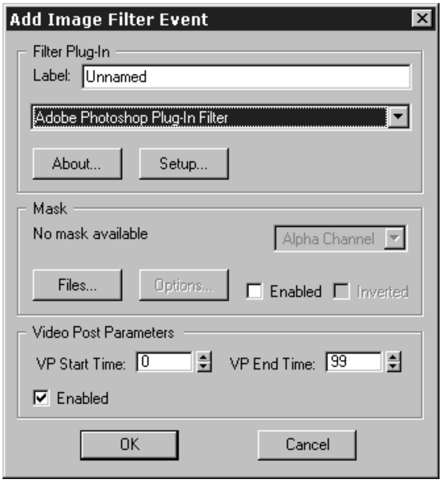


Image Filter Events provide image processing for images and scenes. Several kinds of filters are provided. For example, the Negative filter inverts the colors of an image, and the Fade filter fades an image in or out over time.

An Image Filter event is usually a parent event with a single child—the child can itself be a parent with children.

Valid children for an Image Filter event are:

- A Scene event
- An Image Input event
- A Layer event that contains Scene or Image Input events
- A Filter event that contains Scene or Image events

You can also add an Image Filter without a child event. In this case, the Image Filter processes the result of previous events in the queue.

Adding an Image Filter Event

Image Filter events are controlled by the Add Image Filter Event dialog.

To add an image filter event:

1. Select a valid child event, or make sure no event is selected in the queue.

2.  Click Add Image Filter Event to display an Add Image Filter Event dialog.

3. Choose the kind of filter you want from the Filter Plug-In list.

If the Setup button is enabled for this kind of filter, click Setup to set the filter options.

4. Choose a mask if you want the filter to be masked or if the kind of filter you're using requires it. See "Using Masks in Filter and Layer Events."

5. Adjust other Image Filter settings, and then click OK.

If you selected a child event, the Image Filter event becomes its parent. If no event was selected, the Image Filter event appears at the end of the queue.

Kinds of Filters

3D Studio MAX provides many kinds of filters. Filters are plug-ins, and additional filters might be available on your configuration.

To set up a filter:

1. Choose the kind of filter by choosing its name from the Filter Plug-In list.

2. Click About to display information about the chosen filter.

3. Click Setup to set the parameters for the chosen filter. Setup is grayed out if the filter has no parameters.

Tip: For the time-based filters, Fade and Simple Wipe, align the range bar of the underlying image or scene with the start or end of the filter event, and make sure the image's range is at least as long as the range bar for the filter event itself.

Adobe Photoshop Plug-In Filter—Provides access to image-manipulation filters for Adobe Photoshop from third-party developers. See online reference for use.

Adobe Premiere Video Filter—Provides access to animated image-manipulation filters for Adobe Photoshop from third-party developers. You can use a single filter event in the queue to apply multiple Premiere Video filters. See online reference for use.

Contrast—Provides contrast and brightness controls. See online reference for setup options.

Fade—Fades an image in from black or out to black over time. The rate of the fade is determined by the length of the Fade filter's time range. Setup options are Choose In to fade in or Out to fade out.

Glow—Adds a glow to a material or object, making it appear self-illuminating and with a halo. See online reference for setup options.

Image Alpha—Replaces the image's alpha channel with the channel specified by the filter mask. If you don't choose a mask, this filter has no effect and is not added to the queue. No setup options.

Negative—Inverts the colors in the image, making it negative. Using parametric animation, you can blend the amount of the negative effect.

Pseudo Alpha—Creates an alpha channel for the image based on the image's first pixel (the upper-left corner pixel). All pixels that have the same color as this pixel become transparent. No setup options.

Because only *one* pixel color becomes clear, edges of the opaque areas are aliased. The main use for this filter is when you want to composite a bitmap whose format does not have an alpha channel.

There is also a Pseudo Alpha Layer Event. See the next topic.

Simple Wipe—Either reveals or erases the image by a wipe transition over time. The rate of the wipe is determined by the length of the Wipe filter's time range. The area not covered by the image renders as black unless you use an Image Layer event to composite the Wipe filter with another image. See online reference for setup options.

There is also a Wipe Layer event. Unlike Wipe Layer, the Wipe Filter wipes across a fixed image. See the next topic.

Starfield—Generates a realistic starfield with optional motion blur. This filter requires a camera view. Star motion results from moving the camera.

Image Layer Events

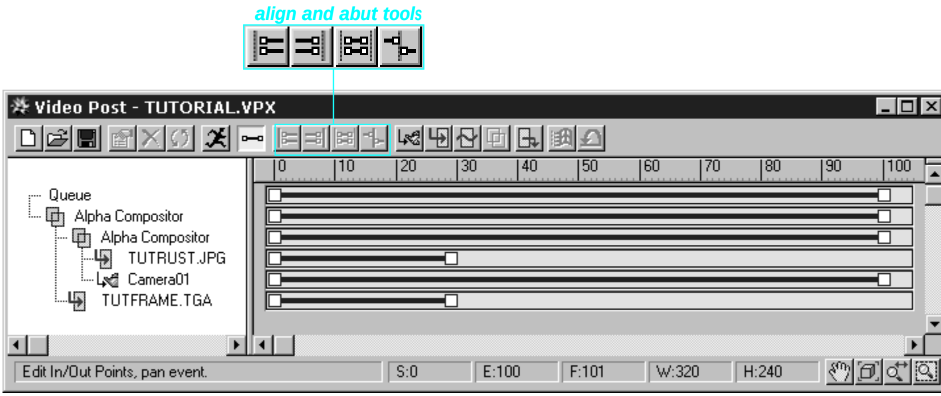


Image Layer Events composite other images and scenes. Several kinds of layer events are provided. For example, the Alpha Compositor uses the top image's alpha channel, and the Cross Fade Transition fades one image out and the other image in, over time.

An Image Layer event is always a parent event with two children—children can themselves be parents with children.

Valid children for an Image Layer event are:

- Scene events
- Image Input events
- Layer events that contain Scene or Image Input events
- Filter events that contain Scene or Image events

Image Layer events are controlled by the Image Layer Event dialog.

Tip: The order of the children of an Image Layer event is important. Think of the first image in the queue as the bottom layer and the second image as the top layer. Images that share a time range *must* be composited with an Image Layer event; otherwise, the second image in the queue "overwrites" the first.

Adding an Image Layer Event

First make sure the two child events are in the order you want the Image Layer event to use them.

To add an image layer event:

1. Select the two events. Click to select the first event, then hold CTRL and click to select the second.

2.  Click Add Image Layer Event. An Add Image Layer Event dialog appears.

3. Choose the kind of layer event you want from the Layer Plug-In list. If the Setup button is enabled for this kind of layer event, click Setup to set the options.

4. Choose a mask if you want the layer event to be masked. See the next topic, "Using Masks in Filter and Layer Events."

5. Adjust other Image Layer settings, and then click OK.

The Image Layer event becomes the parent of the two child events you selected.

Kinds of Layer Events

You choose the kind of layer event by choosing its name from the Layer Plug-In list.

To choose the kind of layer event:

1. Click About to display information about the chosen layer event.
2. Click Setup to set the parameters for the chosen layer event.

3D Studio MAX provides six kinds of layer events. Because layer events are plug-ins, additional layer events might be available on your configuration.

Tip: For the time-based layer events such as Cross Fade and Simple Wipe, align the range bars of the underlying images or scenes with the start or end of the layer event, and make sure the images' ranges are at least as long as the range bar for the layer event itself.

Adobe Premiere Transition Filter—Provides access to transition filters for Adobe Premiere from third-party developers. See online reference for use.

Alpha Compositor—Composites the two images using the alpha channel of the second image. The first image appears in areas where the second image's alpha channel is transparent. No setup options

Cross Fade Transition—Composites the two images over time, crossfading from the first image to the second image. The rate of the crossfade is determined by the length of the Cross Fade Transition filter's time range. No setup options.

Pseudo Alpha—Composites the two images by creating an alpha channel for the second image based on the image's first pixel (the upper-left corner pixel). All pixels that have the same color as this pixel become transparent in the second image. No setup options.

Because only *one* pixel color becomes clear, edges of the opaque areas in the second image are aliased. The main use for this layer event is when the second image is a bitmap whose format does not have an alpha channel.

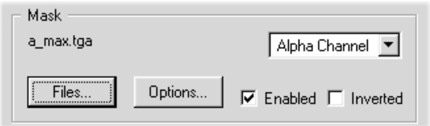
Simple Additive Compositor—Composites the two images using the second image's intensity (HSV value) to determine transparency. Areas of full intensity (255) are opaque; areas of zero intensity are transparent; and areas with intermediate transparency are translucent. No setup options.

This layer event can be useful when the second image is a bitmap whose format does not have an alpha channel.

Simple Wipe—Either reveals or erases the second image by a wipe transition. The rate of the wipe is determined by the length of the Simple Wipe layer event's time range. See online reference for setup options.

