

STRATA™ VISION^{3d}

*The All-In-One
3-D Solution*

FIRST PRINTING

June 1997

©1989-1997 Strata Incorporated. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means—electronic, mechanical or otherwise—without the prior written approval of Strata Inc.

ACKNOWLEDGMENTS:

Strata Inc. acknowledges with gratitude the contributions of its development team, which produced the software described in this document.

- **Software Design and Programming** by Gary Bringhurst, Jim Boyd, Jim Wadley, Brad Campbell, Bruce Cardwell, John Adams, Brian Hill, and Ken Bringhurst
- **User Manual** by Scott Summerill and Julie L. Applegate
Ammended by Tammy Brinkerhoff
- **Interface Design** by Ken Bringhurst and Kyle Maxwell

Strata Incorporated



2 West St. George Blvd. • Ancestor Square, Suite 2100
St. George, Utah 84770 • (801) 628-5218

Technical Services (801) 628-9751
FAX (801) 628-9756

Copyright Statement

©Copyright 1995 Strata Incorporated and its licensors. All rights reserved.

These manuals, as well as the software described in them, are furnished under license and may only be used or copied in accordance with the terms of such license. The contents of these manuals is furnished for informational use only and is subject to change without notice. It should not be construed as a commitment by Strata Incorporated. Strata Incorporated assumes no responsibility or liability for any errors or inaccuracies that may appear in this manual.

Except as permitted by license, no part of these manuals may be reproduced, stored in a retrieval system, or transmitted in any form or by any means — electronic, mechanical, or otherwise — without prior written permission of Strata Incorporated.

Please remember that the files provided for the various tutorials presented in the manuals may also be protected under copyright law. The unauthorized use of such artwork or images in your own projects could be a violation of the rights of the authors. Please be sure to obtain any permission required from such authors.

Strata, StudioPro, and Vision3d are trademarks of Strata Incorporated, which may be registered in some jurisdictions. Macintosh, Apple, Power Macintosh, QuickTime, QuickTime VR, Navigable Movie Player, and Apple QuickDraw 3D are registered trademarks of Apple Computer, Inc. Genesis Multiprocessor System is a registered trademark of Daystar Digital Inc.

Other brand names and products mentioned in this manual are trademarks of their respective holders.

MacApp ©1985-1995 Apple Computer, Inc. Apple Computer makes no warranties whatsoever, either expressed or implied, regarding this product, including warranties with respect to its merchantability or its fitness for any particular purpose. The MacApp software is a trademark of Apple Computer, Inc. and is proprietary to Apple Computer, Inc., licensed to Strata Incorporated for distribution only for use in combination with StudioPro.

For defense agencies, this product is subject to use, reproduction and disclosure restrictions as set forth in sub-paragraph (c)(1)(ii) of the Right in Technical Data and Computer Software clause at 252.227-7013.

For civilian agencies, this product is subject to use, reproduction and disclosure restrictions set forth in sub-paragraph (a) through (d) of the commercial Computer Software Restricted Rights clause at 52.227-19 and the limitations set forth in Strata Incorporated's standard commercial license agreement.

Unpublished rights reserved under copyright laws of the United States.



Congratulations and thank you for choosing Strata Vision3d™ (Version 5.0). You are now on the cutting edge of 3D modeling and rendering.

Vision3d breaks new ground in several areas: speed in both modeling and rendering; additional acceleration capabilities, the ability to model in real-time with textures applied; and the support of new 3D software and file formats technologies.

Strata is the first 3D software company to take advantage of Apple's™ QuickDraw™3D (QD3D) and has incorporated this technology into Vision3d.

QD3D produces interactive, real-time feedback with textures and lighting in the modeling environment.

Faster redraws and rendering times are also a strong point of this version. Vision3d is significantly faster than previous versions, and can become even faster through two new technologies: QD3D hardware acceleration boards and multiprocessing.

Vision3d can take advantage of Apple's QuickDraw 3D hardware accelerator board, which can substantially increase your speed and power. In addition, the accelerator card allows the modeling window to show transparency.

Vision3d also supports one of the most exciting developments in desktop computing: multiprocessing. Vision3d supports new multiprocessing architecture from Apple Computer and Daystar™. With MP support built into Vision3d, processor-intensive functions such as modeling and rendering are substantially faster.

An advantage of Vision3d QD3D support is the use of the 3D metafile (3DMF), a graphic storage format that incorporates geometry, textures and other model information. Models created in Vision3d and saved in 3DMF can be used in other applications that support 3DMF.

3DMF also lets users cut and paste 3-D objects with light source information, basic textures, and other data between 2-D and 3-D applications that support 3DMF. 3DMF files produced on Macintosh computers should also be valid on other platforms that support 3DMF.

Another great feature of Vision3d is the ability to drag-and-drop, and cut and paste 3D models. One of the first sources to allow 3D drag-and-drop is Apple's new Scrapbook, which is included with this product.



Vision3d can also be used to build and render panoramas and create object movies for use in Apple's new QuickTime™VR (Virtual Reality). In QuickTime VR, panoramas are transformed into walk-through, interactive movies in which you control the view of a 360-degree panorama.

Using Vision3d's QuickTime VR Panorama rendering option allows you to avoid multiple PICT renderings and having to use the QTVR Stitcher. Simply use your Vision3d panorama with the "Dicer" from Apple's QuickTime VR Authoring Tools Suite.

"Hot spots" in the panorama allow activation of a function or object. If a hot spot contains an object, the viewer could pick up and manipulate that object. Hot spots can also be used to launch the viewer into another scene, or even into another type of media.

To create an object, use Vision3d's special model, the QuickTime VR Object Camera. Then use Apple's Navigable Movie Player to turn it into a VR object.

Vision3d can also output in Virtual Reality Modeling Language (VRML). VRML files can then be used to create navigable, 360-degree 3D environments on an Internet Web page using any browser that supports VRML.

In Vision3d, you will, of course, be able to model objects and scenes and create animations using all of Vision3d's regular features. So have fun!



Introduction i

CHAPTER 1: MODELING IN VISION3d 5.0

Introduction 1.1

Working with QuickDraw 3D 1.1

 Rendering Settings 1.2

 Modeling Window Settings 1.2

 Low memory warning 1.3

3DMF 1.4

 Importing, exporting in 3DMF 1.4

Other new features 1.6

 Rendering options 1.6

 Copy and paste, drag-and-drop 1.6

CHAPTER 2: QTVR AND VRML

Introduction 2.1

QTVR 2.1

 How to create a VR panorama 2.1

 How to create a VR object movie 2.3

World Wide Web 2.8

 VRML 2.8

 URL addresses 2.9

CHAPTER 3: METABALLS

Introduction 3.1

Using Metaballs 3.2

Using the interface 3.2

Undoing a Metaballs operation 3.4

Showing a screen approximation of the Metaballs entity 3.5

Understanding the Expert options 3.6

 Ball weight 3.6

 Surface focus 3.7

 Radius scale 3.9



Rendering the image	3.10
Adding textures	3.10

CHAPTER 4: BOOLEAN MODELER

Introduction	4.1
Boolean Operations	4.1
Cutting	4.3
Union	4.3
Subtraction	4.7
Intersection	4.12

CHAPTER 5: MORPH

Introduction	5.1
Common items	5.1
Morphing multiple objects	5.1
Working with grouped objects	5.1
Caveats	5.2
Particle Morph	5.5
Description	5.6
Using Particle Morph	5.8
True Morph extension	5.9
Description	5.10
Using True Morph	5.11

CHAPTER 6: PATHEXTRUDE

Introduction	6.1
Using PathExtrude	6.1
Creating a separate path	6.2
Creating a manually scripted path	6.5

CHAPTER 7: WARP

Introduction	7.1
Using Warp	7.1
Wrap degrees	7.2
Pivot angle	7.7
Twist degrees	7.9

TOC2



TABLE OF CONTENTS

Using the Expert dialog 7.11
 Bend on wrap 7.11
 Arc on wrap 7.11
 Base Circumference 7.12
 Final Circumference 7.12
Animating a Warp operation 7.13
Adding textures 7.14

CHAPTER 8: ENHANCEMENTS

Introduction 8.1
Alternate Resources Files 8.1
Focal length resynchronizing 8.4
Skin modeler additions 8.5
Map optimization 8.6
Phong with Shadows 8.7
Raydiosity 8.8
DXF file handling 8.9
Alpha channel enhancement 8.10



CHAPTER 1: MODELING IN VISION3d 5.0

Introduction

In this chapter, you will discover what it's like to model with textures and light sources in real time. Strata Vision3d 5.0, using Apple's™ new QuickDraw™ 3D, is a lot of fun.

Modeling with applied textures and lights is a trade-off between memory and appearance. But don't worry - Vision3d gives you all the flexibility you need to adjust quickly to your specific needs.

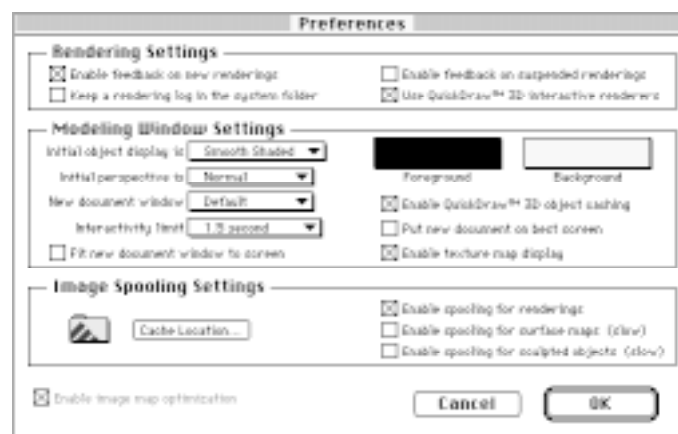
In the early stages of your project, you may not need all of the real-time texture options available in this application. At this point, you may prefer speed. Simply turn off QuickDraw 3D, and model with Vision3d's other features.

In addition to the optional use of QuickDraw 3D, you have a range of display methods and spooling options to choose from.

To install Apple™'s QuickDraw 3D, and for further instructions and system requirements, please see the Read Me included for this release.

Working with QuickDraw 3D

With Vision3d 5.0, you have several new Preference settings dealing with QuickDraw 3D, object display, texture map display, image spooling, and bounding boxes.





Rendering Settings

In Rendering Settings, there is a checkbox titled `Use QuickDraw™ 3D interactive renderers.' Enabling this option will allow you to model with textures and lights in real time.

Disabling this option will gray out three options: `Enable QuickDraw 3D object caching,' `Enable texture map display,' and `Interactivity limit.' See below.

Modeling Window Settings

In the Modeling Window Settings section of the preferences dialog, you will find four new options:

Initial object display - This pop-up menu offers three choices of on-screen display methods: Wireframe, Quick Shaded, and Smooth Shaded. These display methods may also be changed in the modeling window as your needs change.

Modeling in the Smooth Shaded display lets you take advantage of all of Vision3d's modeling features, but takes longer to redraw. To speed up redraws but retain some color and lighting information in the modeling window, use Quick Shaded. For the fastest redraws, use the Wireframe option.

In order to see textures while modeling, you must select Smooth Shaded, have Enable Texture Map Display enabled, and have QuickDraw 3D installed and enabled.

Note: If QuickDraw 3D is disabled, your available viewing choices revert to Wireframe, Hidden Line and Quick Shaded. (See the User Manual).

Interactivity limit - This pop-up menu allows you to set the speed at which a selected object reverts to simple outline (bounding box) while in motion.

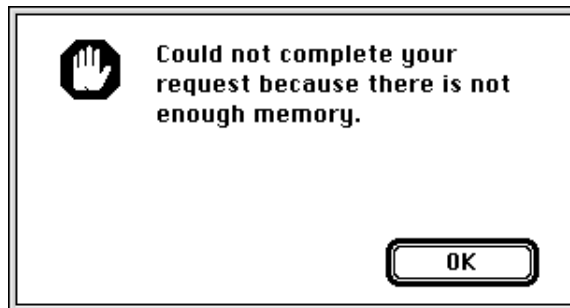
Enable QuickDraw 3D object caching - With this option enabled, on-screen redraws are faster, but more memory is used. To conserve memory, disable this option.

Enable texture map display - This option allows you to see textures and lights as you model (provided QuickDraw 3D is installed and enabled). Disabling this option saves memory.

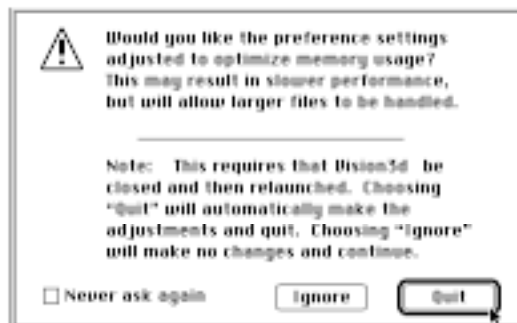


Low memory warning

Because of the high memory requirements for modeling in QuickDraw 3D with textures and lights, you may find yourself getting low on memory. If this happens, this warning will appear:



Click OK. This dialog will appear:



Choosing Quit will make adjustments in your preferences which will cut down on memory use: QuickDraw 3D will be turned off, Enable image map optimization will be disabled, and all three Enable spooling options will be turned on.

This is the slowest way to run Vision3d, but will allow you to handle larger files.



3DMF

3DMF is Apple's new format for storing 3D graphic information. Because the QuickDraw 3D and 3D technology is so new, there are some limitations.

- All textures shown while modeling are approximations, not the actual textures. When you render, however, all textures will appear as mapped. To see the exact texture as mapped, take a small snapshot, and if necessary, use the Change Mapping dialog to adjust it. Changes in the texture mapping will not be reflected in the modeling window. For more information on texture mapping, see Chapter 5 in the User Manual.
- Procedural (solid) textures used while modeling will display only one of the colors that make up that texture. When rendered, however, they will appear as mapped.
- Bump maps are not displayed in the modeling window, but will appear correctly in any rendered image.
- Tiled textures will appear in the modeling window with one tile, stretched to cover the entire object. Texture tiling may also appear inaccurate in the modeling windows, and textures do not reflect view-specific texture applications. But all will appear correctly when rendered.

***Note:** Transparency will only be displayed in the modeling window if you have installed a QuickDraw 3D accelerator card.*

Importing, exporting in 3DMF

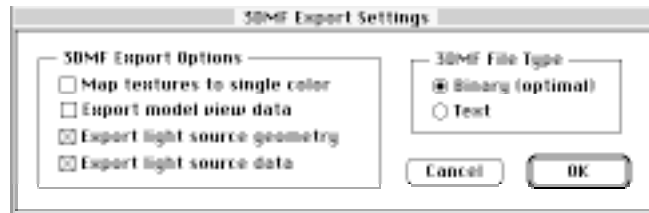
There are several ways to import models in the 3DMF format: drag-and-drop, cut and paste, copy and paste; and with the Open As or Import commands. Importing 3DMF works the same as any other file format. See the User Manual for more information.

When exporting a model, the same texture limitations apply as in the modeling window. 3DMF supports information from global, point and spot lights. It does not support gels.

***Note:** Importing and exporting models in 3DMF works with QuickDraw 3D turned off in your Preferences, as long as QuickDraw 3D has been installed.*



To Save as a 3DMF file, choose Save As from the File menu, then select 3DMF and click Save. The 3DMF Export Options dialog will appear.



3DMF EXPORT OPTIONS

- Map textures to a single color - Use this option if exporting a model to an application that does not accept color map information. The color will be an average of all the map colors.
- Export model view data - Use to save and export the active modeling view. Use the view management tools to get the exact angle you want.
- Export light source geometry - Use this option if you want to see and manipulate the light object in another application. This allows you to see where the original light source was, but will not change the lighting effect.
- Export light source data - Use to export the effect of the light source, but not the geometry of the light source itself.

Note: Choosing "Export light source geometry" alone will NOT give you the effect of the light.

3DMF FILE TYPE

- Use the Binary check box for all normal uses. Both Text and Binary files contain the same information, but store it differently.
- The Text option saves the model as text which can be read and edited in text applications. This is a written description of the model's geometry, texture, and light sources. The Text option will create a much larger file than the Binary option.



Other new features

Rendering options

Vision3d 5.0 offers some exciting new rendering options:

- First of all, the Raytracing and Raydiosity™ renderers are optimized for Vision3d. Rendering is now substantially faster.

***Note:** Raytracing and Raydiosity look different while rendering than earlier versions of Vision3d. This is because the feedback has been changed to scan-line to provide a clearer picture of rendering progress.*

- The new Smooth Shading option corresponds to Vision3d's new display methods (see Getting Started, above.)
- A QuickTime VR Panorama option has also been added. This is discussed in Chapter 2.

Copy and paste, drag-and-drop

You may easily copy, paste and drag-and-drop 3DMF files between Vision3d and any application that supports 3DMF. In addition, objects may be dragged in and out of the new Apple QuickDraw scrapbook that is included with this release.



CHAPTER 2: QTVR AND VRML

Introduction

Strata Vision3d 5.0 gives users two exciting new ways to generate images for virtual reality navigation.

Using Apple's QuickTime VR, a standard scene constructed in Vision3d can be rendered in a 360-degree panorama, which QTVR can then translate into an interactive view of the scene.

In addition, an object created in Vision3d can be rendered as a VR object, then viewed interactively from any angle.

Vision3d 5.0 can also output in Virtual Reality Modeling Language (VRML). VRML files can then be used to create navigable, 360-degree environments on a network Web page using a VRML browser.

Vision3d 5.0 also has the capability of preserving URL addresses, which can launch the user to a specified location on the Internet.

QTVR

How to create a VR panorama

Because rendering a QuickTime VR panorama simulates an actual camera, it should be rendered through a camera view. Use the Camera Object tool (the small camera) from the Tools palette to place one or more cameras in the modeling window.

1. Select the camera view you wish to use by choosing a camera from the Camera Windows submenu of the Windows pull-down menu.





