

# LightWave™ 10



INTRODUCTION AND MODELER



# LightWave<sup>®</sup> 10

## Reference Manual

License.key number

Win32	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Win64	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
MAC	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	—	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

This publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose without prior written consent from NewTek, Inc.

© Copyright 2010, NewTek, Inc. All Rights Reserved

LightWave 3D™

NewTek  
5131 Beckwith Blvd  
San Antonio, TX 78249  
Tel.: 1-800-862-7837 - Fax: 210-370-8001  
[www.newtek.com](http://www.newtek.com)





## LightWave 3D® Software License and Limited Warranty

PLEASE READ CAREFULLY BEFORE INSTALLING AND/OR USING THIS SOFTWARE. BY INSTALLING AND/OR USING THIS SOFTWARE, YOU AGREE TO BECOME BOUND BY THE TERMS OF THIS LICENSE. IF YOU DO NOT AGREE TO THE TERMS OF THIS LICENSE, RETURN THIS PACKAGE TO THE PLACE WHERE YOU OBTAINED IT WITHIN 15 DAYS OF PURCHASE FOR A FULL REFUND.

### 1. Grant of License

The enclosed computer program(s) (the "Software") is licensed, not sold, to you by NewTek, Inc. (NEWTEK) for use only under the terms of this License, and NEWTEK reserves any rights not expressly granted to you. You own the disk(s) on which the Software is recorded or fixed, but the Software and all copyright rights therein, foreign and domestic, is owned by NEWTEK or its suppliers and is protected by United States copyright laws and international treaty provisions.

The copyright restrictions of this license extend to any further updates, software patches, or bug fixes made available to you by NEWTEK, whether distributed by floppy disc, CD ROM, DVD ROM or in an electronic format via BBS, ftp, email, etc.

This License allows you to use one copy of the Software on a single computer at a time. To "use" the Software means that the Software is either loaded in the temporary memory (i.e., RAM) of a computer, or installed on the permanent memory of a computer (i.e., hard disk, CD ROM, DVD ROM, etc.).

You may use at one time as many copies of the Software as you have licenses for. You may install the Software on a common storage device shared by multiple computers, provided that if you have more computers having access to the common storage device than the number of licensed copies of the Software, you must have some software mechanism which locks out any concurrent user in excess of the number of licensed copies of the Software (an additional license is not needed for the one copy of Software stored on the common storage device accessed by multiple computers).

You may make one copy of the Software in machine readable form solely for backup purposes. The Software is protected by copyright law. As an express condition of this License, you must reproduce on the backup copy the NEWTEK copyright notice in the following format "(c) 1990 - 2010 NEWTEK"

You may permanently transfer all your rights under this License to another party by providing such party all copies of the Software licensed under this License together with a copy of this License and all written materials accompanying the Software, provided that the other party reads and agrees to accept the terms and conditions of this License.

### 2. Restrictions

The Software contains trade secrets in its human perceivable form and, to protect them, YOU MAY NOT REVERSE ENGINEER, DECOMPILE, DISASSEMBLE, OTHERWISE REDUCE THE SOFTWARE TO ANY HUMAN PERCEIVABLE FORM. YOU MAY NOT MODIFY, ADAPT, TRANSLATE, RENT, LEASE, LOAN, RESELL FOR PROFIT, OR CREATE ANY MODIFICATIONS OR OTHER DERIVATIVE WORKS BASED UPON THE SOFTWARE OR ANY PART THEREOF.

### 3. Termination

This License is effective until terminated. This License will terminate immediately without notice from NEWTEK or judicial resolution if you fail to comply with any provision of this License. Upon such termination you must destroy the Software, all accompanying written materials and all copies thereof. You may also terminate this License at any time by destroying the Software, all accompanying written materials and all copies thereof.

### 4. Export Law Assurances

You agree that neither the Software nor any direct product thereof is being or will be shipped, transferred or re-exported, directly or indirectly, into any country prohibited by the United States Export Administration Act and the regulations thereunder or will be used for any purpose prohibited by the Act.