There is also a Wipe Filter event. Unlike Wipe Filter, the Wipe Layer event moves the image, sliding it in or out.

Using Masks in Filter and Layer Events



You can apply a mask to Image Filter and Image Layer events. The filter or layer effect is applied where the mask is at zero strength; it is not applied where the mask is at full strength; areas of intermediate mask strength are partially filtered or layered.

Some events, such as Image Alpha Filter, require a mask in order to work correctly.

Choosing a Mask File

You choose masks in the Mask area of the Filter Event and Layer Event dialogs.

To choose the mask file:

1. Click Files, and use the file dialog to choose the mask file, and then click OK.

2. Choose the channel to use from the list of channels. See “Mask Channels,” following.

To position or resize the mask:

- Click Options.

  An Image Input Options dialog appears, identical to the dialog you use with Image Input events. See “Image Input Events.”

If the mask is animated, you also use this dialog to specify its time range and playback speed.

Enabled—When cleared, the mask is not used. Default=set.

Inverted—When set, the mask is inverted. Default=cleared.

Mask Channels

You can choose the channel to use for the mask. This depends on which type of file you have chosen as the mask.

- *Bitmaps* (including animations).

  You can use the Alpha channel, the Red, Green, or Blue channel, or Luminance.

- *.rla* files

  In addition to Alpha, Red, Green, Blue, or Luminance, you can use the graphics buffer (G-Buffer) channels, Z-Buffer, Material Effects, or Object.

Image Output Events

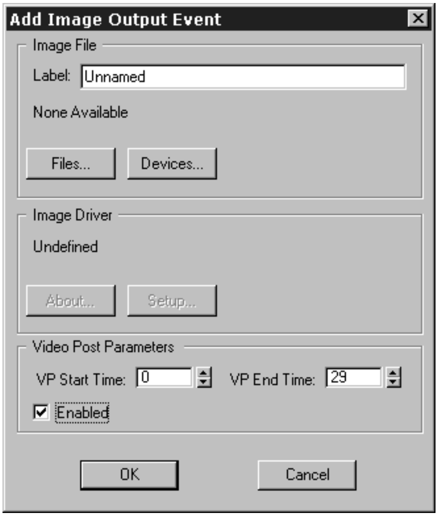


Image Output events send the result of executing the Video Post queue to a file or a device.

You *must* add an Image Output event to the queue if you want to save the final video. With no Image Output, the results of execution are only displayed in the virtual frame buffer window.

The Image Output event saves the results of post-processing up to its location in the queue. If the Image Output event is not at the end of the queue, it saves an intermediate result. The queue can include more than one Image Output event.

Warning: The range bar of the Image Output event must include the entire range of frames you want to output. The Image Output event does not output frames outside its range.

Adding an Image Output Event

Image Output events are controlled by the Add Image Output Event dialog.

Image Output disregards whether any events in the queue are selected or not.

To add an image output event:

1.  Click Add Image Output Event.

2. Click Files to save the final video in a file, or Devices to send the video to a device.

If you click Files, a file dialog appears to let you choose the bitmap or animation file.

If you choose Devices, a Select Image Output Device dialog appears. This dialog has a list of installed device options.

3. Adjust other parameters, and then click OK.

The Image Output Event appears at the end of the queue.

Device Controls

If you choose a device, its configuration controls are enabled.

Image Driver—The two buttons in this area are enabled only when you have chosen a device as the image source.

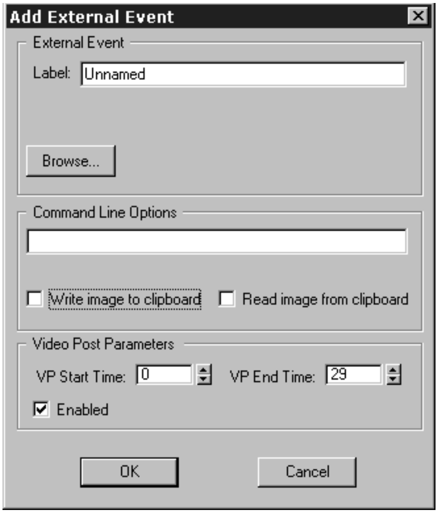
To choose a device:

1. Click About to see information about the device's 3DS MAX plug-in.

2. Click Setup to use the device-specific setup dialog.

Note: The device, its driver, and its 3DS MAX plug-in must all be installed on your system for you to use device output.

External Events



An External event is typically a program that performs image processing. It can also be a batch file, or a utility you want to run at a specific point in the queue. You can also use an External event to transfer images from or to the Windows clipboard.

An External event is always a *child* event. If you select an event in the queue before you add the External event, the External event becomes the selected event's child. (Remember that children are evaluated before their parents.)

External events are controlled by the Add External Event dialog.

Adding an External Event

Select an event, or make sure no event is selected in the queue.

To add an external event:

1.  Click Add External Event to display an Add External Event dialog appears.

2. Click Browse to display a file dialog.

3. In the file dialog, choose the external program you want to execute, and then click OK. You have these options:

- If the external program accepts command-line options, enter these in the Command Line Options field. The command line can contain special options, described below.
- If you want the external program to read the current Video Post image, set Write Image To Clipboard.
- If you want Video Post to use the result of the external program, set Read Image From Clipboard.

If you selected an event, the External event becomes its child. If no event was selected, the External event appears at the end of the queue.

Warning: The image that the External event reads from the clipboard is placed in the Video Post queue. If the external program does not do what you want, this can erase or overwrite the result of all Video Post post-processing.

Command-Line Options

You can include options in the command line that send real-time data to the external program. The options are placeholder strings that are converted to 4-digit numbers when the external event is executed. These options are:

%f—Current frame number.

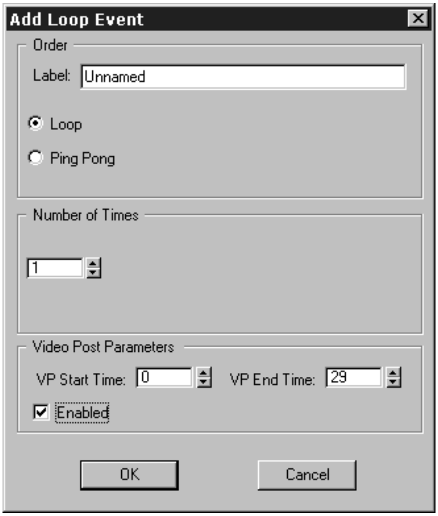
%w—Width of the Video Post image.

%h—Height of the Video Post image.

For example, if the command line contains the string **–w%w –h%h –oframe%f.tga**, the string sent to the external program might be:

–w0640 –h0480 –oframe0001.tga

Loop Events



Loop events cause other events to repeat over time in the output video. They control sequencing, but perform no image processing.

A Loop event is always a parent event with a single child—the child can itself be a parent with children. Any type of event can be the child of a Loop event, including another Loop event.

Adding a Loop Event

Loop events are controlled by the Add Loop Event dialog.

To add a loop event:

1. First select the child event.

2.  Click Add Loop Event. An Add Loop Event dialog appears.

3. Choose the loop settings, and then click OK. The Loop event appears as the parent of the selected event.

The Loop event repeats the child event over the course of the Loop event's range.

Loop—Repeats the child event by starting it over when the child event reaches the end of its range. For example, if the child event has four frames, they play in the order 0123 0123, and so on. Default=Loop.

Ping Pong—Repeats the child event by playing it first forward, then backward, then forward, and so on. To make the action smoother, the last frame is not repeated. For example, if the child event has four frames, they play in the order 0123 210 123, and so on.

Number of Times—The number of times the loop is repeated. This value is in *addition* to the first time the child event plays back.

Note: The Loop Event's range bar displays the original duration of the child event's playback in color (blue if unselected, red if selected), and the range of looped frames in gray.

You can change the duration of the child event's playback by dragging the child's frame range *or* the child's original range in the Loop Event's track, but you can adjust the full length of the loop (the gray portion of its range bar) only by changing the Number Of Times parameter.

Arranging Events in Time

Range bars control when events occur in the final output video, and whether they occur in sequence or simultaneously. You can select range bars, change their position or their length, or use the align and abut tools while the Edit Range Bar button is active.

- When they are unselected, range bars are displayed in blue.
- When they are selected, range bars are displayed in red.

Selecting and Adjusting Range Bars

Selecting a range bar begins the procedures for adjustment.

To select a range bar, do one of the following:

- Click the range bar.
- Click the associated event name or icon in the queue.

To select multiple range bars:

1. Select the first range bar.

2. Hold CTRL and click additional range bars.

To select multiple contiguous range bars:

1. Select the first range bar.

2. Hold SHIFT and click another range bar. Both range bars and all range bars in between them are selected.

Base range—In a multiple selection, the last range bar you select becomes the *base range*. The base range is displayed with red squares in its endpoints. The align commands, described next, use the base range.

To move a range bar:

- Drag the center of the range bar left or right.

To change the length of a range bar:

- Drag one of its endpoints left or right.

  If multiple range bars are selected, dragging one endpoint changes all range bars.