You can also select a camera by double-clicking the camera you wish to use for the rendering view.

2. When the camera view is selected, a new window will appear showing the view from the selected camera. In this window, set the lens control to 15 millimeters.

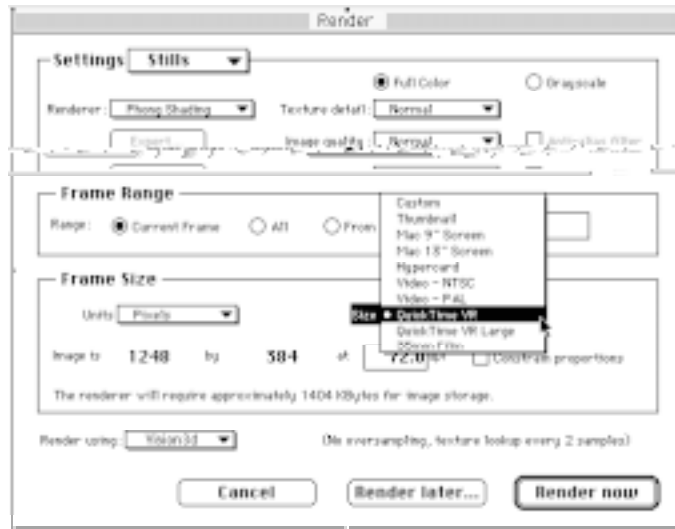
Note: While 15 millimeters is the ideal setting for eliminating distortion, the setting you choose will depend on your desired result.

3. Open the Render dialog box by selecting Render from the Rendering pull-down menu, or by using the Command-Y keystroke shortcut.



4. When the Render dialog box appears, select QuickTime VR Panorama from the Renderer pop-up menu in the Settings section of the dialog.
5. Then, in the Frame Size section of the Render dialog box, find the Size pop-up menu. Select QuickTime VR or QuickTime VR Large as your rendering size. Because the QuickTime VR Large option is four times the size (and resolution) of the QuickTime VR option, it will also take roughly four times as long to render.

2.2



When rendering is complete, your scene is ready for the QTVR Dicer tool, found in Apple's QuickTime VR Authoring Tools Suite. (This is NOT included with Vision3d 5.0.)

The Panoramic rendering gives a 360-degree by approximately 120-degree rendering that will appear somewhat distorted as a PICT file. Don't worry, that's normal.

How to create a VR object movie

Any 3D object can be rendered in Vision3d 5.0, creating a QuickTime Movie that can then be viewed interactively using Apple's QuickTime VR Navigable Movie Player. When completed, this process allows the viewer to manipulate the object in any direction, and view it from all angles.

The object movie is created by using a pre-scripted camera to capture views of the object from all directions. The renderer will take a picture of your object every 10 degrees, starting directly overhead - for a total of 684 frames.

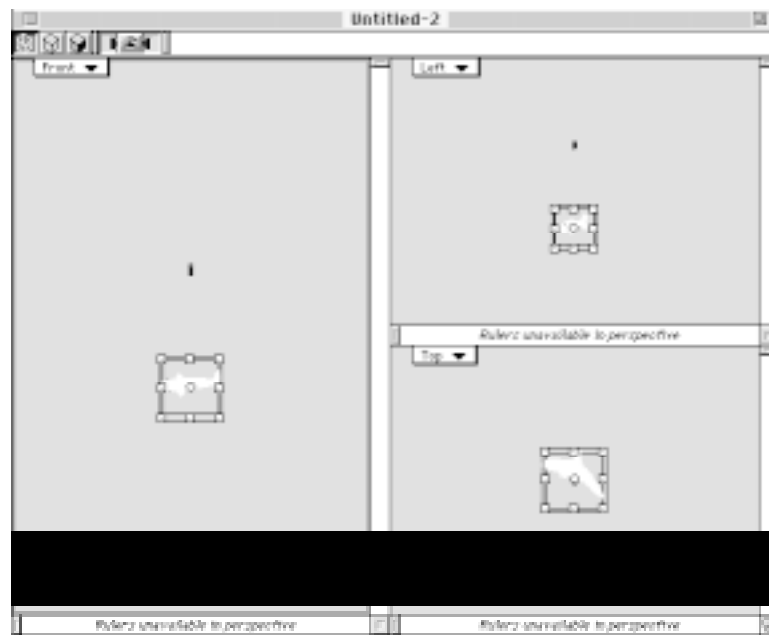


Rendering time will of course depend on your machine, the complexity of the model, and number of colors you specify. To reduce rendering time, keep complexity as low as is practical for your project.

You may create an object movie of more than one object, as long as all are within the cube. The objects will function as a group, not independently.

To create an object movie:

1. Open the Vision3d QuickTime VR Object Camera, located in your Stationary folder. This model automatically gives you three views to help you center your object. Notice the small built-in camera and the cube.



Place your object(s) inside the cube. There are several ways to accomplish this, including the Import command. For more information, refer to the Vision3d Users Manual.

2. The object must be completely within the cube for the rendering to work properly. To place the object within the cube, first scale it to the

2.4



appropriate size. (Don't forget to hold down the shift key to scale proportionately.)

To center your object within the cube, use the Align command:

- a. Use a marquee to select the object and the cube.
- b. Select Align from the Modeling menu.
- c. Enable the Center radio buttons in the Horizontal, Vertical and Depth options.
- d. Click OK.

Note: If you have difficulty grabbing your object once it is inside the cube, select and hide the cube using the Hide Selected command in the Select menu.

3. If you want light sources, place them now.
4. Hide the cube before you render! If you don't, you will end up with a lovely rendering of a solid gray cube.
5. Render the object movie using the QTVR Camera:
 - a. Select the QTVR Camera from the Camera Window option of the Windows pull-down menu, or double-click the camera.
 - b. From the Rendering dialog, select All in Frame Range, then click Render Now.
 - c. In the dialog box that appears, specify a file name and location, and select QuickTime Movie File, as shown in the following illustration. Click Save.





6. a. In the Compression Settings dialog box, choose the setting which best suits your needs.

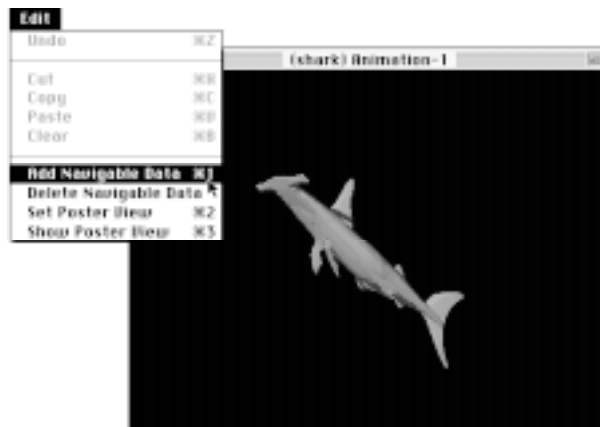


We recommend the Animation setting, which is the default. If you choose another setting, or use another compressor, make sure that every frame remains a key frame.

- b. Choose the lowest number of colors, and the lowest quality setting, that suit your needs. Click OK.

CONVERTING YOUR VISION3D OBJECT MOVIE

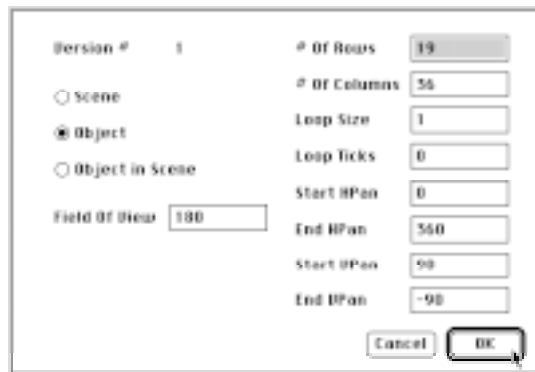
1. When the rendering is finished, save a backup copy.
2. Open Apple's Navigable Movie Player.
3. Now open the QuickTime Movie you just made from within the Player using the File menu.
4. Choose Add Navigable Data from the Edit menu.



2.6



5. In the dialog box which appears, change the specifications as shown in the following illustration:



Leave the other settings as they are, and click OK.

Note: Make sure these settings are correct. Entering incorrect values in the Navigable Movie Player may cause you to crash, and may corrupt the object movie.

Voila - here is your object in real-time 3D, with a grabber tool. Use the grabber tool to navigate around your object in any direction.



You may also manipulate your object movie by moving the hand tool to the edge of the viewing window. Your hand tool will change into a small arrow, either straight or curved.



The arrows allow you to run the movie continuously by holding down your mouse. On the upper and lower edges, a straight line may also be seen. This means the object will not rotate any further in that direction.

The new QTVR Movie can also be played in the QTVR Movie Player.

World Wide Web

VRML

Vision3d 5.0 is capable of outputting files in Virtual Reality Modeling Language (VRML). VRML is a language for describing virtual interactive worlds. VRML allows virtual worlds to be networked via Internet and hyperlinked with the World Wide Web using a VRML browser.

VRML files can be used to create navigable, 360-degree environments on a Web page. For example, you could create a room or scene as your home page on the Net. Viewers could navigate around the room. Any item in the room, such as a product package lying on a table, could be a hot spot. After clicking on the package, the user could be launched on a tour of your business.

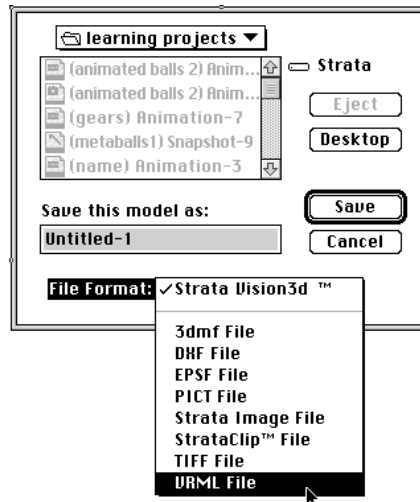
To use VRML, use the Save As command from the File menu, then choose VRML from the pop-up menu

Click Save. The VRML Export Settings dialog will be displayed. Use the check boxes to enable these choices:

- Export light source geometry - Use this option if you want to see and manipulate the light object in another application. This allows you to see where the original light source was, but will not change the lighting effect.
- Export light source data - Use to export the effect of the light source, but not the geometry of the light source itself.

Note: Choosing the Export light source geometry option alone will NOT give you the effect of the light.

Keep in mind that Vision3d 5.0 will NOT open a VRML file. If you think you may want to change something, be sure to save a copy of your model in Vision3d file format as well as in VRML.



URL addresses

With the ability to save in VRML, Blitz users can preserve custom information such as URL (Universal Resource Locator) addresses, which will link to a specified location within the Internet. Viewers could click an object which would act as a hot spot, launching viewers to another room or another location on the Web. Applescript addresses may also be used.



URL addresses are attached to surface textures by using the Link Address Settings dialog box. This is accessed through the Texture Editing dialog box.

With a textured object selected, click Edit in the Texture palette to open the Texture Editing dialog. This dialog can also be accessed by choosing Edit Selected from the Textures menu. In the Texture Editing dialog, click the Links button.



In your model, each texture will accept only one URL address. If you wish to attach two different addresses to models having the same texture, rename one of the textures using the Copy From option in the Texture Editing dialog, and attach the second address to the new texture.

URL addresses are not preserved with the texture unless you save the file in VRML.

Note: *Procedural textures, or volumetric textures such as Camouflage and Green Marble, will not accept a URL address. You must use a surface texture such as brick, firestorm or metal.*

2.10



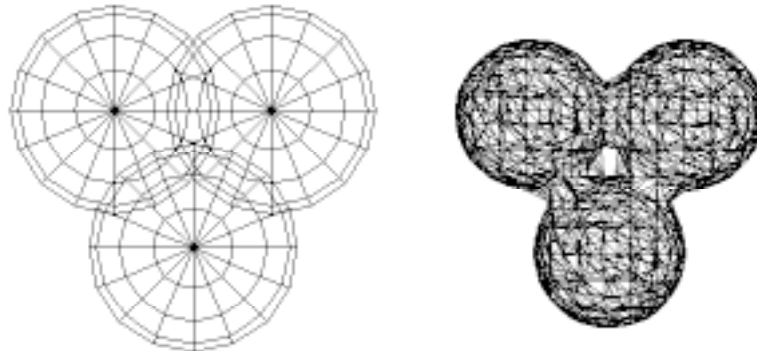
CHAPTER 3: METABALLS

Introduction

Your first glimpse of Strata's Metaballs extension can be as exciting as the first time you pedaled down the street on your two-wheeled bicycle without training wheels. The special effects created with this extension produce unique graphics that will amaze the casual observer as well as the Strata "power user."

The Metaballs modeler allows you to transform separate spherical shapes into a single procedural entity that seems to melt the spheres into each other as they come into close proximity. The effect is similar to what happens when drops of Mercury are brought closer together—each seems to "reach out" for the other, bridging the space separating them until they merge into a single drop.

*The wireframe image on the left is an example of what you would see on your screen with the **Show only spheres in modeling window** option selected in the Metaballs dialog. The right illustration is the resulting view with the **Approximate metaballs surface in modeling window** option selected.*



The extension allows the user to assign a "weight" classification to spheres included in the operation. The software then generates an image which fills in the space between the spheres based on the weight and focus settings in the Metaballs dialog. If the spheres are scripted prior to the Metaballs operation, that movement will be incorporated in the generated entity. For example, if the spheres are scripted to travel toward each other from a distance, pass through each other, then travel away from each other again, the rendered animation will show the spheres

3.1



traveling toward a shared point in space, globbing together as they get closer to that point until they seem to become a single sphere, and then pulling apart into individual shapes again as they go their separate ways.

The extension performs all the calculations internally once the dialog settings have been made, making the process simple for the user. However, understanding how each setting affects the final entity is essential when planning for the Metaballs effect. Therefore, this section will discuss each setting option and explain how they affect other settings.

Using Metaballs

The Metaballs extension files must be present in the Strata Vision3d Extensions folder prior to launching the program. If properly installed, the Metaballs extension can be accessed through the Join command of the Metaballs submenu in the Modeling menu or via the button bar. If the Metaballs extension does not appear, please refer to the installation instructions accompanying the software.

To use the extension you must select the shapes you wish to perform the operation on prior to accessing the dialog. All shapes selected must be spheres. The Metaballs operation will not work on shapes other than spheres. If the sphere is elliptic or oblong-shaped, it will be transformed into a perfectly round sphere during the operation. The actual size of any elliptical sphere included in the operation will be derived from the shortest radius.

You should note the Metaballs extension requires ungrouped shapes to perform the operation. Shape instances inserted into the work window must first be ungrouped to be eligible for inclusion in the Metaballs operation. For more information on shapes and shape instances please refer to the main User Manual.

If the Metaballs option is dimmed in the Modeling menu, you have not met one or more of the conditions required to perform the operation.

Using the interface

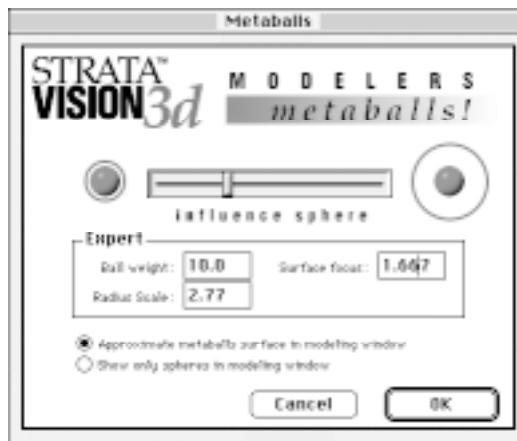
The Metaballs interface has been designed to accommodate all Strata users from beginner to expert. The influence sphere slider control gives you an easy method for setting up your model for the Metaballs



operation, and the Expert box allows you to “tamper” with the recommended settings to create your own special effects.

It is strongly recommended that you use the Metaballs dialog slider control whenever possible, especially if you are working on an important project. Experimenting with the Expert fields could produce unexpected results if you do not fully understand how the fields interact. Therefore, such experimentation should only be done when you have time to explore the modeler’s features.

When accessed, the Metaballs dialog will allow you to adjust the various options for the operation through a simple slider control or with keyboard entries in the Expert dialog box.



When accessed, the Metaballs dialog will default to a ball weight of 10.0. The Radius scale and Surface focus field will open in the setting established when the modeler was last used. Moving the influence sphere slider changes the Radius scale and Surface focus fields proportionately to keep the final metaballs entity the same size as the original spheres were before the operation.

The influence sphere slider controls the invisible sphere of influence surrounding each sphere in the operation. This sphere of influence is used to determine when and to what extent two balls will merge. For the merging to occur, the spheres of influence of two or more balls must overlap.

As you can see by the dialog’s graphic display, moving the influence sphere slider to the left inhibits the metaballs effect from occurring between spheres until they are in close proximity. Sliding the bar to the right extends the area of influence, allowing the metaballs deformation to begin while the spheres are some distance from each other.



Increasing the sphere of influence actually enlarges the original spheres by increasing the Radius scale. However, because the Surface focus field is simultaneously adjusted to maintain the model's proportions as they were drawn, the metaballs entity displayed on your screen and the rendered image will appear the same size as the original spheres in the model. This will be true whenever the slider control is used.

However, altering the fields manually through the Expert box could make the metaballs entity larger or smaller than the original model depending on the settings. This relationship will be fully explained later in discussions about the Expert box.

Undoing a Metaballs operation

You can undo any metaballs operation by choosing Undo from the Edit menu immediately after the operation is finished or by using the Unjoin command in the Metaballs submenu of the Modeling menu.

The Unjoin command can be used at any time on a metaballs entity to return it to its original state provided the link between the entity and the Metaballs modeler has not been broken.



The link between the modeler and the entity is established during the Metaballs operation. If the entity is further edited in another modeler, the link will be broken. However, you must approve the break through a warning prompt that will appear when you try to use the entity with a different modeler. More information on modeler-object links is available in the User Manual.