### 5. Limited Warranty and Disclaimer, Limitation of Remedies and Damages.

YOU ACKNOWLEDGE THAT THE SOFTWARE MAY NOT SATISFY ALL YOUR REQUIREMENTS OR BE FREE FROM DEFECTS. NEWTEK WARRANTS THE MEDIA ON WHICH THE SOFTWARE IS RECORDED TO BE FREE FROM DEFECTS IN MATERIALS AND WORKMANSHIP UNDER NORMAL USE FOR 90 DAYS FROM PURCHASE, BUT THE SOFTWARE AND ACCOMPANYING WRITTEN MATERIALS ARE LICENSED "AS IS." ALL IMPLIED WARRANTIES AND CONDITIONS (INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE) ARE DISCLAIMED. YOUR EXCLUSIVE REMEDY FOR BREACH OF WARRANTY WILL BE THE REPLACEMENT OF THE MEDIA OR REFUND OF THE PURCHASE PRICE. IN NO EVENT WILL NEWTEK OR ITS DEVELOPERS, DIRECTORS, OFFICERS, EMPLOYEES OR AFFILIATES BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INCIDENTAL OR INDIRECT DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, AND THE LIKE), WHETHER FORESEEABLE OR UNFORESEEABLE, ARISING OUT OF THE USE OR INABILITY TO USE THE SOFTWARE OR ACCOMPANYING WRITTEN MATERIALS, REGARDLESS OF THE BASIS OF THE CLAIM AND EVEN IF NEWTEK OR AN AUTHORIZED NEWTEK REPRESENTATIVE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

The above limitations will not apply in case of personal injury only where and to the extent that applicable law requires such liability. Because some jurisdictions do not allow the exclusion or limitation of implied warranties or liability for consequential or incidental damages, the above limitations may not apply to you.

### 6. General

This License will be construed under the laws of the State of Texas, except for that body of law dealing with conflicts of law. If any provision of this License shall be held by a court of competent jurisdiction to be contrary to law, that provision will be enforced to the maximum extent permissible and the remaining provisions of this License will remain in full force and effect.

### 7. Trademarks

NewTek, LightWave, LightWave 3D, LightWave CORE, LightWave 10, Viewport Preview Renderer, VPR, Virtual Studio Tools, TriCaster, TriCaster TCXD300, TriCaster TCXD850, 3PLAY, TriCaster PRO, TriCaster STUDIO, TriCaster BROADCAST, TriCaster DUO, SpeedEDIT, Virtual Set Editor, VSE, iVGA, LiveText, LiveSet, LiveMatte, TimeWarp, VT, [VTS], Video Toaster, Toaster, 3D Arsenal, Aura, are trademarks of NEWTEK. All other brand names, product names, or trademarks belong to their respective holders.

### 8. US Government Restricted Provision

If this Software was acquired by or on behalf of a unit or agency of the United States Government this provision applies. This Software:

- (a) Was developed at private expense, and no part of it was developed with government funds,
- (b) Is a trade secret of NEWTEK for all purposes of the Freedom of Information Act,
- (c) Is "commercial computer software" subject to limited utilization as provided in the contract between the vendor and the government entity, and
- (d) In all respects is proprietary data belonging solely to NEWTEK.

For units of the Department of Defense (DoD), this Software is sold only with "Restricted Rights" as that term is defined in the DoD Supplement to the Federal Acquisition Regulations, 52.227-7013 (c) (1) (ii).

Use, duplication or disclosure is subject to restrictions as set forth in subdivision (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at 52.227-7013. Manufacturer: NEWTEK, 5131 Beckwith Boulevard, San Antonio, TX 78249.

If this Software was acquired under a GSA Schedule, the US Government has agreed to refrain from changing or removing any insignia or lettering from the software or the accompanying written materials that are provided or from producing copies of manuals or disks (except one copy for backup purposes) and:

(e) Title to and ownership of this Software and documentation and any reproductions thereof shall remain with NEWTEK,

(f) Use of this Software and documentation shall be limited to the facility for which it is required, and,

(g) If use of the Software is discontinued to the installation specified in the purchase/delivery order and the US Government desires to use it at another location, it may do so by giving prior written notice to NEWTEK, specifying the type of computer and new location site. US Governmental personnel using this Software, other than under a DoD contract or GSA Schedule, are hereby on notice that use of this Software is subject to restrictions which are the same as or similar to those specified.

## Copyright and Trademarks

LightWave® 10

© Copyright 1990-2010 NewTek, Inc. All rights reserved.