Aligning and Abutting Event Ranges

The four buttons to the right of Edit Range Bar are the align and abut tools. These buttons move or resize range bars based on the range bars' relative positions. They are grayed out unless two or more range bars are selected.

The first three align tools base their operation on the base range (with red squares at its endpoints). The abut tool does not.

Align Selected Left—Aligns all selected range bars to the start frame of the base range.

Align Selected Right—Aligns all selected range bars to the end frame of the base range.

Make Selected Same Size—Makes all selected range bars the same length as the base range. The relative position of the range bars doesn't change.

Abut Selected—Moves the selected range bars so their end points abut in sequence. This command disregards the base range—the first range bar in the queue comes first, followed by the next in sequence, and so on.

A

Appendix

This appendix provides in-depth detail in three areas:

- Theory of objects in 3D Studio MAX, including Object Space and World Space, and object data flow. How copies, instances, and references are derived.
- Mathematical concepts related to the Expressions controller, including a review of trigonometric functions. Lists of operators, functions, and predefined variables for the Expressions controller.
- Information related to system setup and configuration. Includes topics on the hardware lock, display drivers, and command-line options for starting 3DS MAX.

Objects and Copies in 3D Studio MAX

The topics in this section cover the underlying theory of how 3D Studio MAX handles the conversion of objects into copies, instances, and references. Concepts such as instancing also apply to maps and materials, modifiers, and animation controllers.

This topic provides some basic definitions as a foundation and background for the later topics.

Object Space and World Space

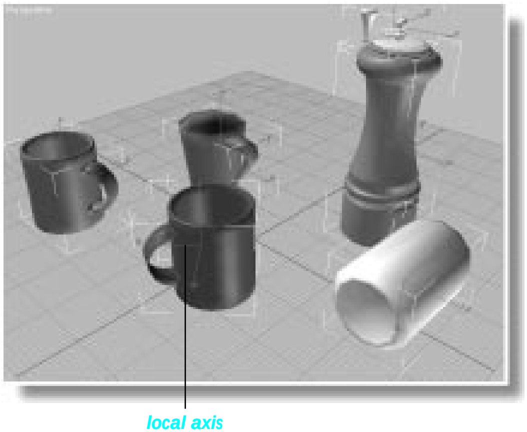
There are two spatial coordinate systems at work in 3D Studio MAX. These two systems are called *Object Space* and *World Space*.

Object Space

Object space is the coordinate system unique to each object in your scene that tracks the location of everything applied to an object. The location of object vertices, the placement of modifiers, mapping coordinates, and materials are all defined in object space.

As mentioned in the previous topic, each object has its own local center and coordinate system as defined by the location and orientation of the object's pivot point. The local center and coordinate system of an object combine to define its *object space*.

- When you choose Use Pivot Point Centers from the toolbar or Use Pivot Points from the Modify panel you are telling 3DS MAX to use the Object Space origin of one or more selected objects as the center of a transform or modifier effect.
- When you choose Local from the coordinate systems pulldown of the toolbar, you tell 3DS MAX to use a selected object's object space for the orientation of the active coordinate axes.



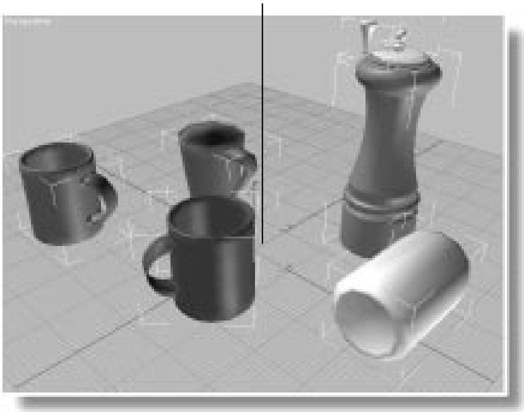
Local object center

World Space

World space is the universal coordinate system that 3DS MAX uses to track objects in the scene. When you look at the Home Grid in the 3DS MAX viewports, you see the World Space coordinate system. world space is constant and immovable.

All objects in your scene are located in world space by their position, rotation, and scale (their transforms).

Space warps also operate in world space. A space warp defines an area in world space that is affected by the space warp's parameters. Any object that is bound to the space warp is affected as it moves through the space warp's area of world space.



World space center

Modifiers and Transforms

Modifiers are applied in Object Space and Transforms in World Space, but they differ in more ways than which space they are applied in. They also differ in how they affect an object and the order in which they are applied to an object.

Modifiers

Modifiers are operations that act on the internal structure of an object in object space. For example, when you apply a modifier such as Twist to a mesh object, the position of each vertex of the object is individually changed in object space to produce the twist affect.

Modifiers operate at the sub-object level and are dependent on the internal structure of the object when the modifier is applied.

Modifiers have the following properties. They are:

- Applied to all of an object, or part of an object (using sub-object selection).
- Dependent on the order of application. Applying a Bend followed by a Twist produces a result different from applying a Twist followed by a Bend
- Displayed as individual entries in the modifier stack where you can turn them on and off, and apply them in a specific order.

Transforms

Transforms are the most basic of 3D manipulations. Unlike modifiers, transforms are independent of an object's internal structure; they act directly on the object's local coordinate system.

The local coordinate system of an object can be expressed as a matrix of values that specify the following information in world space:

- Position of the object's center in world space
- Rotation of the object in world space
- Scale of the object along its local axes

The matrix is called the *transformation matrix* and its information relates directly to the transforms of Move, Rotate, and Scale. These transforms represent addition and multiplication of the values in the transformation matrix.

Transforms have the following properties. They are:

- Applied to the entire object.
- Stored as single values independent of the order of application. No matter how many times you transform an object, the results are stored as one set of values in the matrix.
- Applied after all object modifiers have been evaluated.

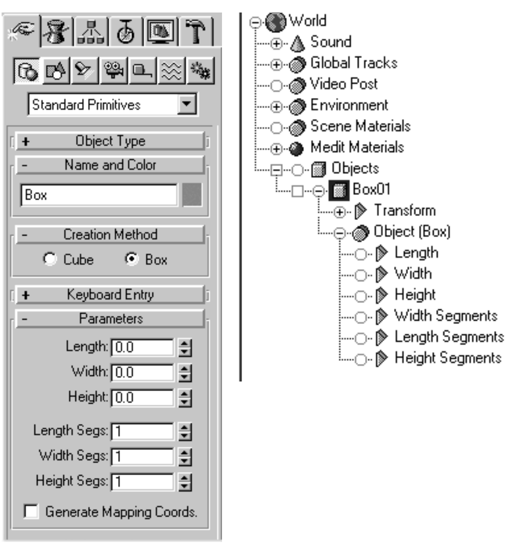
Understanding Object Data Flow

Once you have defined an object, 3D Studio MAX evaluates changes affecting the base object and displays the result in the scene. What these changes are, and the order in which they are evaluated, is called the *object data flow*.

Master Object

Master object refers to an object defined by a set of creation parameters and the original position and orientation of its pivot point. You never see the master object. What you see on the screen is always the result of at least the following data flow:

Master Object
-> Object Transforms
-> Object Properties

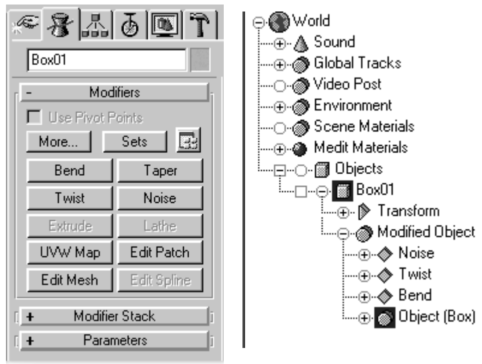


Object creation parameters in the Create panel and Track View

Object Modifiers

The Object Modifiers are the next group evaluated in the data flow. Each modifier is evaluated in the order it was placed on the Modifier Stack. The modifications all occur in the object's object space and the result is called the *modified object*.

mp to replace Track View in following illo]

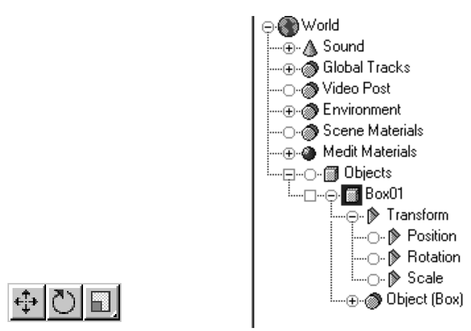


Object modifiers in the Modify panel and Track View

Object Transforms

Once the modified object has been evaluated, it is transformed within the world coordinate system. Transforms cover the position, rotation, and scale changes applied from the transform buttons on the toolbar.

replace Track View in following illo]



Object transforms in the toolbar and Track View

The method of evaluating all modifiers first and then evaluating the combined transforms has important ramifications for the way you work with 3D Studio MAX.