3.4

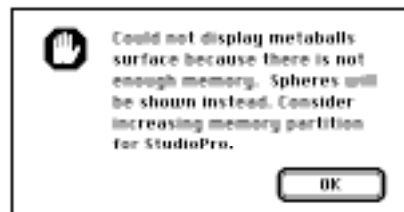


Showing a screen approximation of the Metaballs entity

You can direct the Metaballs modeler to display an approximation of the final entity on your screen by turning on the *Approximate metaballs surface in modeling window* radio button. When this option is turned on, the screen will redraw an approximation of the metaballs entity in the modeling window.

In cases of low memory, you will be warned there is not enough memory to display in this mode and the display will automatically switch to the *Show only spheres in modeling window* option.

You can free up memory by increasing the application's partition size on 68K machines or by turning off other applications on Power Macintosh machines. If all else fails, you should consider installing more RAM memory.



If the display was switched as described above, you must use the Edit Primitive command from the Edit menu to change the view to display the entity approximation after you have increased the available memory.

You can also change the Expert field settings in this dialog to refine your model without going through the process of undoing the operation and resetting the fields.



Deciding which display method you want to use depends largely on the amount of RAM memory you have and how fast you want the screen



to redraw. Obviously, displaying an approximation of the entity takes more memory and requires more time to redraw.

Understanding the Expert options

This portion of the chapter will discuss the Expert fields in the Metaballs modeler to give you a better understanding of how each affects spheres in the operation. This information will benefit you in your exploration of the Metaballs extension and help you achieve the results you want from the modeler.

The Expert options available in the Metaballs interface allow you specific control over the special effects produced by the extension. Changing the setting in any one field will have an impact on the total effect, so learning how each works is imperative for planning your project.

***HINT:** There are no limits to the number of spheres you may use in a single Metaballs operation, memory permitting. For simplicity, this discussion will not attempt to address every combination. However, the techniques described in this section will hold just as true for a model with 100 spheres as it would for a three-sphere operation.*

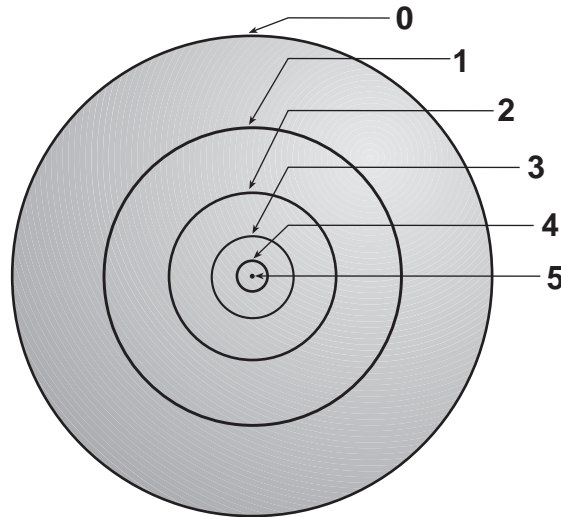
Ball weight

The Ball weight field is a global setting that affects each sphere in the operation. The Metaballs extension establishes a series of spherical shells layered throughout each sphere like the skin of an onion. Each shell is assigned a value based on the weight setting for that operation. For example, a sphere with a ball weight of 5.0 would be continuously segmented by the shells into sections, which in a two dimensional environment would resemble a target. The rings of the target become progressively denser toward the center with a diminishing distance ratio between each ring. The sphere's center point (the target's bullseye) would have a value of 5.0 and the surface of the outer shell would have a value of 0.0. The shell values are then used to establish the surface focus.

The diminishing distance ratio is what allows the modeler to calculate a smooth, curved deformation skin between spheres. If each ring were scaled using the same distance ratio, the deformation skin would appear linear and cumbersome. However, by using a diminishing ratio the entity displays a more graceful, almost fluid deformation skin.



This illustration represents a 2-D cross-sectional view of the previously described sphere. In the 3-D world, each of the target's rings would be rotated to form a spherical shell around a common center point and the entire surface of each shell would carry that segment's weight value.



These shell segments will not show up on your screen and you will not see a segmented object when the entity is redrawn or rendered. The segmentation is merely a method of performing the calculations for the Metaballs effect. However, the importance of this segmentation as it relates to the Surface focus field will become apparent during the discussions of the other Expert fields.

Surface focus

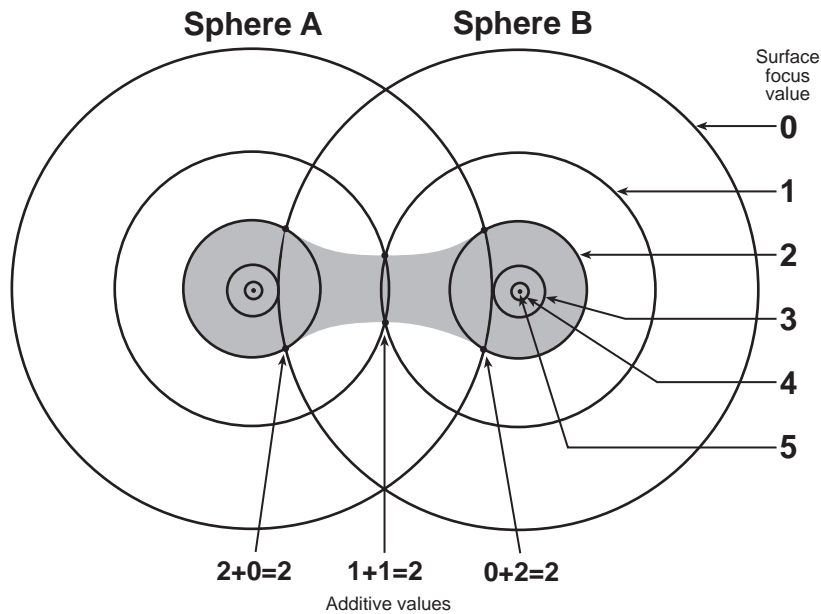
This field determines where on a sphere's surface it will meld with other spheres. As mentioned earlier, the Surface focus has a direct correlation to the Ball weight setting assigned to spheres in the model. The Surface focus can also be thought of as a density threshold. The higher the value, the denser the weight required to satisfy the threshold.

The Surface focus setting establishes each sphere's perimeter in the final metaballs entity and the threshold of deformation joining the spheres together. To establish the surface area of the deformation bridge—the shared area melding the spheres together—the extension uses an additive calculation where the spheres intersect.



Consider two equal-sized spheres (A and B) with weight settings of 5.0 overlapping such that the outside shell of one sphere shares a single, intersecting point with the other sphere's 3.0 shell segment and a Surface focus setting of 2.0. The melding structure's surface points will be established where sphere A's 2.0 shell segment intersects with sphere B's 0.0 segment (because the sum of the two segment values equals the Surface focus setting). Another ring of surface points will be set where each spheres' 1.0 segments intersect (because 1.0 plus 1.0 equals 2.0). A third ring of surface points occurs where sphere B's 2.0 segment intersects with sphere A's 0.0 segment. The meld will be constructed along these points and appear as a smooth, curved bridge between the spheres in the final entity.

This cross-sectional diagram demonstrates how the additive calculations of the Metaballs extension forms the bridge between two spheres as described in the previous discussion. The technique is used on all models, even if the spheres are different sizes or if the variables are fractional.



This same explanation applies to all spheres in the operation sharing intersecting points. The surface areas of two spheres, as determined by the Ball weight and Surface focus settings, must intersect at some point for the Metaballs melding process to occur. If the intersection is a single point, the melding will be minimal. As the circumference of the intersecting surfaces expands, the globbing effect will increase between the spheres.

3.8



HINT: The influence sphere slider bar changes the size of each ball in the operation to expand the sphere of influence. Therefore, while the display and rendered images of the Metaballs entity will appear to be the same size as in the original model, the balls will have actually been scaled up if the slider was moved to the right.

Spheres with a weight assignment equal to or less than the Surface focus may not be visible in the rendering because the focus will be constrained to the sphere's center point (if the weight and focus are the same), or to a nonexistent segment in that sphere (if the weight is less than the focus setting).

You should note that setting the Surface focus to a value close to that of the Ball weight could make it impossible for the application to accurately redraw the entity on your screen. For example, if the Surface focus is set to 9.0 and the Ball weight is 10.0, it may be difficult for the screen redraw to display the resulting metaballs entity when the *Approximate metaballs surface in modeling window* option is turned on. When this happens, you will be warned that the viewing method is being switched to the *Show only spheres in modeling window* option.

If you change the setting later to avert this situation and want to view an approximation of the metaballs entity, you must change the viewing method in the Edit Primitive dialog as described earlier in this chapter.

HINT: The metaballs entity described in the previous example should render as expected in most cases even though the redraw is forced to show spheres only. However, you should always choose the Raytracer renderer for such models to ensure each pixel in the model is rendered.

Radius scale

This field sets the scale of the entity produced during the Metaballs operation. A setting of 1.0 produces an entity with the same size general dimensions as the original spheres. Obviously then, a setting of 2.0 would be double that size and a setting of 0.5 would be half the original scale.

It is important to remember that changing the Radius scale without adjusting the Surface focus will result in an entity that is not the same scale as the original model. For this reason, it is strongly recommended that you use the slider control to make adjustments to keep the entity's scale the same as the original model.



If you elect to use the Expert fields to adjust the sphere of influence, you must decrease the Surface focus proportionally as you increase the Radius scale to maintain the relative scale between the final entity and the modeled spheres.

***HINT:** Depressing the Caps Lock key on your keyboard prior to adjusting the Ball weight or Surface focus settings in the Expert box or in the Edit Primitive dialog will automatically force proportional adjustments to the Radius scale. This is especially useful when you want to set either of these fields beyond the influence sphere slider bar's preset limits.*

Rendering the image

You should render metaballs entities using the Raytracer whenever possible. While other rendering engines should produce the same quality rendering of a metaballs image as they would with any other model, the Raytracer will actually work faster and take less memory to render a metaballs entity. It will also produce the most accurate rendering of the metaballs entity.

Which renderer you ultimately choose will depend on your needs and personal preferences. The methods for determining the rendering effects of a Metaballs image is the same as with any other model (consult the main User Manual for more information on rendering).

Adding textures

Like most Vision3d modelers, textures applied to objects before the modeler is used will be lost. This is especially true with the Metaballs extension, which cannot meld different textures across the deformation bridge created in the operation.

However, textures may be applied to the entity once the Metaballs operation is complete. The texture will be applied to each member of the Metaballs entity.



CHAPTER 4: BOOLEAN MODELER

Introduction

By definition, a Boolean operation is a mathematical way of treating variables through the operators AND, OR, NOT, IF, THEN, and EXCEPT. For our purposes, Boolean operations can best be described as a means of accomplishing such amazing geometric feats as physically punching a hole through a 3-D object, combining two objects into one, and creating a 3-D object from the intersecting points of two separate objects.

Strata's Boolean Modeler has been designed to accomplish these tasks with maximum efficiency and accuracy while presenting as little stress to the user as possible. In effect, complex geometry which could have taken hours of painstaking modeling to create, or been impossible, can now be accomplished with relative ease using the Boolean Modeler.

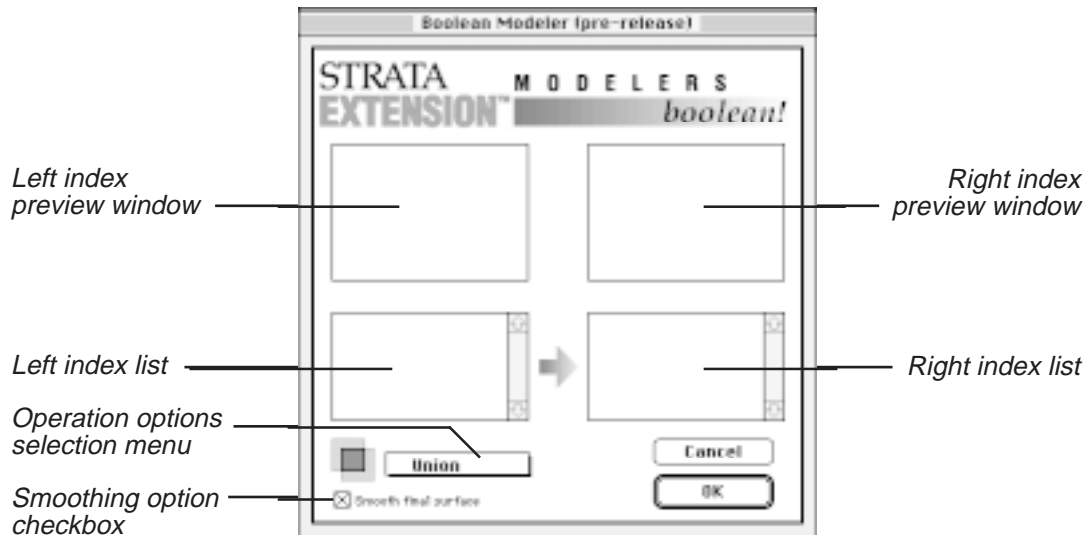
Boolean Operations

There are four distinct Boolean operations in the modeling extension: Union, which physically fuses two objects wherever they intersect to create a single, more complex object; Intersection, which, as the name implies, creates a geometric object based on the intersecting points of two separate objects; Subtraction, which negates a specific portion of an object to create a hole or indentation; and Cutting, which cuts away everything except the part of the first object which was inside the second object.

To use the Boolean Modeler it must be present in the Strata StudioPro Extensions folder prior to launching the program. Please refer to the installation instructions included with your program disks and in the READ ME files on the first installation disk for more information.

If present in the Extensions folder, the Boolean Modeler can be accessed through the Modeling menu or by clicking on the Boolean button on the button bar. When opened, the Boolean dialog box will appear to allow object and operation selection.

All objects in the active modeling window will be listed in the left and right index lists of the Boolean Modeler. However, Boolean operations can only be performed on those objects sharing common space. The reason for this should be obvious – Boolean operations join



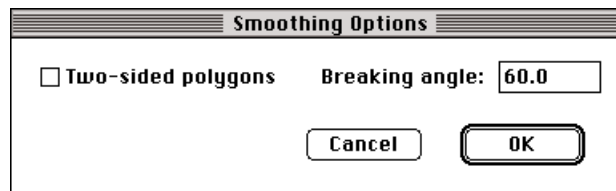
objects where they intersect, remove intersecting areas, or create objects from separate geometry sharing the same area.

When objects are selected from the index lists, a wireframe representation of each is displayed in the corresponding preview window directly above the index. Once two objects are selected, you must decide which of the Boolean operations you wish to perform on those objects.

You also have the option of smoothing the final object during the Boolean operation. If you choose not to smooth the object, the finished shape may appear faceted when rendered.

If you elect to smooth the object from the Boolean Modeler, a Smoothing Options dialog will be displayed when you click OK, allowing you to set the parameters of the smooth operation.

The Smoothing Options dialog's Breaking angle field will default to 60 degrees, which has been determined the most effective by Strata engineers.



4.2



NOTE: *If you choose not to smooth the object as part of the Boolean operation, you can do it later using the Smooth Modeler.*

You can perform any number of different Boolean operations on an object, but each operation must be done separately.

Because the best way to understand the power of Strata's Boolean Modeler is to experiment with it, we have included a simple tutorial which demonstrates Union, Subtraction and Intersection.

Cutting

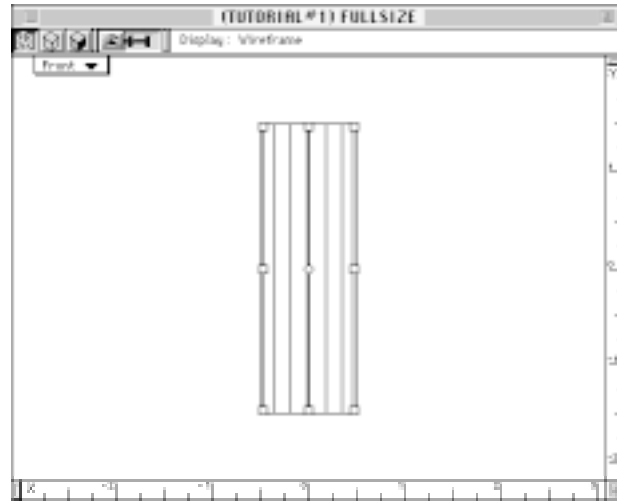
The Cutting function of the Boolean Modeler creates a new shape by deleting all of both objects except that part of the primary object which is inside of the second object when the operation is performed. The operation is similar to Intersection, but the Cutting operation does not supply any additional surfaces. Make sure your objects are listed in the proper order. Remember, the first object is cut by the second object.

Union

The Union function of the Boolean Modeler physically joins two objects at their intersecting points. Geometry sharing the same space is lost during the Union operation in favor of the new, volumetric shape. To demonstrate this melding characteristic, try the following tutorial.

NOTE: *Future tutorials on the various operations of Strata's Boolean Modeler will build on this model, so adhere to the instructions as much as possible, and make sure you save a copy of the file.*

1. With a new window open, create a new shape using one of the methods described in the Strata Vision3d User Manual. Name the new shape FULLSIZE (file and shape names will be capitalized in this manual to avoid confusion).
2. In the front view, insert a cylinder using the primitive cylinder tool in the Tool bar. Position the cylinder toward the top of the window so there is room to stretch it.
3. Use the Scale tool to pull the cylinder to a total length of three inches (you can also use the Transform command to accomplish this task), then use the Re-center command to ensure the cylinder's center point is positioned correctly.
4. Place the cylinder in the center of space by setting the X, Y, and Z coordinates in the Transform command to zero. You can also check



the cylinder's overall dimensions here. It should measure 1.0, 1.0, and 3.0 respectively in the X, Y, and Z Size coordinates.