Additional components Copyright: Steve Worley (1995-2010), Worley Labs (1995-2010), RainMaker (2003-2010) and Daisuke Ino (1995-2010)

FxMonkey™, HyperVoxels™, IntelligEntities™, Motion Mixer™, MultiMeshes™, Particle FX™, P.A.V.L.O.V.™, Skelegons™, SkyTracer 2™, Vmaps™ are trademarks of NewTek, Inc.

LightWave 3D is a registered trademark of NewTek, Inc. LightWave is a trademark of NewTek, Inc. All rights reserved. Windows and Windows Vista are registered trademarks of Microsoft in the United States and/or other countries. Intel and Intel Core are trademarks of Intel Corporation in the U.S. and/or other countries. AMD Athlon is a trademark of Advanced Micro Devices in the U.S. and/or other countries. Mac and Snow Leopard are registered trademarks of Apple Inc. NVIDIA and GeForce are trademarks of NVIDIA Corporation in the U.S. and other countries. All other trademarks are the property of their respective owners.



## Acknowledgements

---

### Engineering:

Chuck Baker, Deuce Bennett, Jamie Finch, David Forstenlechner, Matt Gorner, Mark Granger, Jonas Gustavsson, Bob Hood, David Ikeda, Antti Järvelä, Jarno van der Linden, Marvin Miller, Rob Powers, Prem Subrahmanyam, Jay Roth, Jon Tindall, David Vrba.

### Engineering Contributors:

Richard Brak, Andrew Cross, Josh Darling, Steve Forcucci, Art Howe, Peter Jespersen, Christopher Lutz, David Matthews, James Prentice, Mike Reed, Jarom Schow, William Vaughan, Scott Wheeler, James Willmott.

### Content Assembly:

Leigh Bamforth, Jarrod Davis, Lino Grandi, Luis Santos, Graham Toms, Gildardo Triana

### Installer:

Bob Hood, Dave Vrba

### Documentation:

Marvin Miller, Matt Gorner, Chuck Baker, William Vaughan

### Product Marketing:

Donetta Colboch, David Maldonado

### Special Thanks To:

Don Ballance, Steve Doubleday, Pat Grogan, Tim Jenison, Michael Kornet, Ralph Messana, Rex Olson, Jim Plant, Graham Toms, Andrew Yapple, Franck Lafage, Marc Gaillard, Hiroyoshi Abe, Ron Thornton, Jennifer Hachigian Jerrard, Dave Jerrard, Richard Morton, Reiner Shug, Lee Stringer and to all the Beta Team Members and HardCORE Participants

NewTek, Inc  
5131 Beckwith Blvd - San Antonio,  
TX 78249, USA  
1-800-862-7837 - +1-210-370-8000  
[www.newtek.com](http://www.newtek.com)

NewTek Europe  
Europarc - 17, Av. Léonard de Vinci  
33600 Pessac - FRANCE  
Tel.: +33 557 262 262 - Fax: +33 557 262 261  
[www.newtek-europe.com](http://www.newtek-europe.com)





# Table of Contents





## Contents

Copyright and Trademarks.....	II
Acknowledgements .....	III
.....	
tents .....	V

Table of Con-



# Chapter 1: LightWave 3D®

## Getting Started. . . . . 1

Introduction . . . . .	3
Installation. . . . .	4
System Requirements . . . . .	4
Hardware Lock Installation. . . . .	4
Upgrading From a Prior Version. . . . .	4
Software Installation . . . . .	4
Registering and Licensing Your Software . . . . .	5
Running the Program . . . . .	5
Conventions . . . . .	6
About the Manual. . . . .	6
Typographic Conventions . . . . .	6
Key LightWave Terms and Concepts. . . . .	7
Working With The Interface. . . . .	8
LightWave Panels and Dialogs. . . . .	9
Maths in Input Fields . . . . .	9
Enter/Tab Keys with Input Fields . . . . .	9
Yes and No . . . . .	9
Color Selection. . . . .	9
Color Pickers . . . . .	10
Standard List Windows . . . . .	10
Reorganizing Lists. . . . .	10
Context Pop-up Menus. . . . .	10
Customizing Your Interface . . . . .	11
Menus. . . . .	11
Keyboard Shortcuts . . . . .	12
Generic Plug-ins . . . . .	13
Configuration Files . . . . .	13
Custom Configuration Files . . . . .	13
Layout Startup Command . . . . .	14
Customizing Layout Viewport Navigation . . . . .	14
Customizing Camera Presets. . . . .	14
LightWave 3D Basics . . . . .	15
LightWave Scene and Object Statistics . . . . .	15
Measurement Units . . . . .	15
Filenames and Extensions . . . . .	15
Object Import/Export . . . . .	16
Image and Animation Types . . . . .	16
The Hub . . . . .	18
Command IPC . . . . .	19