Transforms are completely independent from the order of application. For example, imagine a cylinder that is 40 units high. If you scale the cylinder along its *Z* axis, increase its height, and then bend the cylinder 180 degrees, you get the same result as if you bent the cylinder first and then scaled it. The scale transform is always evaluated after the bend modifier.

If you want to apply a transform that is evaluated in a specific order in the Modifier Stack, use the XForm modifier. See chapter 16, “Applying Geometric Modifiers”.

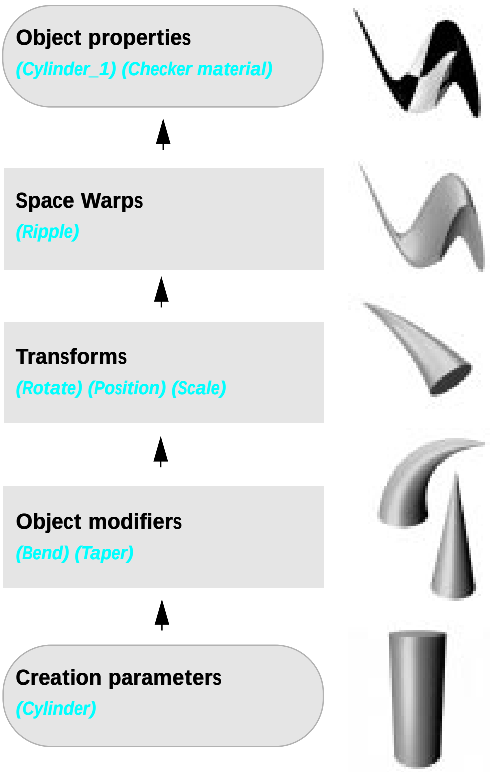
Space Warps

Space warps are evaluated after transforms. They distort objects bound to the space warp based on the position of the object in world space. For example, a Wave space warp causes the surface of an object to undulate in the form of a wave. As the object moves through world space, the wave moves across the object’s surface.

Object Properties

Object properties are the last to be evaluated before the object is displayed. These are properties unique to each named object such as its name, wireframe color, and shadow casting properties. This is the end of the data flow and the result is the *named object* you see in your scene.

- Right-click an object and choose Properties to display its Object Properties menu.



Data flow diagram of a typical 3DS MAX object

Understanding Object Copies, Instances, and References

3D Studio MAX objects can be copied, instanced, or referenced. The difference between a copy, instance, and reference lies in how the data flow branches on its way from the master object to the named object.

Evaluating Copies

When you copy a 3DS MAX object, you create a new, independent master object and data flow resulting in a new named object. The copy duplicates all of the data of the original object at the time it is copied. The copy has no connection to the original object.

Evaluating Instances

When you instance an object, you are displaying multiple named objects based on a single modified object. Each named object instance has its own set of transforms, space warp bindings, and object properties, but it shares the object modifiers and master object with the other instances. The data flow for an instance branches just after evaluating object modifiers.

Because instances share the same master object and object modifiers, changing the creation parameters or applying a modifier to any one instance changes all related instances.

Evaluating References

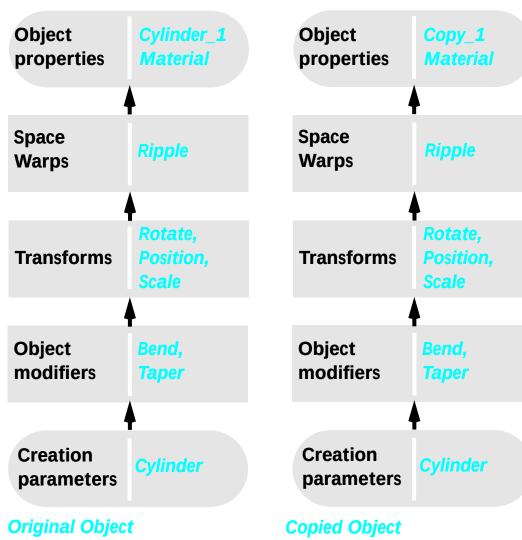
When you reference an object you split the object modifiers into groups of shared and unique modifiers. Like instances, references share the same master object and some object modifiers. The data flow for a reference branches just after the object modifiers but then evaluates a second set of object modifiers unique to each reference.

The data flow for references depends on where in the Modifier Stack the reference is made. Each time you create a reference, the data flow branches after evaluating all of the current object modifiers and adds a container for another set of object modifiers. In the Modifier Stack you can see where branching occurs by looking for a dashed line between modifiers.

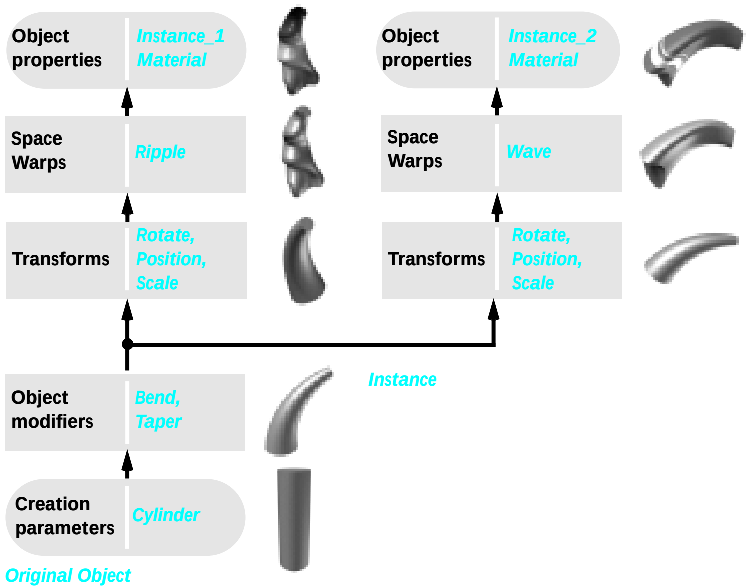
Because references share the same master object, changing the creation parameters for any reference changes all related references.

The results of changing or applying a modifier to a named object reference depends on where in the Modifier Stack it is applied:

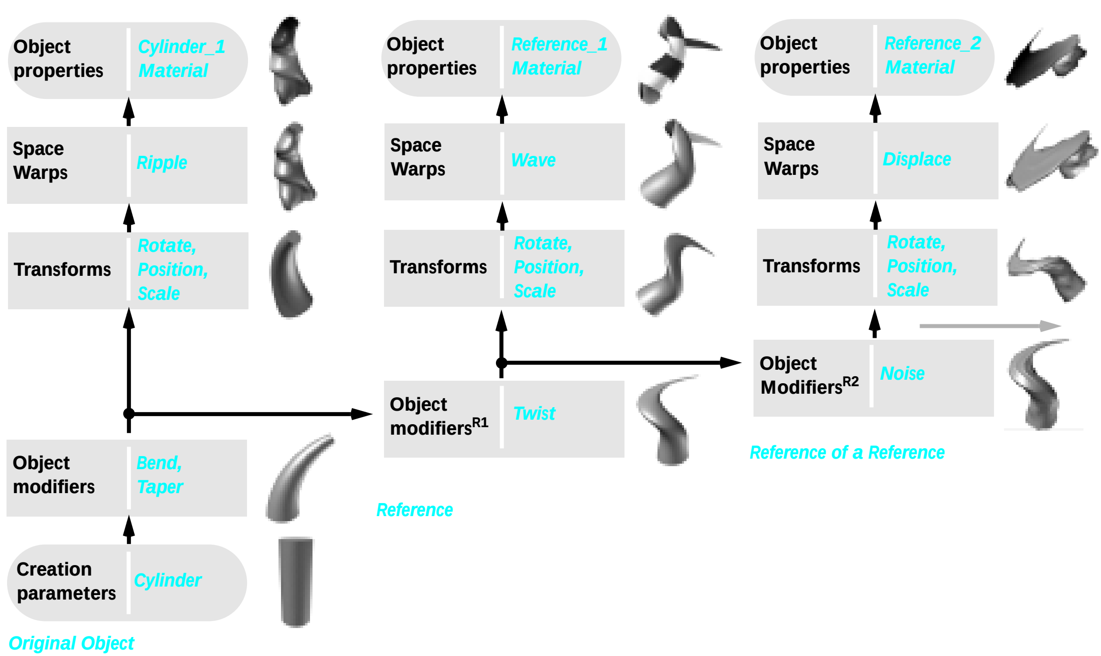
- Applying a modifier to the top of the Modifier Stack affects only the selected named object.
- Applying a modifier below a dashed line affects all references branching above that line.
- Applying a modifier at the bottom of the Modifier Stack affects all references derived from the master object.



Copy



Instance



Reference and Reference of a Reference

Expression Controller Review and Reference

The topics in this section present a brief review of the mathematics used by the Expression controller. See chapter 27, "Working with Controllers," for an introduction to this controller.

As a reference, operators are listed below. Functions and predefined variables are listed in the next topic.

Operators

The meaning of some operators depends on whether the values are scalar or vector values.

In the tables, p and q are any scalar value or expression, V and W are any vector value or expression. (The character "x" is used as the vector cross-product operator.)