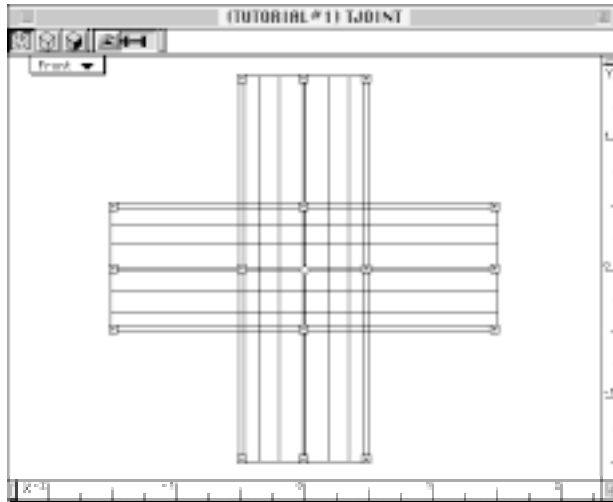
NOTE: This would be a good time to save the model. Give it a name like *TUTORIAL#1*.

5. Close the FULLSIZE WINDOW and create a new shape named TJOINT.
6. Insert an instance of the FULLSIZE shape in the new window (see Chapter 4 in the Vision3d User Manual for more information on shapes and instances).
7. Use the Transform command to ensure the instance is positioned at center space in the new window.
8. Insert a second instance of the FULLSIZE shape in the window and use the Rotate tool to rotate the new cylinder to the right 90 degrees so it is lying on its side relative to the first cylinder (you could also rotate the cylinder 90 degrees to the left, making the feedback dialog read -90 degrees).
9. Repeat Step 4 on this cylinder to position it at center space.

4.4



Your window should look similar to this illustration with two cylinders intersecting at their center points. If the image does not look correct, try using the Re-center command on the cylinders and reposition each to the center of space using the Transform command as described in Step 4.



NOTE: You should save your file again. There will be significant changes to the model in the following steps and having a pristine copy of the file is a good practice in case the results you get are not exactly what you want. You can, of course, use the Undo command immediately after performing any Boolean operation to return the shapes to their original condition, but having a backup file is always a good idea.

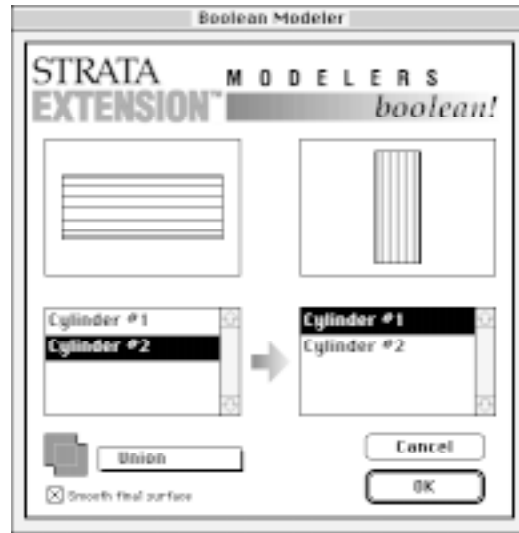
10. Now the fun begins. Open the Boolean Modeler by choosing it in the Modeling menu or by clicking on the Boolean icon in the Button bar.
11. When the Boolean Modeler dialog window opens, select the top listing of FULLSIZE from the left index list and the bottom listing of FULLSIZE from the right index list.

NOTE: Right and left index selections are somewhat arbitrary when performing the Union and Intersection operations. Note that the center point of the resulting object created by these operations will be the center of mass of the original objects. This is important because it prevents the final object from being moved in space during the Boolean operation. When performing the Subtraction operation, deciding the order of object selection becomes crucial. This relationship will be fully explained in the Subtraction section of this manual.



- The modeler opens with the Union operation selected by default, so you do not need to make any changes there (you may, however, want to click on the operations option box just to see the other choices).

This illustration shows the wireframe representations of each selected shape in the window directly above the corresponding index list. Note how the position of each cylinder clearly lets you know if you have selected the two shapes correctly to perform the Union operation.



- When you are sure each cylinder is selected and the Union operation is on line, click OK (obviously, clicking Cancel will abort the operation and return you to the active window without affecting the model). A dialog will appear to keep you informed on the progress of the operation.



- When the operation is complete the screen will redraw to display a single, T-shaped object.
- Switch the window's view orientation to Left and insert a third instance of the FULLSIZE shape.
- Rotate the cylinder instance right 90 degrees as in Step 8 and center the cylinder using the Transform command.

4.6



NOTE: You may want to double-check the T-shaped object to ensure it too is positioned at center space before performing the next operation. If the objects are not properly lined up the resulting shape may not be exactly what you want.

17. Repeat Steps 10-13 to fuse the third cylinder (listed in the index as FULLSIZE) with the newly created POLYGON MESH #2.
18. When the operation is complete the screen will redraw, this time displaying a polygonal mesh which, with a little imagination, resembles a toy jack.

This illustration indicates what the new polygonal mesh should look like from the Isometric view orientation.



19. Save a copy of this model as TUTORIAL#2 using the Save a Copy As command in the File menu. You will need this file as it now exists for the next tutorial session on the Subtraction operation.

The new, single shape is now complete as far as this tutorial is concerned. The new shape can be placed in the main workspace window as part of a larger model or pulled into the 3d Sculpter for detailed modeling. You may also want to apply a texture and render the image to satisfy your graphic curiosity. However, remember to save a copy of this file as it is for use in future Boolean Modeler tutorials.

Subtraction

The Subtraction function allows you to extract geometry from an object to create holes, indentations, and voids. This is accomplished by



completely removing the geometry of a shape from the model, taking with it any geometry sharing the same space in the active window.

In the real world this operation can be loosely compared to using an apple core remover to carve out the center of an apple. You begin with two complete objects—the apple and the apple core remover. When the remover is inserted through the apple’s center and pulled back out a section of the apple is removed, leaving a hole that is the exact shape of the core remover.

Obviously, the shape to be removed should be in contact with the shape that is to be carved. Otherwise, the geometry would merely be removed from space itself, leaving other shapes untouched. If removing the shape from the active window without affecting other shapes is the only purpose for calling the Subtraction operation, the delete key would be a much more efficient method.

For the purposes of this discussion, it will be necessary to return to the Union operation briefly to create the shape we want to remove from the existing TJOINT shape. This will serve as a valuable review of the Union operation and will demonstrate how Boolean operations can be used in combination to create complex models.

1. The first step in this exercise is to open the TUTORIAL#2 file you saved in the last tutorial.
2. Open the FULLSIZE shape window using the Edit Shape dialog in the Shapes menu and re-size the cylinder to be half an inch (0.5) in diameter and four inches long. This can easily be done using the Transform command by setting the Size coordinates for X to 0.5, Y to 0.5, and Z to 4 (make sure the radio button in the bottom, left corner is set to Absolute to perform this operation).

NOTE: *In most modeling scenarios, changes made to an object in the Shape window will carry over to all instances of that shape. However, once the instances of the FULLSIZE shape were transformed in the Boolean Modeler, all connections to their parent shape were severed. This is an important distinction to remember when using the Boolean Modeler.*

3. Use the Re-center command in the Modeling menu to reposition the cylinder’s center point.
4. Close the FULLSIZE window and create a new shape window named INNER TJOINT.

4.8

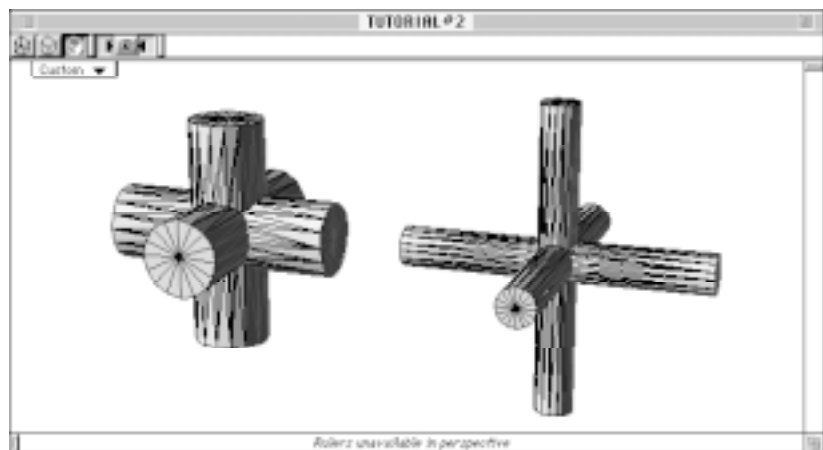


- Return to the Union operation tutorial (TUTORIAL#1) and repeat steps 6 through 18.

NOTE: When constructing the INNER TJOINT shape, try inserting and positioning all three instances of the revised FULLSIZE shape before using the Boolean Modeler. Just make sure you select a different instance of the cylinder in each of the index lists before you perform the Union operations. If each instance were built as a separate shape, they would have a unique name in the index list. Because our example uses multiple instances of the same shape, each has the same name in the index list. However, the modeler still treats each instance as a unique shape.

- When you have completed the Union operations, you will have created a second jack-shaped object identical to the first except in actual dimensions (each arm of the INNER TJOINT shape will be longer and thinner than those of the first TJOINT shape).

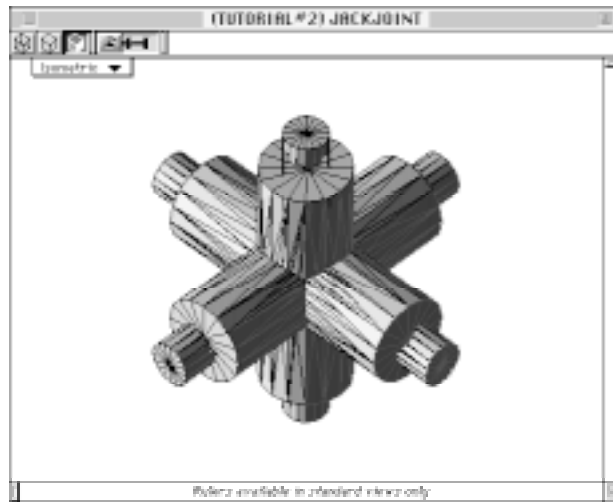
This illustration of the TJOINT (left) and INNER TJOINT shows how each shape should look in comparison to the other. It should also give you a hint of what we will accomplish in this tutorial.



- Create a third shape window using the New Shape command. Name this shape JACKJOINT.



8. Insert the TJOINT and INNER TJOINT shapes into the new window ensuring both are positioned at space zero (the X, Y, and Z coordinates set to 0.0) using the Transform command. This would be an excellent time to save your file again since the next step will result in significant changes to the model.



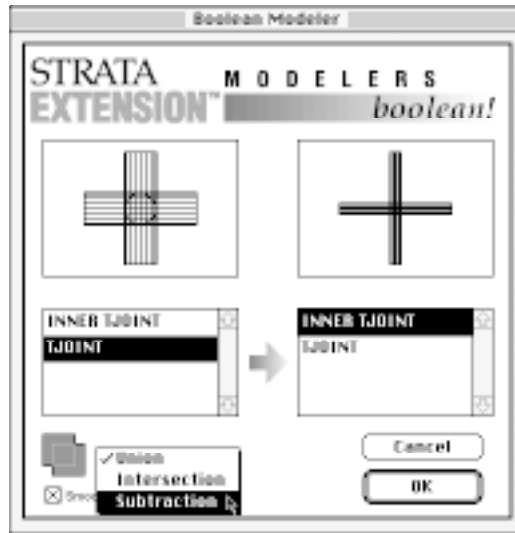
9. Open the Boolean Modeler and select the TJOINT and INNER TJOINT shapes from the left and right index lists respectively. It is imperative you select the correct shape in the proper index list for the Subtraction operation to yield the expected results. It is important to remember the Subtraction operation works on the mathematical premise of X (left index list) minus Y (right index list) equals Z (a new, single polygonal mesh).

NOTE: If you were to select the shapes in the wrong order and perform the Subtraction operation, the resulting geometry would be six half inch cylinders floating in space—those portions of the INNER TJOINT shape that extend beyond the geometry of the TJOINT shape.

4.10



If you have correctly selected the shapes from the proper index lists, the Boolean Modeler dialog should look similar to this illustration.



10. Select the Subtraction option from the operations dialog and click OK. When the operation is complete, the screen will redraw to display a new version of the jack-like shape with interconnecting holes running through the center of each cylinder.

The new shape you have created should look similar to the isometric illustrations shown here.



11. Save a copy of this model, this time with the name TUTORIAL#3, using the Save a Copy As command in the File menu. You will need this file intact for the next tutorial session on the Intersection operation.



To further emphasize the power of the Boolean Modeler, consider trying to drill a cylindrical hole halfway through a cube. Achieving such a feat geometrically without the Boolean Modeler would be virtually impossible. Though the effect could be simulated with complex texture maps, it would be merely a simulated look and may not meet your needs.

Intersection

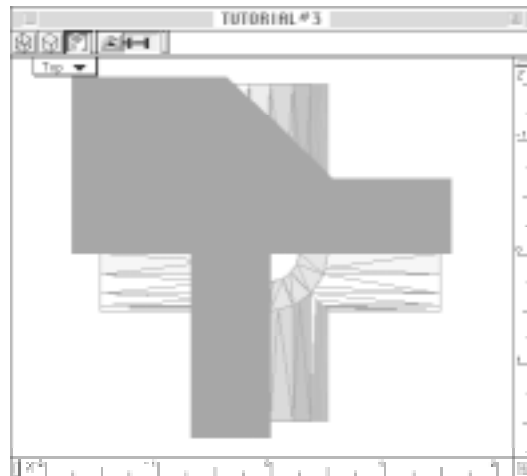
The Intersection operation of the Boolean Modeler creates an object derived from the geometry of two shapes which share the same area in 3-D space and eliminates all geometry that is not sharing space. While the result of this operation is similar to those produced with the Subtraction operation, the functions work opposite from each other.

This function is useful for creating cross-sections of the internal structure of objects and defining those areas in a model which overlap.

To demonstrate the Intersection operation, try the following tutorial.

1. Open the TUTORIAL #3 model you saved after the last tutorial session.
2. Insert the JACKJOINT shape into the main workspace window.
3. Change the view to the Top orientation, and using the closed, filled Polyline tool, draw a plane that covers various portions of the JACKJOINT shape.

You can follow the outline of this illustration or design your own plane. Keep in mind the new object will be constructed from the areas of the JACKJOINT shape that lie inside the borders of the plane.

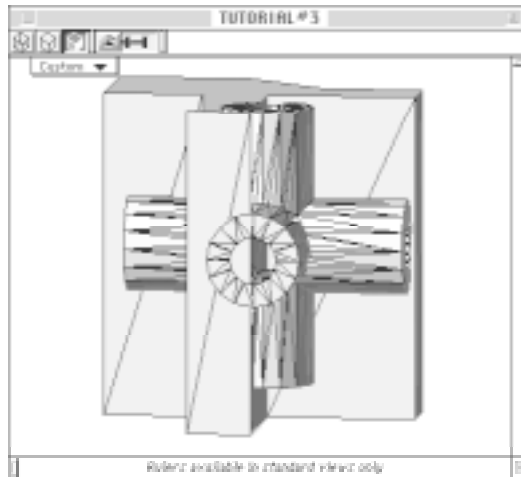


4.12



4. In the Front view, move the plane to a position slightly above the JACKJOINT shape on the Y axis (a quarter inch higher will be sufficient).
5. Return to the Top view, and using the Extrude extension, extrude the plane to extend below the JACKJOINT shape (a depth of 3.5 inches should be enough).

This Customized view provides an idea of how the JACKJOINT shape should be imbedded in the extruded plane. Again, those areas of the JACKJOINT shape lying inside the extruded plane will form the new shape's geometry during the Intersection operation.



6. Open the Boolean Modeler and select the JACKJOINT shape and Group #2, which is the extruded plane. As in the Union operation, deciding which shape is chosen from the left and right index lists is completely arbitrary.
7. Ensure the Intersection function is selected from the operational dialog and click OK to start the calculations.
8. When the operation is complete, the screen will redraw to display the cross-sectional view of the JACKJOINT shape.

This illustration should be similar to that on your computer screen if you have followed the tutorials faithfully.





This is the final exercise on the Boolean Modeler. From this point on, you can apply textures, attend to details in the 3d Sculpter, render an image, or perform any other function you would normally do with any other three dimensional object.

The examples used in the Boolean Modeler tutorials have been designed to present information on the modeling extension's functionality and develop basic user skills, all of which can be applied to more complex projects. Experimenting with the modeler to discover the nuances of each operation and how to coordinate the functions for optimum productivity is the best way to discover the power of this addition to the Strata extensions family.

As with all Strata products, the Boolean Modeler puts the tools for creating award-winning computer graphics in the hands of users. Adding imagination and creativity is your job.

4.14



CHAPTER 5: MORPH

Introduction

This chapter will describe the operation of two extension sets that comprise the Morph package. They are the Particle Morph and True Morph extensions and they provide the tools to create changes called metamorphosis in an object's geometry over time.

The fact the changes happen over time should give you the first clue that this is an animation package and requires a minimum of three frames to depict the metamorphosis from one object to another (if the animation were only two frames you would see an instantaneous shift between two snapshots with no interim transformation frame). Obviously, the more animation frames that are dedicated to the morphing process the more subtle the changes between frames will be and the smoother the transformation will appear when the rendered animation is played back.

Common items

Although the two metamorphosis extensions contained in this package create a different animation effect, there are some common elements to their operation. To avoid redundancy, these are presented here. Again, these points apply to both extensions.

Morphing multiple objects

The animation effect produced from each of the extensions can only be given to one object or group at a time. This does not mean only one object can use the animation effect in a model, or that only one object can be morphing at a time and must be fully morphed before the next object can begin. This means you can only work with one object at a time when you apply a particular animation effect.

You can, in fact, have as many objects morphing as you want, and they can all do it at the same time. The number of objects it is possible to morph depends entirely on the available memory.