Optimizing RAM Usage .....	19
LightWave 3D Resources .....	19
Internet Resources .....	19
NewTek Web and FTP Sites .....	19
Community.....	20
Technical Support.....	20
Customer Services .....	20

## Volume i ..... 21

## Modeler ..... 21

### Chapter 2: Introduction to Modeling . 23

Introduction to Modeling.....	24
Components of a 3D Object .....	24
Points, Polygons, and Edges .....	24
Editing Objects .....	25
Selection Mode: Points, Edges and Polygons .....	25
Interface.....	30
File Menu.....	33
Starting a New Object .....	33
Loading an Existing Object from Disk .....	33
Importing an Object .....	33
Saving Object Files.....	35
Closing Object Files .....	36
Sections .....	36
Exporting Objects.....	36
Quit .....	36
Edit Menu .....	37
Basic Point, Edge, and Polygon Maintenance .....	37
Undoing Operations .....	37
Redo .....	37
Customizing Keyboard Shortcuts .....	37
Edit Menu Layout .....	38
Edit Modeler Tools .....	39
General Options .....	39
Display Options .....	42
Windows Menu .....	52
Multi-Layer Object Standard.....	52
Multi-document Environment .....	52
Layout Communication .....	52



Layer Navigation .....	52
Layer Browser Panel .....	53
Vertex Map Panel .....	54
Hide Floating Windows On/Off .....	55
Hide Toolbar On/Off.....	55
Viewports .....	55
Save View Presets .....	56
Resetting Tools .....	56
Bottom Edge Menu .....	57
Info Display .....	57
Grid Info Field.....	57
Modes: Action Center Control.....	57
Numeric Panel .....	57
Make Command .....	58
Symmetry Mode .....	58
Surface: Change Surface .....	58
Vmap Shortcut.....	59
Help Menu.....	59

## Chapter 3: Create Tab ..... 61

Create Tab .....	62
Primitives.....	62
Text Tools.....	82
Text Layers.....	87
Points .....	88
Polygons .....	91
Curves.....	97

## Chapter 4: Modify Tab ..... 105

Modify Tab .....	106
Translate.....	109
Rotate .....	117
Transform.....	124





## Chapter 5: Multiply Tab . . . . . 139

Multiply Tab . . . . .	140
Extend . . . . .	140
Duplicate . . . . .	
Subdivide . . . . .	175

165

## Chapter 6: Construct Tab . . . . 189

Construct Tab . . . . .	190
Reduce . . . . .	190
Remove . . . . .	193
Combine . . . . .	195
Patches . . . . .	202
Convert . . . . .	203
Meta . . . . .	205

## Chapter 7: Detail Tab . . . . . 207

Detail Tab . . . . .	208
Points . . . . .	208
Polygons . . . . .	211
Unify Normals . . . . .	215
Curves . . . . .	216
Edges . . . . .	217
Measurement . . . . .	219
New Selection Options . . . . .	222

## Chapter 8: Map Tab . . . . . 223

Map Tab . . . . .	224
Introduction . . . . .	224
General . . . . .	227
Weight . . . . .	233
Color . . . . .	237
Texture . . . . .	263
Poly Map . . . . .	270
Point Map . . . . .	272
Morph . . . . .	280



## Chapter 9: Setup Tab . . . . . 285

Setup Tab . . . . .	286
Skelegons . . . . .	286
Layout Tools . . . . .	292

## Chapter 10: Utilities Tab . . . . . 295

Utilities Tab . . . . .	296
Commands . . . . .	296
LScript . . . . .	298
Plugins . . . . .	299