Scalar Operators

These are the arithmetic operators for scalar values:

Operator	Use	Meaning
+	p+q	addition
-	p-q	subtraction
-	-p	additive inverse
*	p*q	multiplication
/	p/q	division
^	p^q	power (p to the power of q);
**	p**q	^ and ** are the same operation

You can also use logical (Boolean) operators with scalar values. These operators all return 1 if true, 0 otherwise:

Operator	Use	Meaning
=	p=q	equal to
<	p<q	less than
>	p>q	greater than
<=	p<=q	less than or equal to
>=	p>=q	greater than or equal to

<table>
<tr><th>Operator</th><th>Use</th><th>Meaning</th></tr>
<tr><td>|</td><td>p|q</td><td>logical OR
returns 1 if either p or q is nonzero; otherwise, returns 0</td></tr>
<tr><td>&</td><td>p&q</td><td>logical AND
returns 1 if p and q are both nonzero; otherwise, returns 0</td></tr>
</table>

Note: Logical operators are useful with the "if" function.

Vector Operators

For vectors that have a variable name, you can use a special component operator (.) to refer to the three scalar components of the vector:

Use	Meaning
V.x	first component (*X*)
V.y	second component (*Y*)
V.z	third component (*Z*)

These are the operators for vector arithmetic:

Operator	Use	Meaning
+	V+W	addition
-	V-W	subtraction
*	p*V V*p	scalar multiplication
*	V*W	dot product
x	VxW	cross product
/	V/p	scalar division

Operator Precedence

When you evaluate an expression, 3D Studio MAX first scans the expression to find what the components are, and in what order they must be evaluated. The order is determined by operator precedence—the higher the precedence of an operator, the earlier it is evaluated.

3D Studio MAX expressions use eight levels of precedence, with the operators arranged as follows, from highest at the top to lowest at the bottom of the list:

- +	**as unary operators—e.g., -8, +25**
.	the component operator—as in V.x
** ^	
x	cross product
* /	
+ -	
= < > <= >=	
| &	

Parentheses are a special case. They are a grouping or subexpression operator that is provided so you can override the precedence order of the other operators.

Functions and Predefined Variables

For reference purposes, this topic lists all the functions provided for 3D Studio MAX expressions. There is also a list of predefined variables.

Functions

In this list, p, q, and r represent scalar values or scalar expressions; V and W represent vector values or vector expressions.

Note: You can click Function List in the Expression Controller dialog to display a list of expression controller functions online.

Trigonometric Functions

The sine, cosine, and tangent functions take an angle in degrees and return a floating-point value. The arc functions take a floating-point value and return a value in degrees.

sin(p)	sine
cos(p)	cosine
tan(p)	tangent
asin(p)	arc sine
acos(p)	arc cosine
atan(p)	arc tangent

Hyperbolic Functions

Hyperbolic functions take a floating-point value and return a floating-point value.

sinh(p)	hyperbolic sine
cosh(p)	hyperbolic cosine
tanh(p)	hyperbolic tangent

Conversion Between Radians and Degrees

radToDeg(p)	takes p in radians and returns the same angle in degrees
degToRad(p)	takes p in degrees and returns the same angle in radians

Rounding Functions

ceil(p)	smallest integer greater than or equal to *p*
floor(p)	largest integer less than or equal to *p*

Standard Calculations

ln(p)	natural (base e) logarithm
log(p)	common (base 10) logarithm
exp(p)	exponential function — exp(p) = e^p
pow(p,q)	p to the power of q (p^q)
sqrt(p)	square root
abs(p)	absolute value
min(p,q)	minimum — returns p or q, depending on which is smaller
max(p,q)	maximum — returns p or q, depending on which is greater
mod(p,q)	remainder of p divided by q

Conditional

if(p,q,r)	works like the common spreadsheet “if” — if p is non-zero then “if” returns q, otherwise “if” returns r

Vector Handling

length(V)	the length of V
comp(V,i)	i’th component (i=0,1,2)— comp([5,6,7],1) = 6
unit(V)	returns a unit vector in the same direction as V
vif(c,V1,V2)	“vector if” value is V1 if c is true (non-zero), otherwise V2. Test statement c should be a scalar. The results are vectors.

Note: The comp function is an alternative to the notation V.x, V.y, V.z.

Special Animation Function

noise(p,q,r) — 3D noise—returns a randomly generated position

p, *q*, and *r* are arbitrary values used as a random-generation seed; re-use these values to ensure that noise() returns the same value

Predefined Variables with Variable Values

These are the predefined variables that have a variable—time-based—value in 3D Studio MAX expressions (variable names are case sensitive):

F — Frame number.
For each frame, F equals the current frame number, counting from zero. The range of frames can vary depending on the number of frames in the active time segment.

NT — Normalized time.
By definition, normalized time (NT) ranges from 0 to 1 over the active time segment, regardless of how many frames are in the segment. If you base an expression on NT, its effect happens exactly once over the range. You can also multiply NT by a factor to cause the expression's effect to happen a certain number of times (for example, 2*NT causes the expression's effect to happen twice). Expressions based on NT speed up or slow down if you later change the length of the time segment.

S — Seconds—elapsed time in seconds.
Elapsed time is measured from the first frame to the current frame. The range of seconds can vary depending on the total time of the active time segment.

T — Ticks—elapsed time in ticks.
There are 4800 ticks per second. Elapsed time is measured from the first frame to the current frame. The range of ticks can vary depending on the total time of the active time segment.

Expression Controller Techniques

The first part of this topic summarizes some useful expression techniques. The next part of the topic reviews trigonometric functions and vector arithmetic—if you're familiar with these subjects you can skip the review topics.

Commonly Used Expressions

This topic lists some expressions that you might find useful in various situations when you animate.

Circular Path

[Radius * cos(360*Time),
Radius * sin(360*Time), 0]

where *Time* is one of the predefined time variables such as NT or S.

If you make the two Radius values unequal, you get an elliptical path.

If you specify a non-zero *Z* component, the path is no longer planar.

Following Another Object

[X, Y, Z] + Position

where *Position* is the Position controller of the second object.

The vector [X, Y, Z] can be an offset from the second object. (If it's [0,0,0], the two objects occupy the same position.) It can also be a vector expression that specifies some movement in itself, as in the last example of this chapter's Quick Start topic.

Keeping an Object Between Two Objects

(Position1 + Position2) / 2

where *Position1* and *Position2* are the Position controllers of two objects.

The divisor 2 constrains the object to be halfway between the two other objects. Other values constrain the object to other locations.

Bouncing Between Other Objects

(1+sin(360*Time))/2 * (Pos1-Pos2) + Pos2

where *Time* is one of the predefined time variables such as NT or S; *Pos1* and *Pos2* are the Position controllers of two other objects.

The subexpression (1+sin(360*Time))/2 is a value that oscillates between 0 and 1 over time. (Pos1-Pos2) is the vector between the two other objects. Multiplying the two and then adding Pos2 as an offset locates the object along this vector.

Changing the Number of an Object's Segments Based on Camera Distance

This expression varies the number of segments in a cylinder based on the distance of a camera. It is assigned to the cylinder's Segments creation parameter.

if ((length(Camera-Myself) > 35),
3 + (50*Height) / length(Camera-Myself),
MaxSegs)

where *Camera* is the position controller of the camera; *Myself* is the cylinder's position controller; *Height* (= 70) is the cylinder's height; *MaxSegs* (=100) is the maximum number of segments.

When the camera is closer, more segments make the cylinder smoother; when the camera is distant, the smoothing is less important and fewer segments render more quickly.

The if() function returns its second argument if the first argument is true; otherwise, it returns its third argument. In this example, if the camera is more than 35 units away from the cylinder, the expression calculates the number of segments; if the camera is 35 units away or closer, the number of segments is the MaxSegs constant.

The values in the second argument are chosen so that as the distance decreases toward the threshold of 35, the number of segments increases toward MaxSegs. The addition “3+” ensures that the cylinder always has at least three segments, even when the division rounds to zero (Segments is an integer).

Notice that to the expression, it doesn’t matter whether the camera is moving, or the cylinder, or both.

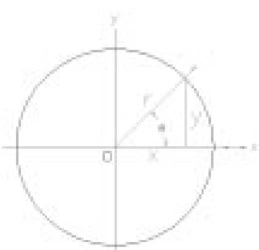
Trigonometric Functions

This topic is a quick review for readers who need a reminder about this area of mathematics. If you're familiar with trigonometry, you can skip this topic. If this topic is difficult to follow, you might consult a more basic reference on mathematics.

Trigonometric functions are principally used to model or describe:

- The relation between angles in a triangle (hence the name).
- Rotations about a circle, including locations given in polar coordinates.
- Cyclical or periodic values, such as sound waves.

The three basic trigonometric functions are derived from an angle rotating about a unit circle.