Working with grouped objects

With True Morph you are expressly limited to morphing between single objects that are bézier in construction. However, using the Particle



Morph extension you can morph a group of objects to a single object or another group. While this is allowed, it is not usually the best way because any textures that may be applied to the individual objects within a group will be lost.

A better technique is to ungroup the objects and morph between selected components of the two groups, as shown in the following illustration:

Whenever possible, you should morph between individual objects. This allows you to keep, and morph, between the separate textures applied to each object.

Simple morphing between groups loses the textures applied to objects within the groups.



The same loss of textures occurs when attempting to morph between shape instances where the texture has been applied at the shape level. Unless the texture is applied to the instance, it will be lost in the morphing process. The texture must be applied at the same level as the morphing operation to be retained.

In the case of shapes, a usable technique for keeping applied textures during morphing is to ungroup the instances (breaking their link to the respective shapes), then apply the desired morph animation effect to them as original objects (instead of shape instances).

Caveats

- A. The most important thing you should realize about these animation extensions is that they require a significant amount of additional memory to perform their tasks. You should anticipate this requirement to ensure that you: 1) have enough memory within the Vision3d partition, and 2) will still have enough memory afterward

5.2



to render an animation of the morph event. This avoids modeling yourself “into a corner,” metaphorically speaking.

- B. Once a morphing operation has been done, a procedural morph entity replaces the original objects in order to save memory. The Undo command is available immediately after the operation, but once you move on to other things there is no way to recalculate the morphing. To change the objects between which morphing occurs, you must start over.
- C. The morphing extensions create a significant number of additional objects in your model. It can be difficult to remove all of the objects and restore the original objects if you change your mind later. Thus, you should consider making a copy of the model before using any animation extension. It will be easier to return to the previous version of the model than to remove the animation effect (unless you use the Undo command immediately after). Yes, you can use the Revert to Saved command, but what if you’ve saved the model with the animation effect(s) in place, then change your mind? It is always a good idea to make copies of your model(s) at significant “crossroads” in a project.
- D. If you wish to change the time references of the morphing animation effect you must use the Edit Primitive command found in the Modeling menu. The Re-Animate command is not available for morphed segments.



This command provides a dialog for the “Morph” procedural entity similar to the one shown in the following illustration. This is where you can shift the beginning and/or change the duration of a

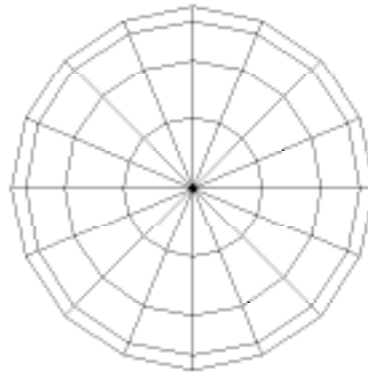
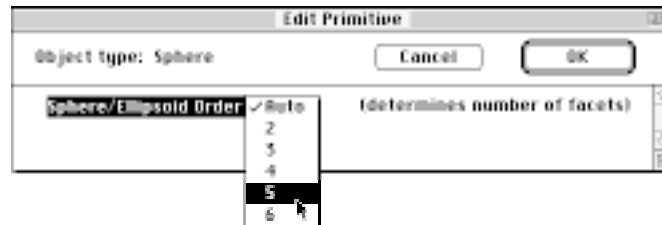


morphing event on the time line. Alternately, you can simply drag the event markers on the time line in the Animation palette.

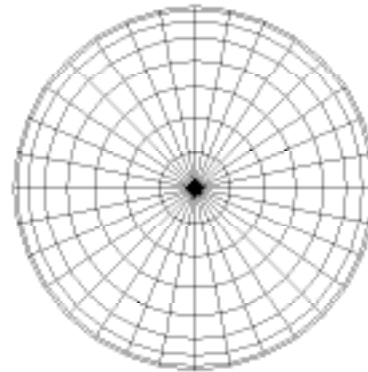


- E. In the process of creating a procedural morph entity, you may notice a slight loss in the morph entity's simulation of the original chosen objects' geometry. This is most noticeable when working with primitives and bézier spline objects in the Particle Morph extension.

To generate a finer geometry in the Procedural Morph Entity, set the facet order of the original object higher BEFORE morphing.



Facet order set to "Auto"



Facet order set to "5"

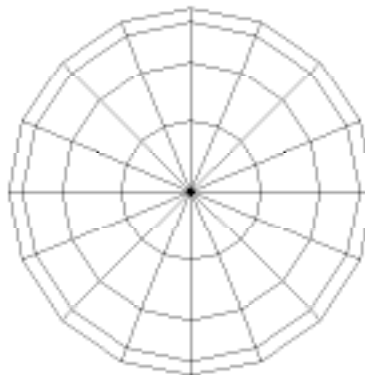
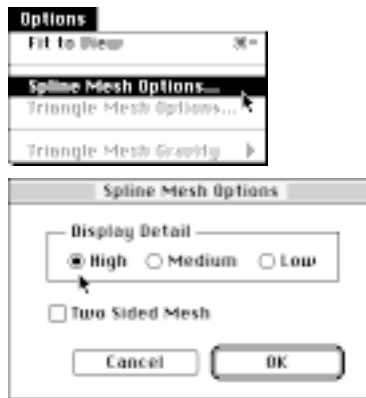
If you require tighter fidelity to the original objects when using Particle Morph, you can select each of the objects BEFORE morphing them and increase the faceting order settings for those objects in the Edit Primitive dialog. This will generate a finer representation from which the morph entity will be created. Note that this technique also increases the number of polygons and uses more memory.

5.4

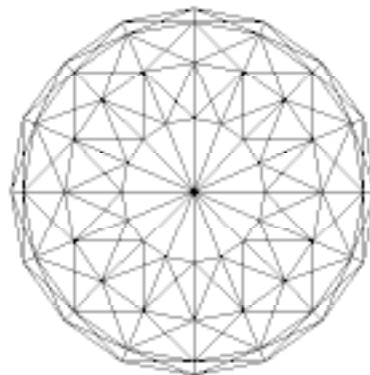


To improve the geometric quality of True Morph entities made from primitives, convert the beginning and ending primitives to bézier and set their display detail to high BEFORE morphing.

The Spline Mesh Options command and its dialog box are found in the 3d Sculpter modeler.



Normal sphere primitive with facet order on "Auto"



Sphere converted to bézier in 3d Sculpter

You can also improve the representation of primitives in morph entities when using the True Morph extension. If you open the primitive in the 3d Sculpter modeler (which converts the primitive to a bézier object) you can increase the display detail setting while in that modeler. This helps create a smoother profile or edge on the morph entity.

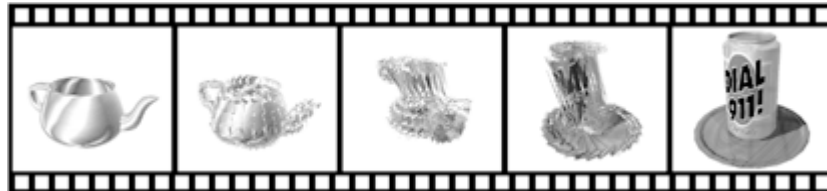
Particle Morph

As the name implies, the Particle Morph extension turns an object into a collection of particles that migrate into new positions as the

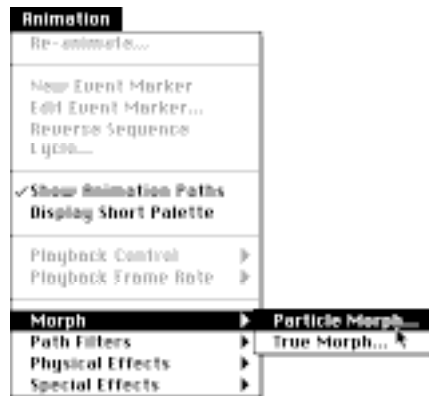


The Particle Morph animation effect is demonstrated in this illustration of a teapot changing into a soft drink can.

geometry of a second object is created. Any object may be particle morphed into another object. There is no limitation on the object types.



When the Particle Morph extension is present in the Vision3d Extensions folder its command appears in a Morph sub-menu of the Animation menu. This is how you access the Particle Morph animation extension.



If the Particle Morph command does not appear in the Animation menu as shown, the extension is not correctly installed. It must be within the Vision3d Extensions folder BEFORE the application is launched.

Description

By definition, metamorphosis requires a beginning and an ending object. The Particle Morph extension allows you to choose the objects between which a morph operation is to be done as well as the time it takes to complete it. These variables are set through a dialog that appears when a morphing command is chosen. The time at which the morphing process begins is set with the current time pointer on the Animation palette BEFORE calling this extension.



The Particle Morph effect is achieved by breaking the surface of the first object into a group of triangle-shaped polygons. To achieve a uniform distribution of the polygons as they replace the original object, special attention must be given to making all of the polygonal triangles the same size. These polygons are then moved into their respective positions on the second object during the specified duration.

It should be noted that the original objects (both beginning and ending) are deleted from the model and are replaced by a procedural entity that changes geometry over time. For this reason, there is no provision for re-morphing the two objects. However, you can use the Undo command in the Edit menu to reverse the creation of this procedural entity, if used immediately afterward. Otherwise, you must revert to the previous version of the model to completely remove the Particle Morph's effects from a model.

For this reason, if sufficient memory is available, it is recommended that you put the original objects in a shape workspace and perform the morphing operation on instances of the shapes. Remember, to keep the applied textures you must manually ungroup the instances before using this extension to break the link to shapes or the textures will disappear during morphing.

The benefit of using this extension is that it performs all of the calculations required to automatically create this morphing effect for the selected objects. A structure is created within the morph entity to manage the triangulated polygons. Morph entities can not be ungrouped.

The calculations performed during the morph operation extend to scripting the motion of the polygons as they move from the beginning object into position to form the ending object; thus achieving the desired morphing effect.

Only one set of two objects or groups may be morphed at a time. The number of objects that may be morphed within a single model is limited only by the available memory, but only one set of two objects can be scripted at a time. Please refer to the "Caveats" section at the beginning of this chapter concerning efficient memory usage.

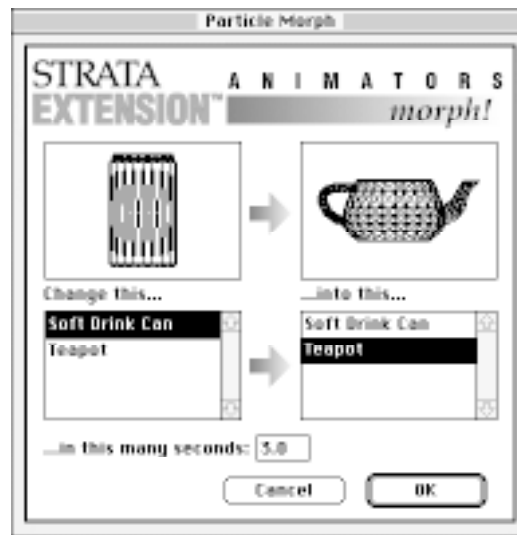
If an animation path is already scripted for an object that is subsequently used in a morphing animation effect, the morphed procedural entity will be substituted for the original object. The position of the ending object (in the model before the morph entity is created) is not retained. That means the morph entity will not automatically move



between the positions of the two original objects. However, after the morph entity has been generated, you may script movement for the morphing entity as a whole.

Using Particle Morph

The Particle Morph extension may be used in either the model space or a shape workspace. To use this extension you must already have at least two objects drawn in the workspace. To begin, position the time pointer to when you want the morphing to begin and select the Particle Morph command. A dialog box opens to set the effect's parameters.



The dialog allows you to select the beginning and ending objects, as well as the duration in which the morphing operation is to take place.



Upon clicking the OK button, the Particle Morph extension begins calculating the animation script, then generating the appropriate event markers and supplemental objects.



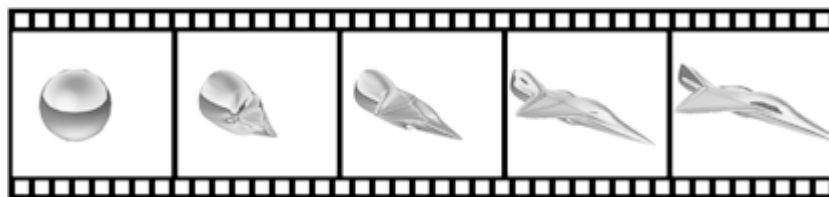
A progress bar appears to indicate activity and length of time required to generate the animation effect. When the Particle Morph extension completes its task, it closes and you are returned to the model.

This would be a good time to save your model before proceeding onto other aspects of the project. You may want to save the model with this special animation scripting under a different name in case you want to later remove or change the morph animation effect or go a different direction with your project. This is suggested because it is very difficult to restore the model to its previous state. As mentioned earlier, the original objects chosen for morphing are deleted from the model during this operation and replaced with a procedural morphing entity. You cannot go back to them afterward.

True Morph extension

The True Morph animation extension creates a metamorphosis between two objects that is visually smooth. The first object simply changes geometry into the second object. Unlike Particle Morph, the True Morph extension keeps the surface area intact during the entire process. However, only certain types of objects can be used with the True Morph extension. They must be "ordered spline objects." That includes all the 3-D primitives and a limited selection of special constructions by some of the modeling extensions. If you are not sure whether a specific object can be used with the True Morph extension, just open its dialog box. Only those objects that qualify are listed there.

This illustration demonstrates the smoothing effect produced by the True Morph extension.



When present in the Vision3d Extensions folder, the True Morph's command appears in the Morph sub-menu on the Animation menu along with the Particle Morph command. This is the method of accessing the True Morph animation extension.

If the True Morph command does not appear in the Animation menu as shown, the extension is not correctly installed. It must be within the Vision3d Extensions folder BEFORE the application is launched.



Description

The True Morph extension allows you to designate the beginning and ending objects, as well as duration. These variables are set through a dialog that appears when the command is chosen. The benefit of using this extension is that it performs all of the calculations required to automatically create a morphing animation effect between the selected objects.

The True Morphing effect is achieved internally by mapping the position of the first object's vertices to those of the second object. The vertices then move over time from their positions in the geometry of the first object to their respective positions in the geometry of the second object. The vertices take the most direct path between their first and second position.

To achieve a uniform movement of the vertices as they replace the original object, special attention is given to making sure there are the same number of vertices on both objects. If this is not the case, vertex points are added to various spline surface patches on the object with the fewest points.

Once the relative positions of each vertex point are established, the two original objects are replaced by a procedural morph entity that changes geometry over time. The original objects are consumed by the morphing operation and are no longer in the model. For this reason, there is no Re-Animate command (as is the case with some of the other animation extensions) that would allow you to select different objects and recalculate the morphing effect.

Once a morph operation is processed, the morph entity becomes part of the model. However, you can use the Edit Primitive command (in the Edit menu) to change the duration of the morphing effect after it has been created.



Only one set of bézier objects may be morphed at a time. The number of objects that may be morphed within a single model is limited only by the available memory, but only one morphing operation can be

5.10

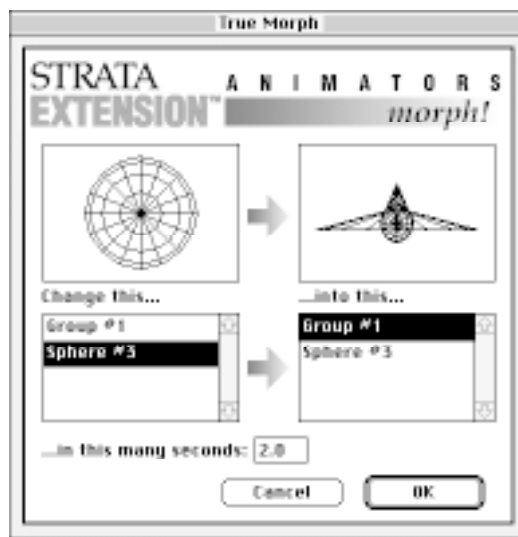


calculated at one time. Please refer to the “Caveats” section at the beginning of this documentation concerning efficient memory usage.

If an animation path is already scripted for an object that is subsequently used in a morphing animation effect, the procedural morph entity will be substituted for the original object. As with the Particle Morph extension, the position of the ending object (in the model before the morph entity is created) is not retained. That means the morph entity will not automatically move between the positions of the two original objects. However, after the morph entity has been generated, you may script movement for the morphing entity as a whole.

Using True Morph

The True Morph animation extension may be used in either the model space or a shape workspace. To use this extension you must already have at least two bézier spline objects drawn or inserted in the model or shape where you want this effect to occur. First, position the time pointer on the time line when you want the morphing effect to begin and select the True Morph command. A dialog box opens to set the effect's parameters.



The dialog allows you to select the beginning and ending objects, as well as the duration for the morphing operation.



Upon clicking the OK button the True Morph extension begins calculating the animation script, then generating the appropriate event markers and supplemental objects.

A progress bar appears to indicate activity and length of time required to generate the animation effect. When the True Morph extension completes its task it closes, and you are returned to the model.

This would be a good time to save your model before proceeding on to other aspects of the project. You may want to save the model with this additional animation scripting under a different name to preserve the model as it was BEFORE incorporating this effect in case you want to later remove all the morph-related objects or go a different direction with your project.

The preceding discussions on the Morph extension sets were designed to provide the basic skills and knowledge you need to use these tools. While your models will undoubtedly be more complex, the same procedures that have been outlined here will serve as a guide in every morphing sequence you order through these extensions.

5.12



CHAPTER 6: *PATHEXTRUDE*

Introduction

The PathExtrude extension gives you the power to extrude a two dimensional object through space on a pre-designated path to create a unique 3-D shape. The new shape can twist and wind through space, changing direction, size, and dimensions.

To script the path the object is to follow, you can either move the 2-D plane manually, adjusting the pitch and angle along the path, or use the Convert to Path and Align to Path extensions to automatically adjust variables along a pre-drawn path you have created. Both methods work equally well with the PathExtrude extension.