## Chapter 11: Selection Tab . . . 301

Selection . . . . .	302
Selection Sets . . . . .	308

## Chapter 12: View Tab . . . . . 311

View Tab . . . . .	312
Viewports . . . . .	312
Layers . . . . .	316
Selection . . . . .	319
Selection Sets . . . . .	325
View . . . . .	327

Glossary . . . . .	i
Glossary of 3D Terms . . . . .	ii





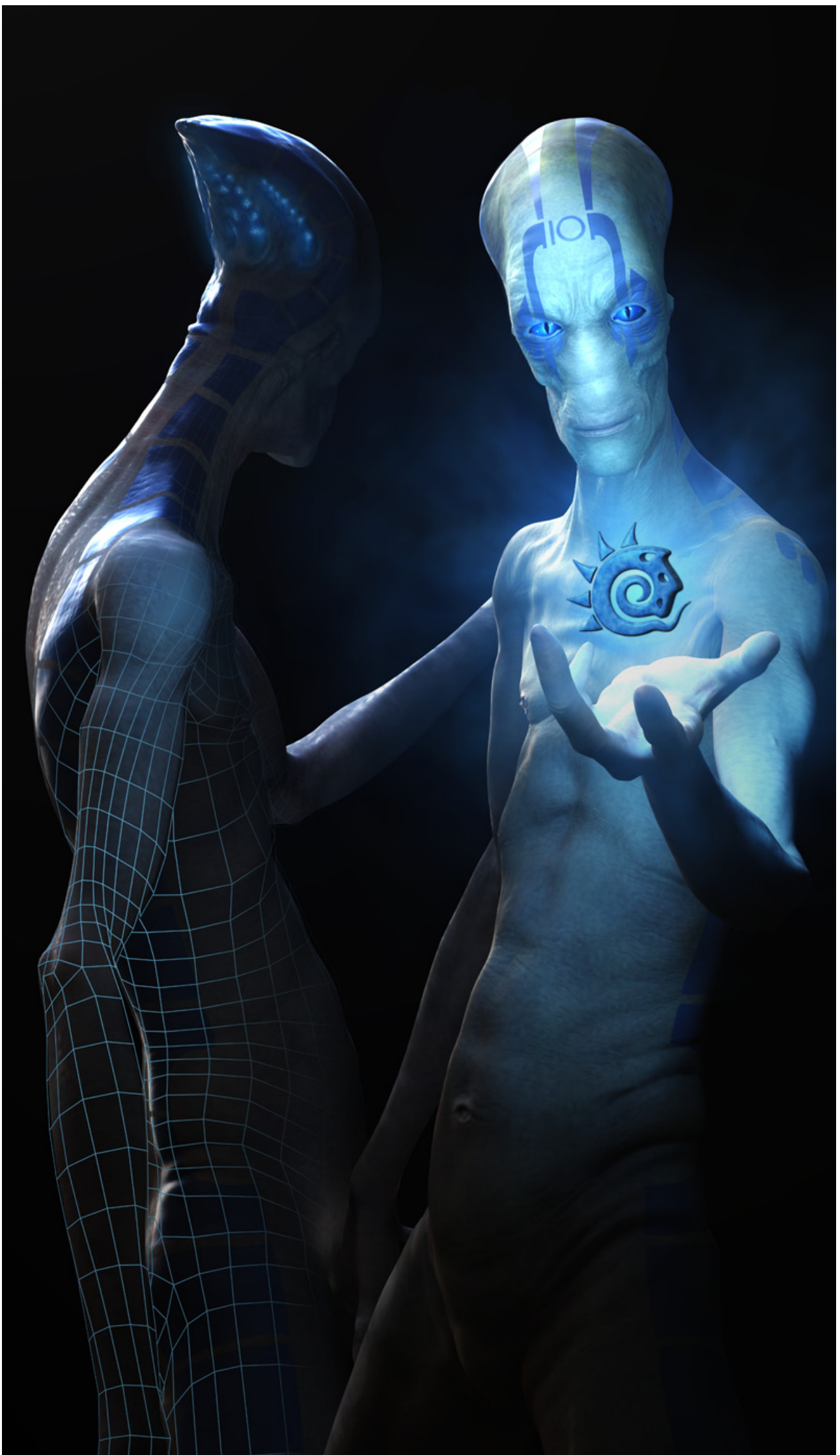
---

# Chapter 1: LightWave 3D<sup>®</sup>

## Getting Started

---







---

## Introduction

---

NewTek LightWave™ 10. Count on it.