Trigonometric functions based on the unit circle

$$\sin\theta = \frac{y}{r} \quad \cos\theta = \frac{x}{r} \quad \tan\theta = \frac{y}{x}$$

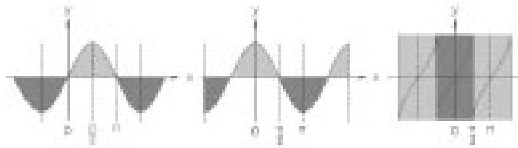
The tangent function is undefined for x = 0. Another way to define the tangent is:

$$\tan\theta = \frac{\sin\theta}{\cos\theta}$$

Because xyr defines a right-angled triangle, the relation between the sine and cosine is:

$$(\cos\theta)^2 + (\sin\theta)^2 = 1$$

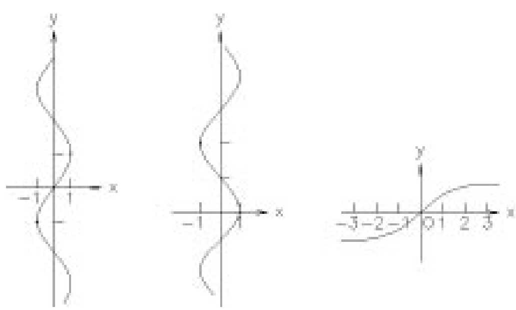
The graphs of the basic trigonometric functions illustrate their cyclical nature:



Graphs of the basic trigonometric functions

The sine and cosine functions yield the same values, but the phase differs along the *X*-axis by $\pi/2$ —in other words, 90 degrees.

The inverse functions for the trigonometric functions are the arc functions—the inverse only applies to values of x restricted by $-\pi/2 \le x \le \pi/2$. The graphs for these functions appear like the basic trigonometric function graphs, but turned on their sides.



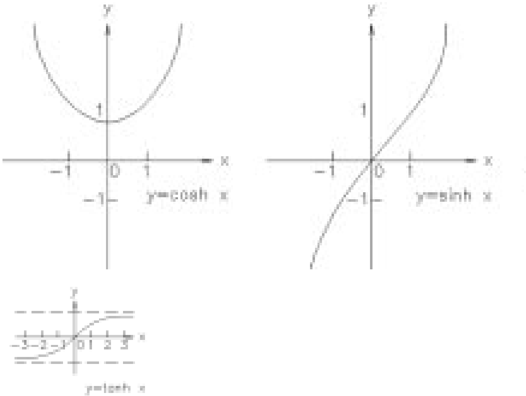
Graphs of the basic arc functions

The hyperbolic functions are based on the exponential constant *e* instead of on circular measurement. However, they behave similarly to the trigonometric functions and are named for them. The basic hyperbolic functions are:

$$\sinh x = \frac{e^x + e^{-x}}{2}$$

$$\cosh x = \frac{e^x - e^{-x}}{2}$$

$$\tanh x = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{\sinh x}{\cosh x}$$



Graphs of the basic hyperbolic functions

Vectors

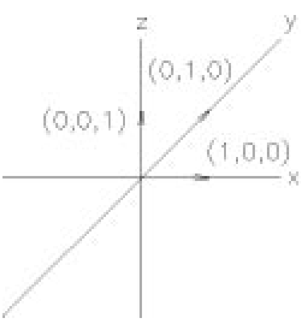
This topic is a quick review for readers who need a reminder about vector arithmetic. If you're familiar with vectors and vector calculations, you can skip this topic. If this topic is difficult to follow, you might consult a more basic reference on mathematics.

A vector expresses a length and a direction in a particular space. The vector is expressed as a point; for example, [5, 5, 7]. The length is the distance from the origin to that point, and the direction is similarly from the origin to (and through) the point.

In 3D Studio MAX, vectors have three values and describe positions in three-dimensional space. They can also represent percent scaling in *X*, *Y*, and *Z*; and—more abstractly—describe locations in RGB color space.

Unit Vectors and Basic Vectors

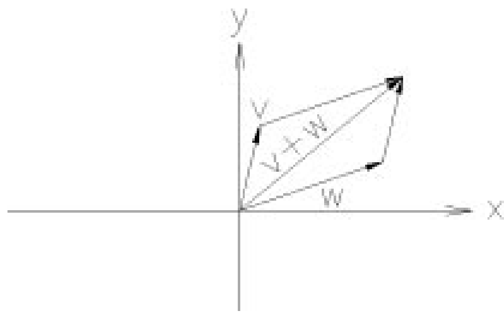
A unit vector has a length of one. Unit vectors are often used to express direction only. The three basic vectors are unit vectors that describe the three axes (*X*, *Y*, and *Z*) of 3D space.



Basic vectors and the *XYZ* axes

Adding and Subtracting Vectors

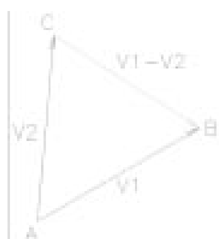
Adding two vectors creates a new vector that combines the length and direction of the original two. Vector addition is commutative: V+W = W+V.



Adding two vectors

$$V + W = [v_1 + w_1, v_2 + w_2, v_3 + w_3]$$

Subtracting two vectors gives the vector between the two points.



Subtracting two vectors

$$V - W = [v_1 - w_1, v_2 - w_2, v_3 - w_3]$$

Scalar Multiplication and Division

Multiplying a vector by a scalar changes the vector's length, as does dividing the vector by a scalar.

Scalar multiplication

$$yV = Vy = [yv_1, yv_2, yv_3]$$

$$V/y = [v_1/y, v_2/y, v_3/y]$$

Vector Length and Direction

The length of a vector is obtained from the Pythagorean theorem.

$$|V| = \sqrt{v_1^2 + v_2^2 + v_3^2}$$

In 3D Studio MAX expressions, the length() function returns this value.

The direction of the vector is the vector divided by its length—this gives you a unit vector with the same direction

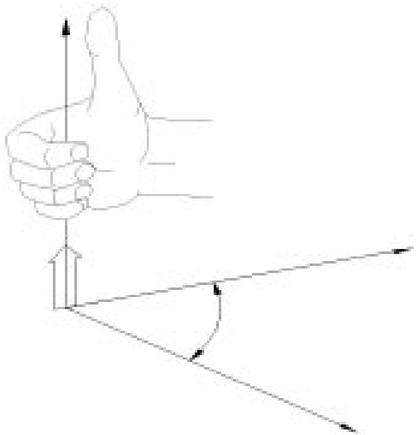
$$\frac{V}{|V|}$$

The distance between two points is the length of the vector between them.

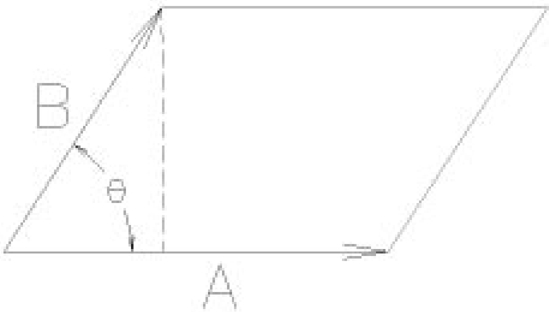
$$|VW| = |W - V|$$

Subtracting vectors to obtain a distance

Vector Multiplication



Finding a right-handed normal



Cross-product length is the area of the parallelogram between the vectors

Vector calculations include two operations that don't correspond to scalar arithmetic. These are the two kinds of multiplicative vector products: the dot product and the cross product.

Dot Product

Two vectors define an angle. This angle enables us to calculate the dot product of two vectors. (The dot product is named for the conventional notation: $V \cdot W$.)

$$V \cdot W = |V||W|\cos\theta$$

where θ is the smallest angle between the two vectors.

The dot product result is a scalar value. It is positive (>0) when the angle θ is acute, negative (<0) when θ is obtuse, and zero (0) when θ is 90 degrees, in which case the vectors are described as being perpendicular or orthogonal to each other. Dot product multiplication is commutative. It is associative and distributive as well.

Cross Product

Unless they are colinear, two vectors also define a plane. The plane can be given a normal based on the right-hand rule.

This unit normal is used in calculating the cross product of the vectors. (The name, like dot product, is based on the conventional notation: $V \times W$. This is also known as the vector product of two vectors.)

$$V \times W = U|V||W|\sin\theta$$

where θ is the smallest angle between the vectors, as for the dot product, and *U* is the unit normal of the VW plane.

The cross-product result is a vector. It is perpendicular to the VW plane in the same direction as the unit normal. If one or both of the vectors has a length of 0, or if the two vectors are parallel, the length of the cross product is zero. In other words, the length of the cross product approaches 0 as θ approaches either 0 or 180 degrees.

The cross product is not commutative — reversing the order ($W \times V$) results in a vector of the same length but the opposite direction. That is:

$$W \times V = -(V \times W)$$

The cross product is not generally associative, either. However, it is distributive for both scalars and vectors:

$$(yV) \times (zW) = (yz)V \times W$$

$$V \times (W + X) = V \times W + V \times X$$

and

$$(W + X) \times V = W \times V + X \times V$$

The length of the cross product — $|V||W|\sin\theta$ — is the area of a parallelogram between the two vectors.