The 3-D shape created by the extension is a FlexObject™ similar to the type created with the Skin extension. However, the FlexObject™ created by the PathExtrude extension does not have animatable ribs like skinned shapes. This is an important distinction you should consider when using this extension. (For more information on the animatable characteristics of skinned objects refer to your main User Manual.)

While you cannot animate the individual ribs of the PathExtrude shape, you can animate the entire object through scripting just as you would any other shape in your model.

When using the PathExtrude extension, any connections to other modelers that may have been present prior to the operation will be lost. This is typical of all Strata extensions and will not prevent you from editing the new shape the same as you would any other shape.

Using PathExtrude

The extension must be present in the Strata Vision3d Extensions folder prior to launching the program. If properly installed, the PathExtrude extension can be called up through the Modeling menu as a sub-menu in the Swept Surfaces options menu.

If the PathExtrude option does not appear in the sub-menu, consult the installation instructions.



Creating a separate path

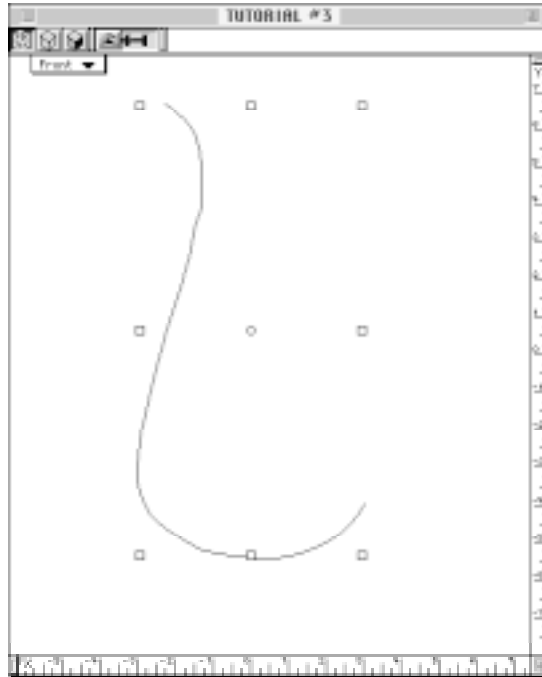
To help illustrate how this extension works, let's start with a simple tutorial that will provide the basic tools you will use for future projects. The following tutorial will take you through the steps of creating an saxophone-like shape with the extension by using a 2-D circle and an open-ended line that will be converted into a path.

1. With a new window open, create a New Shape named CIRCLE and draw, you guessed it, a filled circle in the shape window using the Shift key to constrain the shape. Make the circle one inch in diameter.
2. Open the Animation pallet and set the duration time line to one second.
3. With the circle selected, move the time indicator to the 0.5 second mark and scale the circle to 1.5 inches in diameter. Now slide the time indicator to one second and, with the circle still selected, scale the circle to three inches. (You can use the Transform command to set the scales accurately.)
4. Close the shape window and, with the Freehand line tool, draw an open-ended line resembling the curves of a saxophone. This will serve as a path for the extruded shape to follow. Make sure you draw the line from the top (the mouthpiece section) to the bottom.

6.2



Your line should look similar to this illustration. Use the 2-D Sculpter to smooth the curves and refine the line if necessary.



5. With the line selected, order the Convert to Path extension from the Animation menu and connect the CIRCLE shape to the path. (For more information on this function and that of the Align to Path operation mentioned later in this tutorial please refer to your main User Manual.)

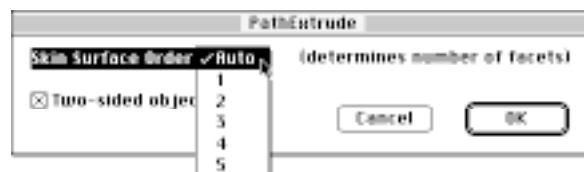




6. Use the Align to Path extension to line up the circle with the path. Make sure the surface area of the circle is facing forward in the Align to Path dialog before clicking OK. After the operation, you can use the Preview button in the Animation pallet to ensure the surface of the circle maintains a 90 degree angle from the path as it traverses the distance. Do not use the Bank on Turns option for this operation.

NOTE: The Align to Path extension does not recognize non-filled 2-D objects. Therefore, any extruded object you create that requires alignment with the Align to Path extension must have filled endcaps. If you must use the Align to Path extension and must also create a "hollow" object, you can facet the object after the PathExtrude operation and remove the endcap, or use the Boolean Modeler to hollow out the object. More information on these operations can be found in the main User Manual and in this Addendum.

7. When you are satisfied the circle is properly lined up along the path, choose the Ungroup command in the Modeling menu. This is an important step to remember because the Convert to Path operation creates a grouped object from the circle and line, and the PathExtrude extension will NOT work on grouped objects. Therefore, you will not be able to call up the PathExtrude dialog until you have performed the ungroup operation.
8. With the object selected and ungrouped, call up the PathExtrude dialog. You can change the Skin Surface Order to increase or decrease the surface detail of the object you are about to create. A setting of "5" will create the highest detail on the object's surface, but it also generates the largest number of polygons in forming the FlexObject™ mesh. If memory is a concern, you may want to set the Skin Surface Order to a lower setting. Use the Auto setting for this model.



You also have the option of making the FlexObject™ a single-sided or two-sided object (consult your main User Manual for more information on single and two-sided objects). Set this option box to create a two-sided object.

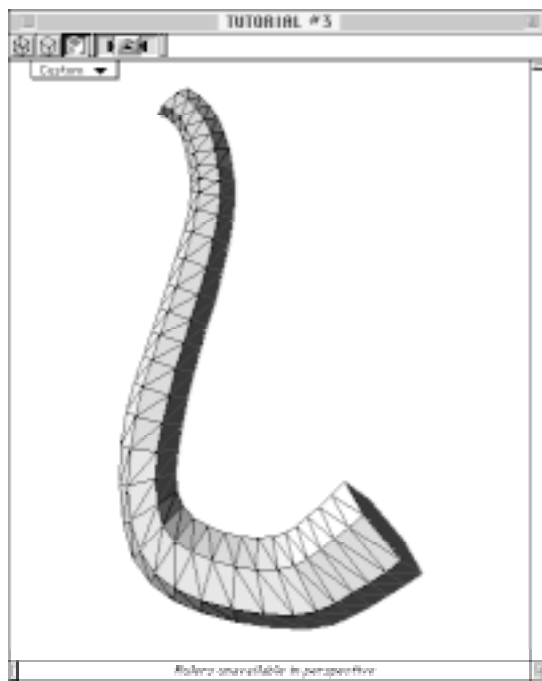
6.4



9. Click OK when you are ready to extrude the circle. When the operation is complete, the screen will redraw to display the new object. If the operation produces something other than what you want, you can use the Undo command to return the model to its previous condition.

HINT: The Undo command is available for all operations, but **must** be used immediately after the operation is complete.

If you followed along with the tutorial, the object on your screen should look similar to this illustration.



You may now fine tune the object in one of the other Strata Modelers, attach other pieces to produce a more realistic saxophone, and apply textures to give the object the look you want in a final rendering.

Creating a manually scripted path

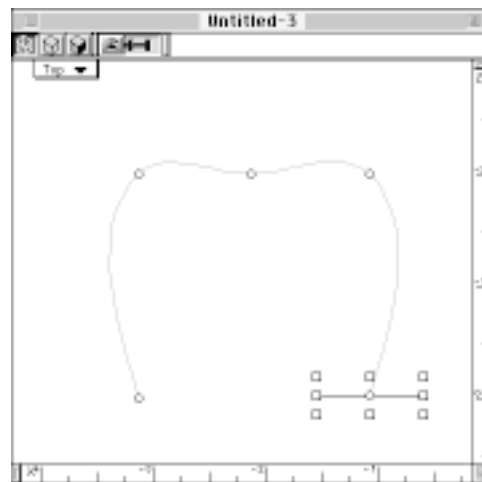
Manual path creation is simply the process of creating a shape and physically dragging it to different positions while advancing the time indicator slider on the Animation pallet. This procedure is fully discussed in the main User Manual. For purposes of this example, no



further explanation of this process will be discussed unless it pertains to the tutorial.

1. Open a new window and set the animation duration to one second (use the L-shaped slider to do this). Then, with the Animation pallet's time indicator (the pointed slider) set to zero, create an unfilled, one inch square with the 2-D rectangle tool (use the Shift key to constrain the rectangle's sides to equal length).
2. Change the view orientation to Top and slide the time indicator to .25 seconds. Then, drag the square two inches toward the top of the screen (toward the negative scale of the Z axis). An event marker should appear in the Animation pallet's time line.
3. Without moving the time indicator, rotate the square 45 degrees with the Rotate tool. Make sure you rotate the square to a positive angle as the next step will move the square in that direction.
4. Slide the time indicator to .50 seconds and drag the square one inch to the right (toward the positive scale of the X axis). Again, rotate the square 45 degrees with the Rotate tool without moving the time indicator.
5. Move the time indicator to .75 seconds and drag the square another one inch to the right. Again, rotate the square 45 degrees.
6. Slide the time indicator to the one second mark. Drag the square two inches toward the bottom of the screen (this time toward the positive scale of the Z axis) and rotate it another 45 degrees.

If you have correctly scripted the animation, your screen should look similar to this illustration.

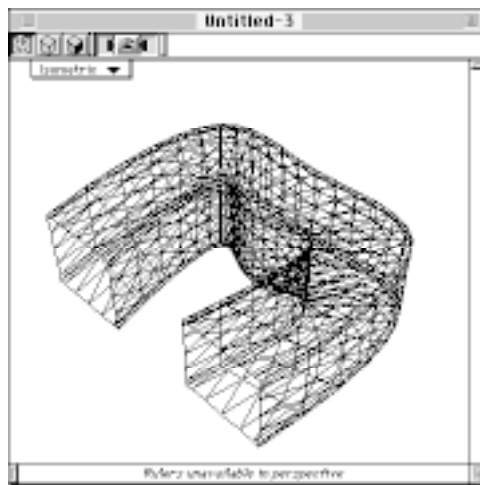


6.6



7. Return the time indicator to the zero second position and, with the square selected, open the PathExtrude dialog.
8. Set the Skin Surface Order to Auto and ensure the check box is marked to create a two-sided object. (Remember, we are using a non-filled object in this exercise. Therefore, you would not be able to see the inside surface of the extruded object if this option is set to single-sided.)
9. Click OK when you are ready to perform the operation.

When the operation is complete the screen will redraw to display a FlexObject™ similar to this wireframe illustration shown in the Isometric view orientation.



The objects created in the PathExtrude modeler may be animated just as any other shape, or added to a larger, more complex model. The uses for this powerful extension are limited only by your imagination.



CHAPTER 7: WARP

Introduction

Strata's Warp extension opens the doors to a vast new world of three dimensional possibilities for Vision3d users, including the power to twist shapes and send them spiraling into space. These exciting effects can be applied to a static object in a snapshot or compiled as an animation sequence to create uniquely artistic movement.

Through the modeler's three slider controls you can warp an object or grouped object by wrapping it around an invisible center post, cause it to spiral toward an ever shrinking vortex, twist it into a tightly coiled shape, or any combination of these effects.

The extension's interface is designed to provide a friendly atmosphere for the user with easy-to-understand dialogs. Controlling how the modeler will affect objects can be done through simple slider bars or by entering numeric settings from the keyboard.

Finally, a simple click turns on the animation option and allows you to create an animation of the Warp transformation.

Using Warp

The extension must be present in the Vision3d Extensions folder prior to launching the application for the Warp modeler to appear in the Vision3d Animation menu. If the Warp modeler is not listed in the Animation menu, please consult the installation instructions included with your Vision3d package.

The extension allows you to use a single object or a grouped object to achieve the desired effect. In fact, using multiple objects grouped together often produces better results because it presents the modeler with more reference points to perform its calculations. As you will see during this discussion on the Warp modeler, the more reference points present during the operation, the better the detail and accuracy of the finished product.

To access the Warp modeler you must have selected the object or grouped object you want to use for the effect. Once the object is selected, open the Warp dialog through the Animation menu. The extension's name will be dimmed if no object is selected.



The dialog presents the user with three slider bar controls, an Expert box, and an animation option. Each slider bar can be set to achieve results based solely on that setting or used in conjunction with any or all of the other controls.

The numeric readout on the right of each slider bar provides feedback on that slider's setting. You can also enter specific settings in the readout boxes through the keyboard. Entering values through the keyboard offers exacting control and lets you to exceed the maximum settings allowed through the slider bars. You will find it necessary in many modeling situations to set these fields manually to achieve the results you want.

Wrap degrees

The Wrap function determines how an object will be transformed around an invisible cylinder. The slider bar allows you to set a range from zero degrees to 360 degrees.

The wrap setting only affects the object's geometry from a horizontal orientation. This is because the calculations are formulated from a stationary point on the X axis which acts as a base. The object is then wrapped around that point along the Y and Z axes.

Two methods of wrapping an object or grouped object are available. These are Bend on wrap, which is the default setting, and Arc on wrap.

7.2



You can toggle between these methods by clicking on the appropriate radio button in the Expert box.

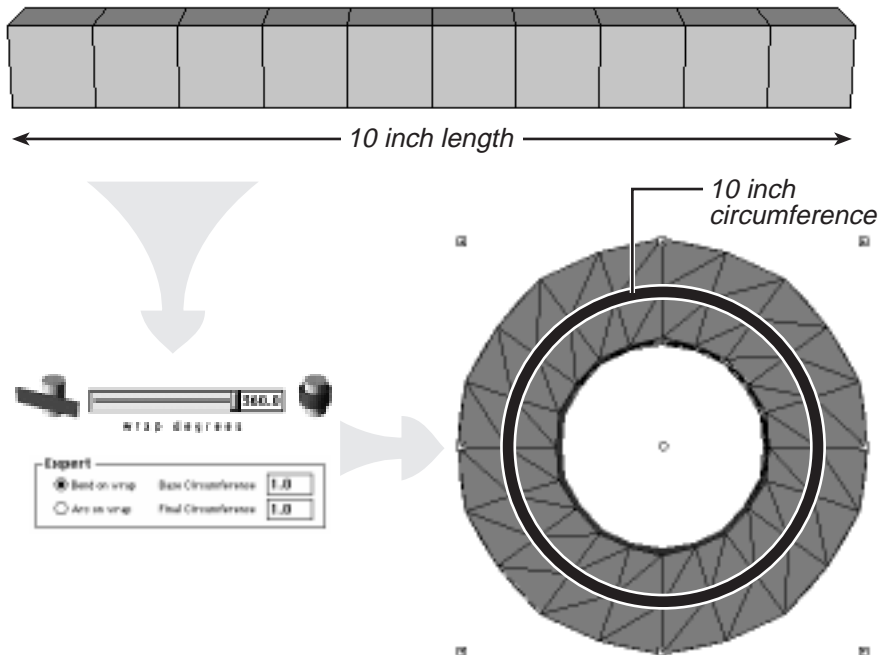
BEND ON WRAP

The Bend on wrap option can be compared to taking a metal rod and wrapping around a fixed post. The more times you wind the rod around the post, the tighter the wrap becomes.

The overall length of the rod does not change using the Bend on wrap option. This is an important distinction to remember because it differs from the Arc on wrap option.

Using the Bend on wrap technique, the modeler uses the geometric center of the object as a base for starting the wrap. The object is then bent equally from each end along the X axis. How far the object is bent depends on the number in the Wrap degrees field.

A setting of 360 degrees using the Bend on wrap Expert option will wrap the object into a complete circle with each end touching the other, as this Top view illustration shows.

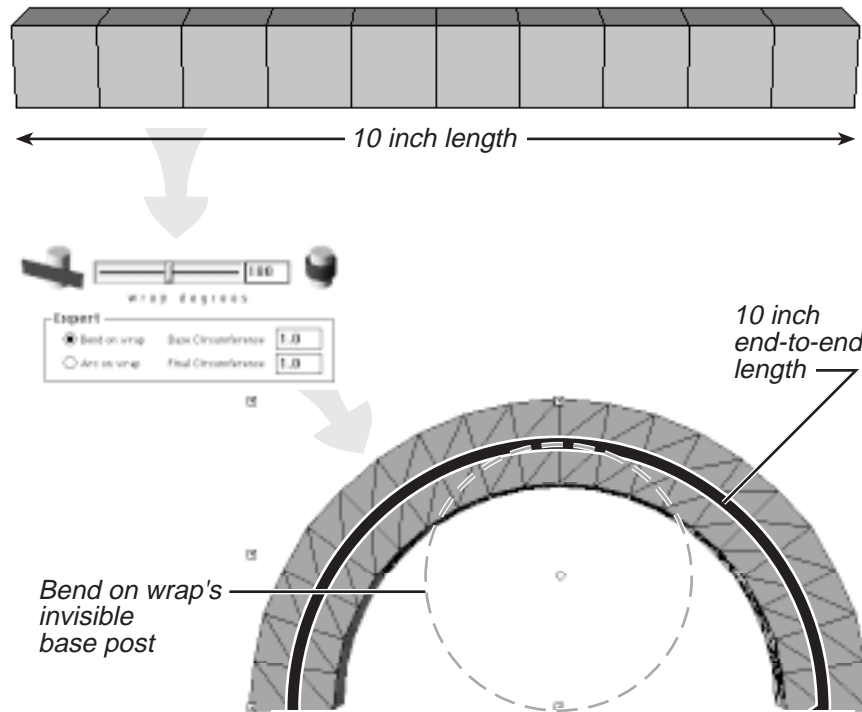


The previous illustration shows how a 10 inch grouped object is bent around the modeler's metaphoric base post. The length of the object, as measured along the X axis through the object's geometric center, is maintained in the wrapped object's end-to-end measurement.



The following illustration shows how a Wrap degree setting less than 360 degrees, 180 degrees in this case, still results in an end-to-end length equal to that of the original object.

Notice how the base post in this Top view is located at the center of the original object. The ends are then bent equally toward the post.