Congratulations on your LightWave purchase.

NewTek LightWave 3D® is the most complete 3D solution to model, render and animate. Used extensively in television and film production, video game development, print graphics and design, LightWave artists have won more Emmy® Awards for visual effects and animation than any other CG artists. Plus, LightWave is designed to be used by a single artist, a small team, or a major facility, with seamless integration into diverse pipelines.

LightWave™ 10 gives you the workflow and functionality you've heard about, along with new, interactive tools for a real-time experience. You get the speed, flexibility and control you need to create quickly, easily and affordably.

This manual will help you learn the basics of 3D with LightWave — from installation and operation, to accessing and using the essential tools to model, render and animate. Once you've installed the software, please download some of our tutorials to learn the landscape. You can also join the online forums, where a passionate community of LightWave users from all over the world, help each other to elevate their craft. Our 3D team also provides valuable tips and tricks for you to increase your knowledge and expertise, so visit our website often for the latest information at: [www.newtek.com/lightwave](http://www.newtek.com/lightwave).

Welcome to the NewTek family.

# Count on it!



## Installation

### System Requirements

#### General Requirements:

Hardware (Windows):

##### 64-bit

Intel® Core™ 2 or better or AMD64.

System RAM: 2GB minimum

##### 32-bit

Intel or AMD equivalent of Intel Core or better

Hardware (Macintosh):

Intel processor

Operating System (Windows):

Windows Vista or Windows 7

Operating System (Macintosh):

Snow Leopard® 10.6 or better

Graphics Card

NVIDIA® GeForce® 8400 series (minimum) or ATI X1600 (minimum)

### Hardware Lock Installation

If this is your first installation of LightWave 3D, this package came with a USB hardware lock ("dongle"). Just plug this into an available USB port on your PC or Mac to unlock the software. You can plug the dongle in while your machine is running.

### Upgrading From a Prior Version

If you already have a version of LightWave 3D installed on your computer, you will already have a hardware key (USB, ADB or parallel), also known as a "dongle". The software will recognise your existing USB or parallel dongle. If you have an ADB dongle, you will need to contact our customer services section to exchange for a USB dongle. Users of a parallel dongle who wish to upgrade to the convenience and platform independence of the USB dongle may contact customer services to make the exchange. Please note that there is a fee and shipping and handling charges to cover the cost of the exchange. ADB dongles will not work under Macintosh OSX and will need to be replaced with USB dongles. Contact information for the appropriate Customer Service office is found at the end of this chapter, please contact them for more information..

Other than the above noted issues regarding hardware keys, the same installation instructions apply for an upgrade as for new users. The installation program will not delete any existing objects, scenes, or images on your system unless an existing item has the same name and is in the same directory as an item that LightWave 10 installs. For this reason, you should install LightWave 10 in a new directory. We strongly recommend that you back up your system before you install LightWave 3D.



**WARNING:** NewTek strongly recommends that you backup your system before you install LightWave 3D.

### Software Installation

To install LightWave 3D on your system, insert the LightWave 3D Program Disc DVD. The installer should automatically start up. If it doesn't, open the DVD and run setup.exe on the PC, or the LightWave installer program on the Mac. Both platform installers are on the same DVD.

#### Windows Vista or 7:

Step 1: If this is a new seat of LightWave being installed for the first time, attach the supplied USB dongle. Please note that if Sentinel drivers for the dongle are not present, then during the installation you will be asked to remove the dongle while drivers are installed.

Step 2: Insert the LightWave 3D Program DVD.

Step 3: The installer will inform you if the Sentinel Drivers must be installed.

Step 4: There will also be a pop-up message about severe problems if the dongle is not removed and the machine is not restarted before installing LightWave 3D.

DO NOT click to close this pop-up yet.

Step 5: When the installer starts the procedure for the dongle driver install, click on 'Complete', not 'Custom'. Let it install the drivers for both the parallel and USB dongles, even if you will only be using the USB dongle.

Step 6: When you are told to do so, remove the dongle from the machine.

Step 7: Allow the drivers to be installed.

Step 8: When done, you will get another message about restarting your computer.

Step 9: Click to close the message from step 4.

Step 10: Click to restart the machine.