Setup and System Information

The topics in this section present information related to system setup and configuration. These topics are generally useful after you've installed 3D Studio MAX.

- For initial installation instructions, see the 3DS MAX package.

Starting 3DS MAX

In routine use, you double-click the 3DS MAX icon to start the program. Should any error messages during startup, see explanations below.

Also see the topic, "Starting 3DS MAX from the Command Line," for specific options available at startup.

Possible Error Messages at Startup

Hardware lock drivers are not installed—Make sure you have administrative privileges. Reinstall the Sentinel component using the Custom option, then restart system.

No hardware lock found—Do nothing to the hardware lock until you have powered down your system. Then make sure hardware lock is firmly seated in parallel port. This message also appears if hardware lock drivers are not found.

TCP/IP Error—The network protocol TCP/IP is not configured properly. Refer to the later topic, "Running TCP/IP with 3DS MAX," for more details.

Authorizing 3DS MAX

Each time you start 3DS MAX after initial installation, you see an authorization message for the next 30 days, or until you authorize your copy of the program. See information in the 3DS MAX package for details on completing authorization.

Using Minimum Resolution

By default, 3D Studio MAX uses a long toolbar designed for higher resolutions than the minimum (800 x 600). If you are using minimum resolution—for example, on a laptop—some buttons in the long toolbar won't be visible. You can easily slide the toolbar to see buttons at either end. This also works for the toolbar in Track View.

To slide the toolbar:

1. Position the cursor over a gray area in the toolbar. The cursor changes to a hand.
2. Drag the hand left or right. The toolbar slides in that direction to reveal buttons that are off the screen.

Using the Short Toolbar

You can also switch to a *short* toolbar that fits a minimum-resolution screen. With fewer buttons, the short toobar does not always match the interface presented in this guide and in online reference.

To use the short toolbar:

1. From the menu bar in 3DS MAX, choose File > Preferences to display the Preference Settings dialog.
2. The General panel is the default for this dialog. Click the General tab if this panel is not visible.
3. In the UI Display area of the General panel, set Short Toolbar, then click OK. The dialog disappears, and the toolbar changes to the short version.

To return to the long toolbar, repeat these steps and clear Short Toolbar.

Returning to System Defaults

At some point you might want to return to the default settings of 3D Studio MAX—those settings that were in effect when you started 3DS MAX for the first time.

In the following procedure, you rename the *3dsmax.ini* file before creating a new one. Renaming gives you these options:

- You can compare the two versions, and where appropriate copy sections of the original *3dsmax.ini* file into the new file.
- You can replace the new file with the original by renaming the original *3dsmax.ini*.

Caution: If you delete the *3dsmax.ini* file, any custom settings stored in it are lost.

To return to system defaults:

1. Shut down 3DS MAX if it's running.

2. In the 3DS MAX root directory, rename the *3dsmax.ini* file—for example, *3dsmax1.ini*.

3. Restart 3DS MAX.

Choose a display driver from those available. Software Z-Buffer is the default. A new *3dsmax.ini* file is created at startup.

Installing the Hardware Lock

A *hardware lock* is a pass-through connector that you attach to a parallel port on your computer. The lock, from Rainbow Technologies, has a male connector at one end, a female at the other.

3D Studio MAX ships with one hardware lock for new packages, and another hardware lock for upgrade packages. They are *not* interchangeable.

- For new packages, install the new hardware lock on a parallel port as directed.
- For upgrade packages, install the new hardware lock by connecting it directly to the 3D Studio MAX Release 1.x hardware lock. The new upgrade lock will not work by itself.

You cannot run the 3DS MAX software until the right hardware lock is properly installed. This takes just a few minutes, and only needs to be done once. The driver for this lock is loaded during a typical setup and registered with the Windows NT Registry. It needs no configuration.

Compatibility Issues

The hardware locks for this release should not conflict with other locks you have on your system, such as those for AutoCAD R13 or R14.

The installation program automatically detects another version of the lock driver and only replaces it if the existing driver is older than the one shipping with 3DS MAX. Changing to a newer lock driver should not affect the performance of either lock.

Peripherals

In some cases, printers and other peripherals using the parallel port might be affected by the 3DS MAX lock. See the 3DS MAX Readme for any known limitations.

- When possible, connect the 3DS MAX lock from a new package directly to the port, then other peripheral locks in sequence.

Problems can sometimes arise when the 3DS MAX lock is between a peripheral and its lock. The peripheral might try to update its lock and affect the 3DS MAX lock instead.

In some cases, for 3DS MAX to work with its lock, peripherals attached through the lock port might need to be turned on and put online before using 3DS MAX.

Installing the Hardware Lock

Warning: Damage can occur to your computer if you try to install or remove the hardware lock while power is on. Always power down your system before doing anything with the hardware lock or connected devices.

To install the hardware lock:

1. Turn off your computer and all peripherals attached to it.
2. Connect the end of the hardware lock marked "COMPUTER" (the male end) to one of the following, depending on whether you have a new or upgrade package:
   - For new locks, connect to an open parallel port on your computer. If all parallel ports are taken, disconnect one of the peripheral devices and connect the hardware lock to the open port. Then reconnect the peripheral to the female end of the hardware lock.
   - For upgrade locks, connect to the Release 1 lock.
3. Tighten the thumb screws to prevent disconnection.

This completes hardware lock installation. 3DS MAX checks the hardware lock each time it starts up. If there's a problem with the lock installation, an error message appears the first time you start the program.

Maintaining the Hardware Lock

Once installed, the hardware lock should be left in place as long as you have 3DS MAX on your system.

If the hardware lock is disconnected while using 3DS MAX:

1. Save your work immediately.
2. Turn off your computer and peripherals.
3. Reattach the hardware lock securely before restarting your computer.

Insurance Advised

One hardware lock is supplied with each new or upgrade package of 3D Studio MAX. To replace a hardware lock that is lost, stolen, or destroyed, you need to purchase another copy of 3DS MAX. You should therefore insure the lock for its replacement cost—the price of a new software package.

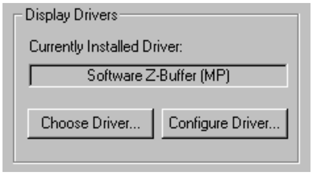
Should your hardware lock be damaged or fail to operate, contact your 3DS MAX dealer. You must return the original hardware lock to get a replacement.

Customizing Lock Ports

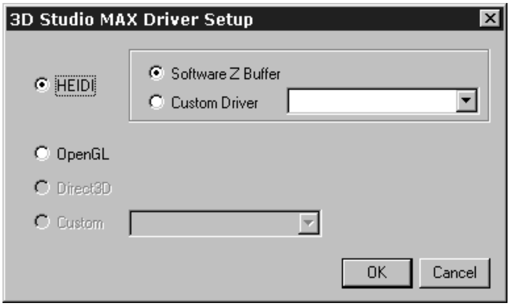
Usually any parallel port available under Windows NT or Windows 95 will work with the hardware lock. In some special cases, you might need to specify a port for this purpose. This, and other information about using the Sentinel hardware lock, is contained in the Sentinel Help file.

The file is *sentinel.hlp*, installed during setup in the Windows NT System32 directory. The default on many systems is *\winnt\system32*.

Changing the Display Driver



Display Area, Viewports panel



Driver Setup dialog

You can change display driver options from within 3D Studio MAX. For example, you might want to use the display driver supplied by the manufacturer of your graphics card, or upgrade to a new version of the driver.

To access options for display drivers:

1. From the menu bar, choose Files > Preferences to display the Preference Settings dialog.
2. Click the Viewports tab to display the Viewports panel. Options appear in the Display Drivers area.
3. Click Choose Driver to display the Driver Setup dialog. This is the same dialog that appears when you start 3DS MAX for the first time.

   When you choose a new driver and click OK, an alert box tells you that the display driver changes will take effect the next time you start 3DS MAX.

To install another display driver:

1. Copy the display driver to the Drivers subdirectory under the 3DS MAX root directory.
2. In the Driver Setup dialog, click Custom Driver and select a driver from the dropdown list.
3. Close 3DS MAX and restart it. The new driver is now active.

Making the Choice of Drivers

On the Driver Setup dialog, under Heidi, there are two options:

Software Z-Buffer—Uses the Heidi Z-buffer driver. This is the default.

Custom—One option uses the Heidi GLiNT driver. Do *not* choose this option *unless* you have a GLiNT graphics card installed in your machine.

There are custom Heidi drivers written for non-GLiNT cards. If your graphics card is supplied with a Heidi 5 driver, install it as a Heidi Custom option. Heidi 5 drivers are significantly faster than Open GL.

Other options:

Open GL—Uses a Microsoft driver or a driver from a hardware manufacturer designed to work with hardware accelerators. These are typically multi-threaded drivers that take advantage of dual processors.

The other options, Direct3D and Custom, are grayed out unless you have appropriate hardware and drivers installed.