The base post will always be centered on the object along the X axis. This is true even if the object's center point is moved. The size of the metaphoric base post depends on the overall length of the original object along the X axis.

Animating the wrap operation with the Bend on wrap option selected lets you see how the object maintains its overall length

7.4

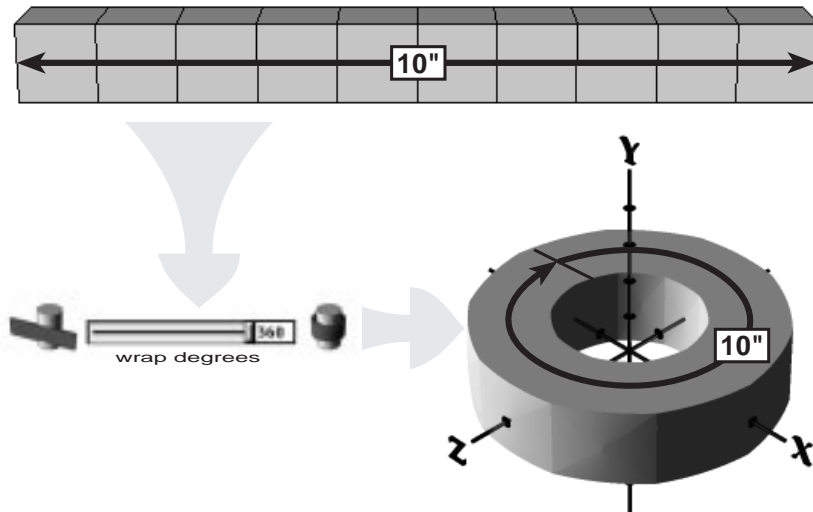


throughout the warping process. This is another distinct difference between the Bend on wrap and Arc on wrap operations. Animating a Warp operation will be discussed more fully later in this chapter.

ARC ON WRAP

To perform the Arc on wrap calculations the extension uses the length of the object along the X axis to form the final object's total circumference as measured through the geometric center of that object. Once the circumference is determined, the extension fits the complete object into the number of degrees indicated in the dialog. This differs from the Bend on wrap function.

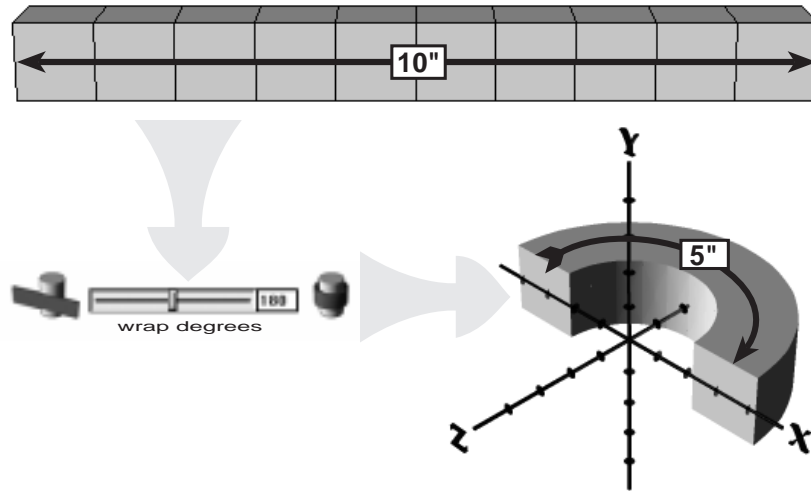
As you can see in this illustration, the length of the grouped object before the Warp operation is 10 inches. That measure is then used to set the circumference along which the object will be wrapped. The degrees in the dialog determines how much space along that circumference the wrapped object will occupy.



As the previous illustration shows, an object that measures ten inches before the operation will be wrapped relative to a ten inch circumference. If the Wrap degrees field was set to 360, the wrapped object would have an end-to-end measure of 10 inches – as measured through its center where the circumference measurement is based. However, if the Wrap degrees is set to something other than 360 degrees while using the Arc on wrap Expert option, the total length of the finished object will not be the same as the original object.



In this example, the Wrap field was set to 180. The base circumference is still 10 inches, but the end-to-end measure of the object is five inches.



HINT: A field setting of 45 degrees, using this example, would result in an object wrapped around a 10 inch circumference with an end-to-end length of only 2.5 inches. Conversely, a field setting of 720 degrees would produce an object measuring 20 inches. A discussion describing when and why you would use settings of more than 360 degrees is included in the Pivot section later in this chapter.

A setting of less than 360 degrees in the Wrap degrees field may produce an object that appears as though it were squashed like an accordion to fit all its parts into the number of degrees allocated. In fact, that is exactly what has happened. The circumference measure (determined by the total length of the object prior to the operation) does not change with the Wrap degree setting. Instead, the entire object gets squeezed into that portion of the circumference designated by the Wrap degrees setting.

You can increase or decrease the circumference around which an object will be wrapped by moving the object's center point along the Z axis. Moving the center point by depressing the Control key and dragging the point with your mouse toward positive coordinates on the Z axis will increase the circumference of the object when it is wrapped. Dragging the center point toward negative coordinates on the Z axis will decrease the final object's circumference.

7.6



How much the circumference will increase or decrease depends on how far you drag the center point. Basically, the object's circumference can be determined by adding (or subtracting if the center point is moved toward negative coordinates) the radius of the original wrap cylinder prior to moving the center point and the distance the center point is moved along the Z axis.

When the operation is complete a new spline mesh object will be displayed on the screen in place of the original shape. If a grouped object was used in the Wrap operation it will be redrawn as a grouped spline mesh. However, such an object can be ungrouped to make each of its parts accessible for editing.

NOTE: Like all functions in the Warp modeler, wrapping an object using the Arc on wrap option can be animated. You should note the beginning object will appear to be a collapsed version of your original model. This is correct and will be explained in the discussion on animating a Warp operation later in this chapter.

As was mentioned earlier, you can increase the maximum number of Wrap degrees by highlighting the dialog box and typing in the number of degrees you want. However, setting the Wrap degrees higher than 360 degrees is nonproductive unless you use it in conjunction with the Pivot angle function. This relationship will become apparent in the following discussion.

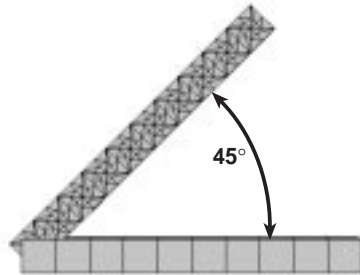
HINT: As was mentioned earlier, multiple objects grouped together often produce the best results when using the Warp modeler. It is strongly recommended you use multiple, grouped objects for Warp modeler operations whenever possible. If a single object is to be used in the operation, you should ensure there are ample vertex points in the object. You can add vertex points to an object in the 3D Sculpter. For more information on this process please refer to your User Manual.

Pivot angle

The Pivot angle setting allows you to rotate the object in relation to the Y axis. This is done by using the object's leftmost corner as an anchor on the Y axis, then rotating that object's rightmost corner up or down along the Y axis to the set number of degrees in the dialog.



If the Pivot angle is set to 45 degrees and the other operational fields are set to 0.0, the object will merely be rotated 45 degrees.

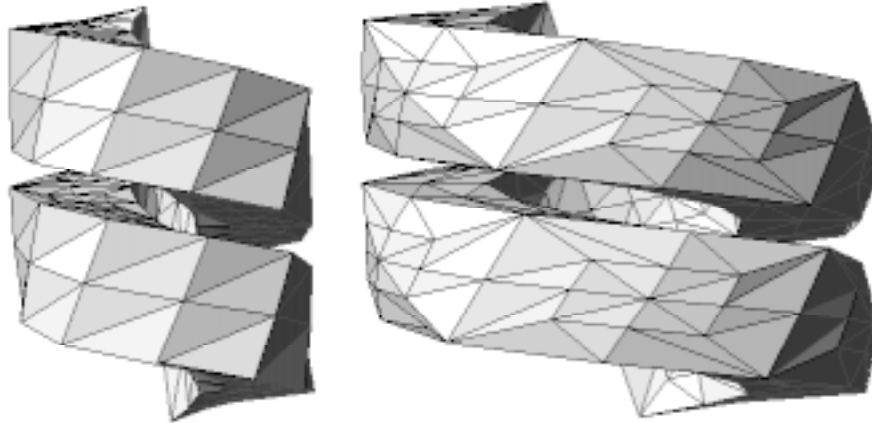


A positive entry in the Pivot angle dialog will rotate the object upward while a negative entry will result in a downward rotation.

The Pivot angle operation is designed to function with the Wrap operation and performs essentially the same with either the Bend on wrap or Arc on wrap Expert option. When the Pivot and Wrap operations are combined they become extremely powerful modeling tools.

For example, wrapping an object 720 degrees with a Pivot angle of 20 degrees would result in a cylindrical, spiraled object with two complete coils. This would be true using either of the Wrap function's Expert options.

These coiled objects each began as 10 one inch cubes lined up along the Y axis and grouped together. Each was warped with a Wrap setting of 720 degrees and a Pivot angle of 15 degrees.



Bend on wrap

Arc on wrap

As you can see in the previous illustration the Pivot angle works the same when using both the Bend on wrap and Arc on wrap Expert options. Obviously, the larger the pivot angle, the looser the coil and visa versa.

7.8



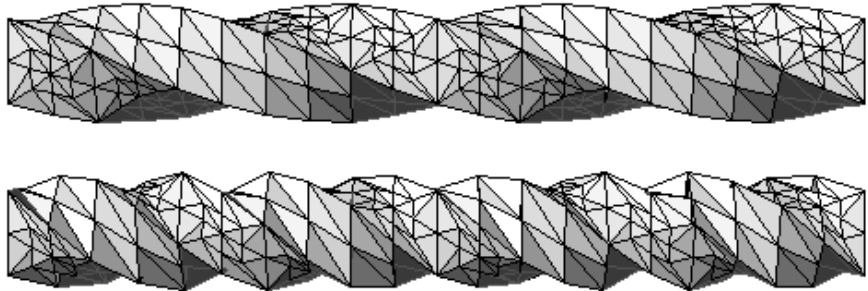
Twist degrees

As the name implies, the Twist degrees field determines the rotational relationship between an object's endcaps as it is twisted around its center. The maximum slider bar setting for this operation is 180 degrees. As with the other Warp operations, this can be set higher through the keyboard.

The Twist function operation is relatively straight forward. The field setting determines the extent the object is twisted from one end to the other along the X axis. It is important to note the twisting action always occurs along the X axis. Therefore, you should ensure the object is positioned correctly before performing a Twist operation to avoid surprises. Of course, experimenting with this and other functions is always encouraged and is the best way to master their use.

The Twist operation can be likened to twisting one end of an elastic band while the other end is held stationary – the more turns you put on the band, the tighter the twists become. You should note that, like the elastic band, an object twisted too tightly can begin to break apart at key polygonal points. This is especially true when a grouped object is used in the operation because, if the group's points are not properly aligned, each piece of the original group can “break” away from contiguous elements of the group when twisted too tightly.

Both objects in this illustration were formed from 10 one inch cubes grouped together. The top object was twisted 360 degrees. The bottom was twisted 720 degrees.



How far you twist any given object before it begins to break apart depends on the object's geometric structure and the number of reference points along its length. In many cases, an object's surface detail, and consequently the number of reference points the Warp modeler can use, can be increased before the operation through the Edit Primitive dialog in the Modeling menu. Other objects, such as cubes and cylinders, may require you add vertex points using the 3D Sculpter to increase the



number of reference points. For more information on the Edit Primitive option and 3D Sculpter please consult the main Vision3d User manual.

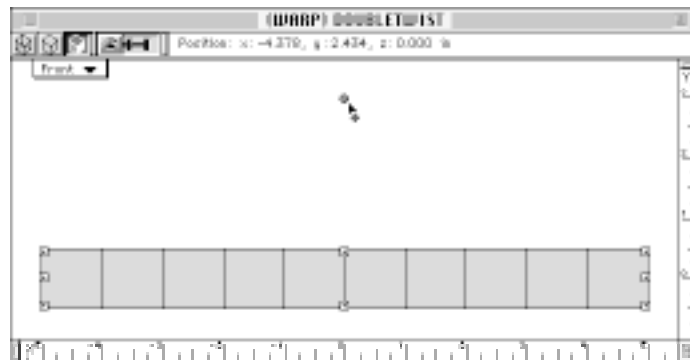
You can, of course, use the Twist operation in conjunction with the Wrap and Pivot functions. The resulting object will reflect each of the operational fields' settings. Therefore, understanding how each operation is performed can help you make intelligent, predictable modeling choices.

ADDING A DOUBLE TWIST

You have seen how an object can be twisted around its center point, but you can also twist the object around a point in space at the same time. The resulting mesh will be twisted around its horizontal center and its length will twist around a preset point lying outside its geometry.

To create this double twist effect, move the object's center point to some position outside the object's geometry by depressing the Command key and clicking and dragging the center point (see the Vision3d User manual for more information on moving center points).

This illustration shows the double twisting effect that occurs when the center point is moved. In this case, the center point of a group of 10 one inch cubes was moved three inches vertically and the Twist dialog was set to 180 degrees. The resulting object twists 180 degrees end-to-end and is warped 180 degrees around its repositioned center point.



Bringing the power of all three operations to bear on one object is a simple process, but careful planning is essential if you are to achieve the

7.10



effect you want on your first try. Experimenting with the Warp modeler on a simple object is a good way to ensure you understand how each function works and how each affects the other when used together.

Again, make sure you have enough vertex points in your model to create a smooth Warp entity. Trying to warp an object that has too few reference points could produce a faceted entity or one that breaks apart at key points.

Using the Expert dialog

The Expert dialog, located directly under the slider controls, allows specialized control over Wrap operations. With the Expert settings you can scale the beginning and ending circumference measurements and control the calculation method of a Wrap operation.

Expert		
<input checked="" type="radio"/> Bend on wrap	Base Circumference	<input type="text" value="1.0"/>
<input type="radio"/> Arc on wrap	Final Circumference	<input type="text" value="1.0"/>

Bend on wrap

The Bend on wrap option tells the modeler to wrap an object around a fixed base post centered on the object. The wrap operation always begins at the center of the object where the metaphoric base post is located and the overall length of the object is maintained throughout the process when this option is selected.

When animated, the object's ends will be drawn together equally around the base post to the set number of degrees in the Wrap degrees field. Again, the overall length of the object will be maintained at all times.

Bend on wrap is the default setting.

Arc on wrap

Wrapping an object using the Arc on wrap option produces an entity wrapped around a metaphoric base post with a circumference equal to that of the original object's overall length as measured along the X axis through the object's geometric center.



This option produces a shape which will not be the same length as the original object unless the Wrap degrees field is set to 360.

When animated, an object wrapped using the Arc on wrap option will literally collapse into itself at the beginning of the animation and grow outward from both ends equally until it is wrapped the number of degrees set in the Wrap degrees field.

For more information on the Bend on wrap and Arc on wrap Expert options please refer to the discussion on Wrap operations earlier in this chapter.

Base Circumference

This field sets the scale of the beginning circumference around which an object will be wrapped. As described earlier in this chapter, the Wrap function determines the circular path, or base post, around which an object will be wrapped. This path can be altered through the Base Circumference setting.

A setting greater than 1.0 in this field increases the size of the base on a percentage scale. A setting less than 1.0 decreases the base size. For example, a setting of 2.0 in this field would double the base circumference from its original size. A setting of 0.5 would shrink the circumference of the wrap path to half its original length.

The default for each of the circumference fields is 1.0, which is 100 percent or actual size.

Final Circumference

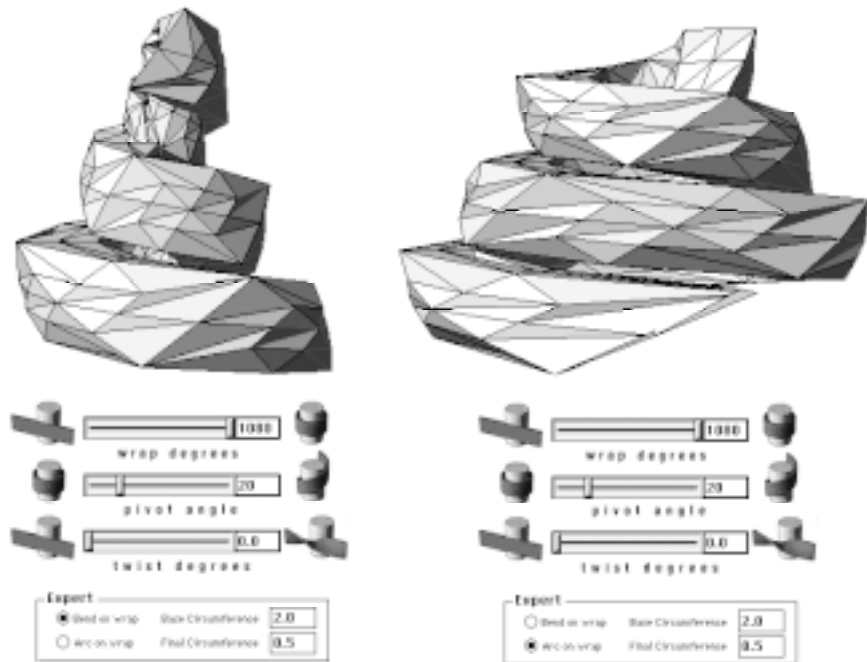
This field works the same as the Base Circumference setting except that it affects the circumference used on the final segment of a wrap operation.

By changing these fields when using the Wrap and Pivot functions, you can create an object that appears to make consecutively tighter circles as it winds upward around an invisible cone and visa versa.

The circumference fields do not alter the other settings used in the Warp operation, but changing their values will have a profound affect on the resulting object.



This illustration shows how the circumference settings can be used to produce a cone-like object. Note the difference between an object wrapped using the Bend on wrap and Arc on wrap options. The Base Circumference for each model was set to 2.0 and the Final Circumference field was set to 0.5.