Step 11: When Windows has completed rebooting, and you see the desktop, reinsert the dongle.

Step 12: When the 'New Hardware Found' Wizard comes up, let it run.

Step 13: If it asks you which driver to use, select 'automatic', and there should only be one driver on C: (or whatever drive you are installing to). Select it.

Step 14: Re-activate the LightWave 3D installer.



Step 15: Install the software.

Step 16: Follow the on-screen steps to register LightWave 3D.

Step 17: The LightWave program icons will be located in the NewTek LightWave 3D folder in the Start > Programs menu.

### Mac OS X (10.6.1 or higher):

Step 1: If this is a new seat of LightWave being installed for the first time, attach the supplied USB dongle.

Step 2: Insert the LightWave 3D Program DVD.

Step 3: Double-click the install icon.

Step 4: Install the software.

Step 5: Follow the on-screen steps to register LightWave 3D.

Step 6: The LightWave program icons will be located in the LightWave folder in the Applications folder.

Step 7: Drag LightWave and Modeler to the Dock for easy access, or create aliases by Option-Dragging the icons to the Desktop

## Registering and Licensing Your Software

Initially, LightWave will create a temporary license key that will allow the application to run in full mode for a certain number of days before you must register the software. After the temporary key expires, LightWave will run only in Demo mode until the permanent license key is installed. Registration is quick and easy, and can be completed 24 hours a day, 7 days a week on our web site. It is important that you register your software — not just to unlock it — as registration allows you to receive free technical support, upgrades, special offers, and new product information.

LightWave 3D includes a registration wizard that automates the process of registering the software, and installing the permanent license key code, if you have installed LightWave on an Internet-connected computer system or have access to such a system.

To register, you will need your Hardware Lock ID and your Serial Number. To get your Hardware Lock ID, just run the Registration Wizard or run LightWave. A dialog box will appear that tells you this information. Your Serial Number is located either on the bottom of your box or inside your CD case, and contains ten digits. We recommend that you record your serial number in the manual, as well.

You can get your permanent license key from NewTek Europe using the Registration Wizard, alternatively you can call NewTek Europe on +33 557 262 262 and register over the phone. NewTek Europe will email you your permanent license key within 48 hours Monday to Friday.

If you don't have Internet access at all you can send a letter containing your address details, LightWave serial number, Product Lock ID, reseller you bought LightWave from and what sort of computer you are using LightWave on to:

NewTek  
Attn: Customer Service  
5131 Beckwith Blvd  
San Antonio, TX 78249

## Running the Program

LightWave 3D is started the same way as any other program on the particular platform on which it's installed. LightWave 3D has two major applications that work together: Layout and Modeler. Each program can be started independently; however, you may also access them by clicking the appropriate button on their respective interfaces.





## Conventions

### About the Manual

LightWave 3D is generally divided into two separate environments: LightWave Layout and LightWave Modeler. (It is a common convention to refer to the animation and rendering toolset as *LightWave*, or just *Layout*, and to refer to the object creation toolset as *Modeler*.)

For Layout, the buttons on the toolbar are often shortcuts to controls that can also be accessed from one of Layout's many panels. The Layout section of the manual begins with a panel and topic-based approach to explaining Layout, since many controls for certain tasks are grouped together on panels, and continues with a breakup of the tools into a workflow based layout.

Modeler is a little different. All of its tools are accessed from the toolbar, and panels are used to execute more focused tasks. Therefore, the Modeler portion of the manual is generally organized by the order of tools in the toolbar.

### Typographic Conventions

The following conventions will be used throughout this manual.

### Directory Structure

Except for a few system files, all of the software will be installed in a single subdirectory on the drive you specified during the installation process. Therefore, subdirectories referred to in this documentation are located in this main subdirectory. (For example, if the discussion refers to the Images drawer, the actual path might be C:\LIGHTWAVE\IMAGES.)

### Typefaces

ALL CAPS	Computer keys, directories, device names (e.g., ENTER, C:\LIGHTWAVE\OBJECTS, CTRL+P, etc.).
Bold	Names of interface menus, fields, buttons, etc. are set in bold type.