Basic Guidelines

These are general guidelines in deciding what driver to use. See the 3DS MAX Readme for any additional information about display drivers.

- Choose the Heidi Software Z-Buffer option if you're using software rather than hardware acceleration. Unless you have special hardware, this is probably the best choice.
- Choose Open GL only if your graphics card supports hardware acceleration and Open GL.

Changing Drivers at Startup

As an alternative to using File > Preferences to change display drivers, you can access Driver Setup as 3DS MAX starts, eliminating a reboot following the change.

To change drivers from the command line:

1. Open a Command Prompt window.
2. In the 3DS MAX root directory, enter:
   3dsmax –h

   The Driver Setup dialog appears.
3. Choose an option. Software Z-Buffer returns to the default driver. Your choice takes effect as 3DS MAX starts up.

See the later topic, "Starting 3DS MAX from the Command Line," for other startup options.

Starting 3DS MAX from the Command Line

You can start 3D Studio MAX from the command line in a Command Prompt window, or include the command line in a batch file. There are a number of switches that you can use on the command line.

To start 3DS MAX from the command line:

1. Open a Command Prompt window.

2. Change directory to the 3DS MAX root directory, if this directory is not in your PATH.

3. Enter **3dsmax** to start the program. Optionally add a command-line switch, discussed below.

Command-line Switches

You can use the following switches after **3dsmax** on the command line.

Switch	Effect
–d	Turns off double buffering in Track View (not recommended).
–h	Allows choice of graphics drivers: Heidi, Open GL, Direct3D, and Custom.
–i otherfile	Starts program using *otherfile.ini* instead of *3dsmax.ini.*
–l	Loads the last *.max* file automatically.
–n	Disables network mode.
–s	Starts program in server mode.
–v	Loads a display driver.
–z	Writes version number to file.
anyscene	Starts program with a file called *anyscene.max.*

Examples:

3dsmax –l

3dsmax –i otherfile

3dsmax anyscene

Using –V Option

You use the –V option to load a different display driver at startup. This option overrides the setting in *3dsmax.ini*.

Follow the –V with one of these letters for the driver you want to use:

S—Loads Software Z-Buffer.

O—Loads Open GL.

D—Loads Direct3D.

N—Loads Null driver.

For example, if you ordinarily run Open GL and want to switch to Software Z-Buffer for a session, you would enter: **3dsmax –vs**

Using –Z Option

If you call Product Support, the representative might ask you to run this option to determine the exact version of your software. This option only writes to a file—it does *not* start 3DS MAX.

Follow the –Z with a file name:
3dsmax –z id.txt

The file (in this example, *id.txt*) is written to the 3DS MAX root directory.

Connecting to External Devices from 3DS MAX

A device in 3D Studio MAX is a plug-in for communicating with an external image-management system. Such external systems can be either other programs or peripheral hardware accessible from your workstation.

You access a device plug-in any time you click the Devices button in a dialog used to specify an image file. Depending on the device, you can specify images for reading, such as for backgrounds or maps, or for writing, such as Rendering or Video Post output.

3D Studio MAX includes a device plug-in for controlling the Accom Work Station Disk (WSD™) Recorder. This is a hardware device for viewing, storing, and transferring images between your computer and other video devices. See the Accom WSD Device plug-in help file for details about setting up a connection from 3DS MAX.

Device Installation

The following description is generic to setting up and connecting to an external device from 3DS MAX. Details depend on the type of device and how it has been installed in your system.

- Install the device in your system. Follow the instructions of the device manufacturer for installing and configuring the device to work with your system.
- Load the plug-in software that allows 3DS MAX to communicate with the device. Check the device manufacturer's instructions for this procedure.
- Establish a connection with the device by clicking Devices in any dialog used to specify image input or output. A device selection dialog appears where you choose a device and proceed to set up the connection.
- Choose a device from the Device Selection list and click Setup.
- Each device has its own setup dialog and procedures. Follow the instructions of the device manufacturer for completing the setup.

For example, click Devices in the View File dialog to select a device and set up the connection. Once you complete the connection from any dialog, it is established for all device dialogs in 3DS MAX.

Network Considerations

Errors usually occur during device setup because 3DS MAX and the device are not yet communicating correctly. Once you provide the correct setup information, the error messages no longer appear.

Getting the correct setup information, and correctly completing the connection, often requires close cooperation between you and your network administrator. Gathering information about your network and peripherals before attempting to set up devices in 3DS MAX will increase the likelihood of success.

Locating Installed Files

Almost all the files copied to hard disk during setup are located in the 3DS MAX root directory (default *\3dsmax2*) or a subdirectory. The default paths—used by 3DS MAX to locate scenes, images, maps, meshes, and so on—point to these subdirectories.

See the next topic, "Configuring File Paths," for details on changing these defaults.

Other Installed Files

A few files work directly with Windows NT. These are copied into two Windows directories.

Windows System32 directory—The default on many systems is *\winnt\system32*. The following files are copied to this directory if the file does not exist there, or if an existing version is older.

aasc32.dll *flccodec32.dll* *flcfile32.dll*	These files play animations (*.avi*) created in Animator Studio or other programs using the Animator Studio codec, and animations in FLC, FLI, or CEL format. No configuration is required. Playback is automatic. For example, if you choose File/View File and locate a *.flc* file, clicking OK brings up the image and a media player.
rnbovdd.dll *snti386.dll* *sentinel.hlp*	These are Rainbow Sentinel hardware lock files and a help file.
msvcrt40.dll	This file is shared with other Windows programs compiled with Microsoft Visual C++ 4.0. *Do not delete this file.*

Windows System32 Drivers directory—The default on many systems is *\winnt\system32\drivers*. The following file is copied to this directory if the file does not exist there, or if an existing version is older.

sentinel.sys	This is a Rainbow Sentinel hardware lock file.

Caution: If you uninstall 3DS MAX, you should not delete any of these files. All might be used by other programs.

Configuring File Paths

The default locations that 3D Studio MAX searches for all file types are specified in the Configure Paths dialog. You can choose to open and save files in any path location.

To access the Configure Paths dialog:

- Choose File > Configure Paths.

The Configure Paths dialog contains three panels for the basic categories of 3DS MAX support files: General, Plug-ins, and Bitmaps. See chapter 1, "Using 3D Studio MAX," for more information on working with files.

Running TCP/IP with 3DS MAX

When 3D Studio MAX starts up, it checks to see if TCP/IP is installed. This is a common network protocol. If TCP/IP is set up incorrectly, 3DS MAX reports the error to you.

Here's what you should do:

- If your machine is on a network, report the error to your system administrator. A TCP/IP error can affect your entire system and should be corrected.
- If your machine is not on a network, you might still have TCP/IP installed. Since you don't need it, go to Control Panel/Networks and remove the listing for "TCP/IP Protocol" if it is there.

As long as TCP/IP is installed on your system, 3DS MAX will detect any errors in its setup. This includes the case where you blank all the entries in TCP/IP configuration. This is considered an error.

The general rules to avoid TCP/IP errors are:

- If you don't need TCP/IP, remove it completely—don't just reset it to void values or otherwise disable it.
- If you're on a network, DO NOT REMOVE TCP/IP without first checking with your system administrator.
- If you really need TCP/IP, make sure it is configured properly.

Uninstalling 3DS MAX

Like other Windows NT programs, 3D Studio MAX ships with an Uninstall program that removes most of its files from your hard disk—if the original setup had administrative privileges.

Uninstall follows these rules:

- Only files that were copied to hard disk during setup are deleted. Any file created or changed following setup is unaffected. This includes *.ini* files and *3dsmax.exe*.
- Subdirectories are deleted only if they are completely empty after files are deleted.
- If 3DS MAX is running, many files might be in use and therefore not available for deletion.
- Shared files are not deleted. You are prompted for uninstalling shared files.

Uninstall depends on a file created during setup. Without administrative privileges, this file is not created. The file, located in the 3DS MAX root directory, is *DeIsL1.isu*.

- If this file is deleted or moved, Uninstall won't work.

In general, Uninstall looks for files and directories in their location at setup. If you rename directories, move them, or remap them using Files/Configure Path, Uninstall won't be able to locate files for deletion.

To uninstall 3DS MAX:

1. Close 3DS MAX before uninstalling.

2. If you installed the network services for rendering manager and server, you need to stop these services before removing them:

   - In the Control Panel, double-click Services, then select each service and click Stop.

   You then run the following commands from a Command Prompt window:

   - **manager –r**
   - **server –r**

   This removes the services from the Windows NT services list before the uninstall procedure deletes the executable files.

3. In the program group, double-click the Uninstall 3DS MAX icon.

   A Confirm File Deletion dialog appears, asking if you want to remove the application and all of its components.

4. Click Yes.

   Uninstall deletes as many files as it can.

5. After Uninstall has finished, check directories and move or delete remaining files and subdirectories manually.

Caution: Files in Windows System and System32 directories should not be manually deleted. See the earlier topic, "Locating Installed Files."