Animating a Warp operation

The Warp modeler gives you the option of animating the operation to produce an image that changes over time according to the modeler's settings. The animation option check box, located in the bottom left corner of the modeler dialog, makes animating as easy as clicking the mouse. In fact, that is exactly how you turn on the Animate option.

When the Animate check box is activated, you can set the time it takes the object to go from its original state to a fully warped condition by entering a number in the Duration field. The default here is two seconds. Once the Warp operation is complete you can render the animation to produce an exciting movie tracking the object's transformation.

With exception of wrapped objects created using the Arc on wrap Expert option, animated entities will retain their basic dimensions during the process. That is, their overall measurements will only change in accordance with the modeler's field settings. The animation sequence will show the object transform from the model you created to a fully warped entity.



When using the Arc on wrap option, however, the object will start as a collapsed entity that will begin extending itself outward in both directions along the X axis as the animation progresses. This outward growth will follow the warping parameters established in the modeler's field settings until it has reached a fully warped state.

You should make sure your settings are correct before you chose to animate a warp sequence and always save your model prior to initiating the operation. This is especially important when animating a warp operation because the resulting object will be a procedural entity. As such, the animation sequence will be embedded in the entity, leaving no event markers with which to edit the animation once it is complete.

Adding textures

Textures may be added to Warped objects after the operation is complete. Textures applied to objects used in the Warp modeler prior to the operation will be lost.

If the shape was a grouped object prior to the operation, the resulting object can be ungrouped to allow application of textures to each element of the new object separately. This is useful when trying to create a unique look for the warped object. Ungrouping the entity also allows more flexibility in editing the final object.

You can use any of the methods described in the User Manual for applying a texture to a warped object.



Introduction

This chapter will discuss additional feature enhancements and functions not addressed in the main Vision3d user manual. These features provide greater control and versatility over your projects.

Some of the enhancements are new to Vision3d while others are improvements to previously existing functions. You should pay particular attention to discussions regarding improvements to existing functions, especially if you were familiar with earlier versions of the software, as the enhancements may have changed the way a feature performs its function or how it is used.

Alternate Resources Files

Vision3d resource files are used internally to determine how texture and gel previews appear in palettes and editing windows and to turn the shutter sound of a rendering camera on and off. Strata engineers have provided four such resource files so you can customize the look and sound of your displays.

You should note that one of the resource files *must* be present in the Vision3d folder at the same level as the Vision3d application and it must be named exactly for the software to perform correctly.

The default Strata Vision3d™ Resources file is automatically placed in the application folder during normal installation and provides a “tile” preview in the Textures and Gels editing dialogs. This resource file also produces a camera shutter sound to signal when an image is done rendering.



Three other resource files are stored in separate folders within the Alternate Resources folder, which is placed in the Vision3d application folder automatically during normal installation. These alternate resource



folders are named differently to identify their function. The actual files are all named the same because, as was mentioned earlier, Vision3d will only recognize the file if it is named Strata Vision3d™ Resources exactly.

- ▼ Alternate Resource Files f
- ▼ Silent Camera Resources f
 - Strata Vision3d™ Reso...
- ▼ Silent Sphere Resources f
 - Strata Vision3d™ Reso...
- ▼ Sphere Preview Resources f
 - Strata Vision3d™ Reso...

Because the resource files all carry the same name, you should use extreme care when changing the resource file. Always remove the current resource file from the application root window and place it in its unique folder within the Alternate Resources Files folder *before* installing a new resource file in the root window. If you fail to remove the unwanted resource file before placing the new file in the root window, the old file could be replaced with the new. A warning dialog will appear in this case allowing you to abort the action before the file is replaced.

If you inadvertently loose a resources file, you can always replace it from the Vision3d installation CD.

The three alternate resource files are stored in folders named:

Sphere Preview Resources f

Creates a sphere object for previews and produces a camera shutter sound to signal the completion of a rendering.

Silent Camera Resources f

Creates a tile object for previews and turns the camera shutter sound off.

Silent Sphere Resources f

Creates a sphere object for previews and turns the camera shutter sound off.

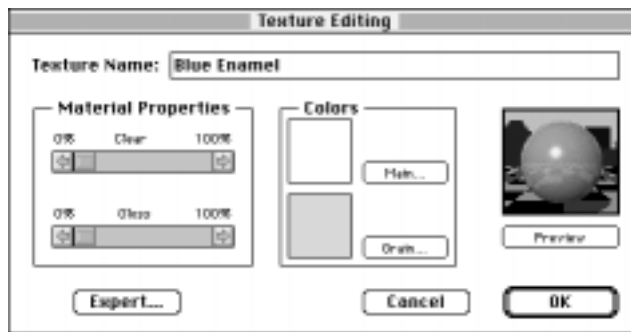
8.2



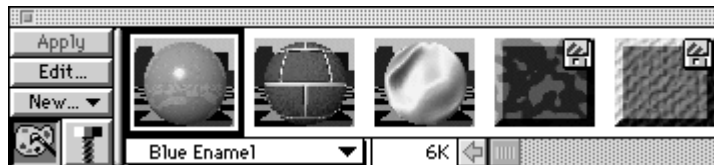
ENHANCEMENTS

You should note that using the sphere resources files may not produce a sphere object in the start-up Textures palette for all textures. This is because many textures are created with an imbedded tile preview. To change a tile preview displayed in the Textures palette to a sphere when one of the sphere resources files are loaded, simply click on the Textures palette Edit button with a texture selected. When the Texture Editing dialog appears, click the OK button to force the creation of a new, spherical preview in the palette. You do not have to change any of the settings in the dialog, nor do you need to click on the dialog's Preview button.

This illustration shows how a texture preview will appear in the Texture Editing dialog when the Preview button is clicked.



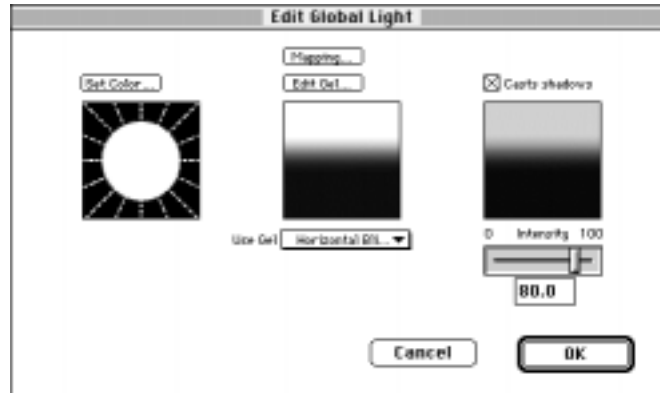
Sphere previews in the Textures palette must be generated through the Edit Texture dialog when the texture contains an imbedded tile preview.



When viewing Gels (accessible from the Lighting palette or when placing a new light Gels in the modeling window using one of the Tools palette light buttons), the different preview styles will have no affect on the Gel preview display. The tile and sphere previews will only be displayed in the Texture Editing dialog when the Edit Lighting dialog's Edit Gel button is clicked.



The Gel preview in the Edit Lighting dialog does not rely on a resource file to determine how the image is displayed.



Focal length resynchronizing

Vision3d now provides a means of resynchronizing a camera's focal length once it has been animated. This feature makes it easy to reset the focal length across a camera's entire animation path to that of the active frame with a single command.

To resynchronize the focal length of a selected camera simply determine the desired focal length in the active camera window and press Command-Option-R. Vision3d will automatically reset the focal length throughout the animation path. This eliminates the need to manually rescript the focal length of each event marker.

You should note that using this feature changes all scripted focal lengths of the selected camera by deleting information and replacing it. Therefore, you should save a copy of your model prior to using the command in case you change your mind once the operation is finished.

If you want to retain some of the focal length shifts along the camera's animation path, you should make a note of the specific event markers in which you want the shifts to occur and the focal settings you want to impose on those markers. You can then go back to each of the selected event markers after the resynchronization command and reset their focal lengths.

The focal length is adjusted by moving the slider at the top of the camera window.



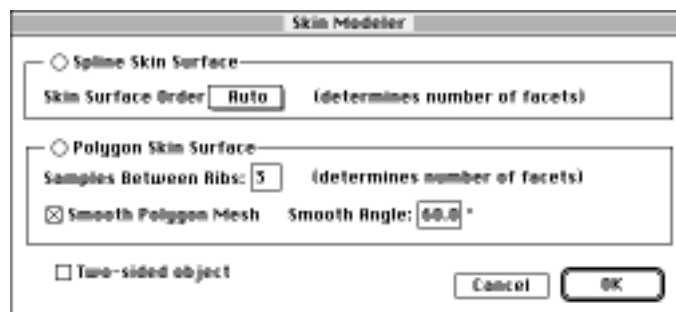
8.4



For more information on the use of camera focal lengths, please refer to Chapter 3 of the User Manual.

Skin modeler additions

Strata engineers have included the ability to select between creating a polygonal or spline (Bézier) object when using the Skin modeler. This flexibility gives you added control over FlexObjects™ created with the Skin modeling extension.



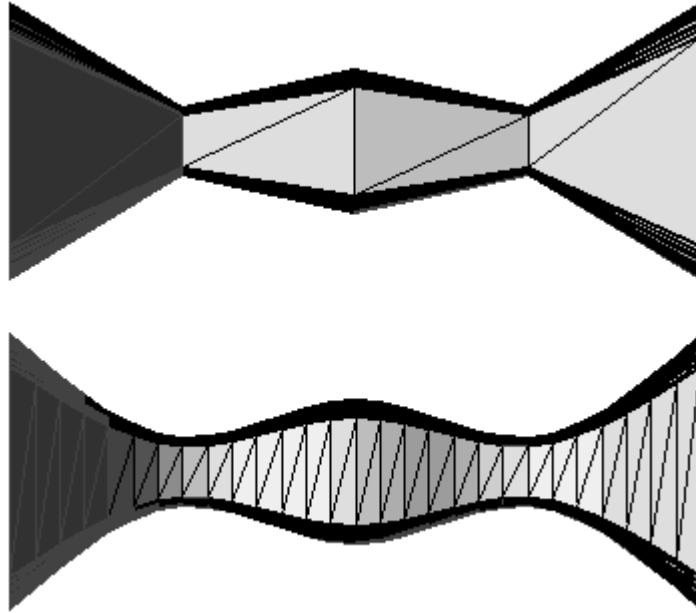
Spline-based FlexObjects are described fully in Chapter 3 of the Vision3d User Manual and most of the information regarding that format will also apply to polygonal-based FlexObjects.

Generally, choosing a specific type of skinned object is a matter of personal preference. There are few noticeable differences between spline and polygonal FlexObjects. However, there are some differences that should be noted.

Creating a polygonal skin object allows you to control the definition of a skin surface as it forms between ribs through the Samples Between Ribs field. Changing this setting can increase the smoothness of surface transformations between ribs. A setting of 0 (zero) in this field will result in no transitional breaks between ribs, which will produce a surface that is flat and straight between rib sections. Increasing this field raises the number of transition points between ribs to produce a smoother, more curved surface.



The top image of this polygonal FlexObject was created with the Samples Between Ribs field set at 0 (zero). The ribs used to create the bottom image are exact duplicates of the top, but the Samples Between Ribs field was set to 6.



The polygonal surface created by the Skin modeler is sometimes more predictable than a spline-based object, depending on the rib configuration. This is because there are often more reference points on a polygonal object, which allows the skin surface to more closely follow the actual rib substructure. You should note, however, that increased reference information usually means larger file sizes.

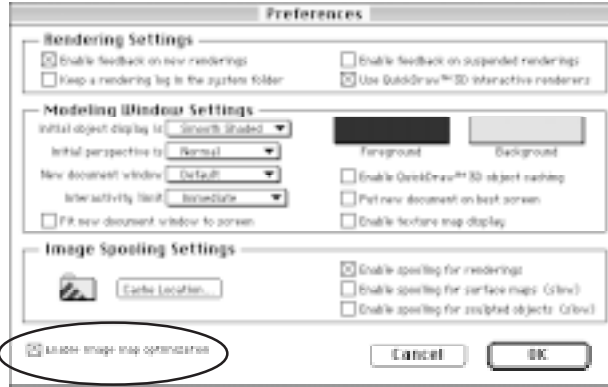
Map optimization

There is a checkbox in the Preferences dialog named Enable Map Optimization that enables a blanket image map optimization. This will happen for any new textures you may create in the current project, or when you open a previous model. Once maps in an existing model have been optimized, they remain in that form when saved. You may wish to save map-optimized models under a different name.



Optimization improves rendering speed by making all color maps in the model 32-bit color; grayscale maps are stored in 8-bit format. This allows the rendering algorithms to run more efficiently.

ENHANCEMENTS



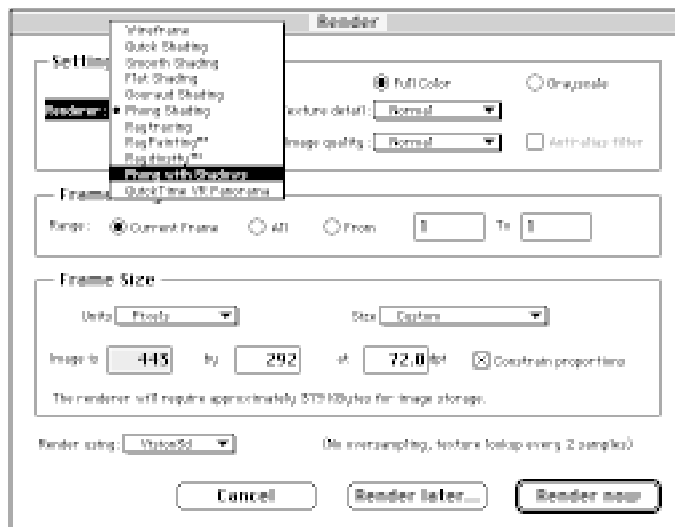
Note that when the checkbox for map optimization is enabled, the color depth pull-down menu in the Picture dialog (accessed when creating a new texture) is inoperable for individual maps you load into a texture, even though it still appears active. Every color texture map will be stored as a 32-bit image, and every non-color map will be stored as an 8-bit image.

You should understand that map optimization takes more memory. Therefore, you should turn this feature off if tight memory is a problem.

Phong with Shadows

The Phong with Shadows renderer produces a Phong shaded rendering with shadows.

Access the Phong with Shadows rendering option the same as all Vision3d rendering engines, through the Render dialog.





The Shadows checkbox in the Rendering Effects dialog must be checked for this renderer to work its magic. If you forget to check the Shadows option in the Rendering Effects dialog, the Phong with Shadows renderer will still render the image, but without shadows.

You should note the render uses the same technique of tracing light rays and producing shadows as the Raytracer renderer. As a result, there may be cases when the Phong with Shadows renderer will actually take longer to render an image than the Raytracer.

Unfortunately, there is no way to determine this prior to rendering the image.

Raydiosity

The Raydiosity renderer has been updated to allow the Glow Factor setting in the Expert Texture Editing dialog box to accept a number greater than 1.0.

The Glow Factor setting can be changed by clicking on the Expert button in the Texture Editing dialog.

Surface Properties	
Ambient Fraction	0.65
Bump Amplitude	1.0
Diffuse Fraction	0.2
Glow Factor	0.0
Reflectivity	0.28
Index of Refraction	1.0
Specular Fraction	1.0
Transparency	0.0
First Smoothness	20
Second Smoothness	3
Second Weight %	33

Some textures do not support the glow feature. In these cases, the Glow Factor field will not appear in the Expert Texture Editing dialog.

You should note that changing the Glow Factor setting to more than 1.0 may result in an exaggerated inter-object illumination.



These illustrations show the affect of varying Glow Factor settings. The texture used in these examples was Metal-brass.

ENHANCEMENTS



Glow Factor 0.0



Glow Factor 0.5



Glow Factor 2.0

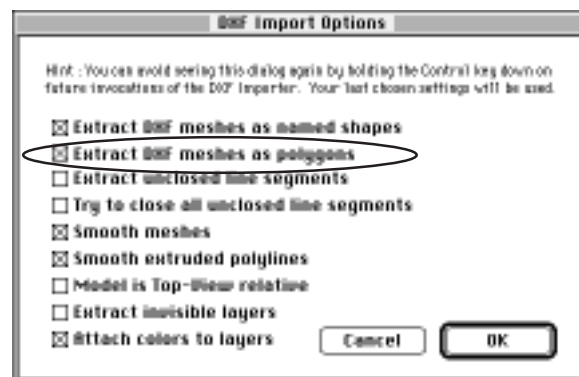
Keep in mind that Glow Factor settings, no matter how high, do not create a light source in your model. This option simply gives the illusion of a light source.

You should also note the affect of changing the Glow Factor setting will vary from one texture to another. Rendering a small sample portion of the image to ensure the setting you use achieves the results you want may save you valuable time in the final rendering.

For more information please refer to the Vision3d User Manual.

DXF file handling

Improvements have been made in the handling of files with very large meshes which are imported in the DXF file format. A new choice is offered in the DXF Import Options dialog:



The new checkbox is entitled 'Extract DXF meshes as polygons.' Check this box if you know you will be importing files with very large meshes.

If this box is not checked, and you try to import a DXF file containing an object with more than 32,000 vertices, this dialog will appear:



Press Continue to proceed with the conversion to polygons. If you press Stop, you will be unable to import the file.

Alpha channel enhancement

Improved handling of the alpha channel is also part of Vision3d 5.0. The alpha channel option appears on the Environment Effects dialog, which is reached through the Rendering pull-down menu. This option allows you to render an alpha channel mask of the background behind your scene, making the background transparent (See the User Manual).

The enhanced alpha channel will appear more jagged on your screen, because the smoothing of the edges of the image is not being littered with pixels from the background.

However, when you take the image into a program such as Adobe Photoshop™ for compositing, the image will actually be cleaner and work better than in previous versions of Vision3d.