### Keystroke Combinations

KEY1 + KEY2	Simultaneous keystrokes. Hold the first key and press the second key.
-------------	---

### Mouse Operations

LMB	Left mouse button
MMB	Middle mouse button (if applicable)
RMB	Right mouse button
Selecting	Single-clicking an element with the LMB so that it becomes active or selected.
Deselecting	Single-clicking an element with the LMB so that it becomes inactive or unselected.

Activating	Selecting an option by clicking on its toggle button.
Deactivating	Unselecting an option by clicking on its toggle button.
Clicking	Placing your mouse pointer over something and then pressing a mouse button. This nearly always means the LMB.
Right-clicking	Clicking an element with your RMB.
Double-clicking	Rapidly clicking an element twice.
Dragging	Selecting an element with your mouse pointer and continuing to hold the mouse button down as you move your mouse. This nearly always means with the LMB.



NOTE: Important Information for Macintosh users: For maximum functionality, a two or three button mouse is recommended with LightWave. Using a single button mouse, a RMB action can be applied by holding the COMMAND (Apple) key while pressing the mouse button.

### Symbols



WARNING: The warning symbol will highlight a discussion that warns you about something. You should pay special attention to text marked with this symbol.



NOTE: The note symbol will highlight a discussion that is particularly noteworthy.



HINT: The hint symbol will highlight tips and suggestions that are usually of a time-saving nature.



FUN FACT: Interesting information that will not necessarily make you smarter with LightWave, but will allow you to amaze and amuse your friends with your inside knowledge of 3D industry trivia.



## Key LightWave Terms and Concepts

In the Appendices you will find an extensive Glossary of terms used in 3D graphics. The following are a brief selection of some common terms and concepts used with LightWave that will help you to get started. These are also covered in the Glossary, in some cases at greater length.

**Animation Channel** — Animation Channel refers to the different Position, Rotation, and Scaling settings an item can have in Layout. It can also refer to other *envelopeable* elements like light intensity. See also *motion channel*.

**Axis** — Axis refers to the XYZ coordinates used as the basis for positioning elements in LightWave's 3D space. It is somewhat like the concept of left/right, up/down, and near/far.

**Bone** — For any object, you can define a skeletal system composed of bones. By moving bones around, you can change the shape of an object.

**Bounding Box** — A Bounding Box is a six-sided box conforming to the outer dimensions of an object. It is commonly used as a quickly drawn stand-in for a more complex object.

**Camera** — As in real life, a camera records events. Looking through LightWave's camera shows you the view as it will be generated.

**Control Mesh** — A Control Mesh is a cage of points used to shape SubPatches.

**Endomorph** — An Endomorph is an object containing one or more Morph Maps.

**Envelope** — An envelope is a way of setting a particular value that usually changes over time using a graphical input mode.

**Frames** — LightWave works with frames. A frame is one image out of many that define an animation. There are 25 frames per second in PAL video, approximately 30 frames per second in NTSC video, and 24 frames per second in film.

**Geometry** — Geometry refers to the positional layout of points and polygons that define an object.

**Item** — An item in Layout refers to an object, bone, light, or camera.

**Keyframe** (Also known as just a *key*) — A keyframe is a frame for which you define a value for an animation channel(s) (e.g., position or rotation) for an item in Layout. Animations are composed of a beginning keyframe, an ending keyframe and usually some number of keyframes in between. See also *tween*, in the Glossary.

**Light** — A light in LightWave is generally used just like a light in real life. Lights illuminate a scene and allow you to see the objects within it.

**LScript** — This is LightWave's built-in scripting language. LScripts can be installed and used just like plugins.

**Motion Channel** — Motion Channel is generally the same as *Animation Channel*, but refers only to position, rotation, and scale. (i.e., not light intensity.)

**Normal** — A *polygon normal* is the imaginary line projecting out perpendicular to a surface at any point indicating the direction of the polygon. A polygon surface normal is represented as dashed lines on selected polygons in Modeler. LightWave sees polygons or faces of an object only from the surface normal side. A single-sided polygon (like a piece of paper) with its normal facing away from the camera will be invisible to the camera from that viewpoint (unless the surface is using the Double Sided option). A *vertex normal's* direction is the average of the polygon normals it is connected to.

**Null Object** — A null object is an object that contains no geometry and will not show up in a rendered image. Nulls are useful for tracking, grouping (parenting), and using with inverse kinematics functions. They are also commonly used to control plug-ins.

**Object** — An object is composed of points and faces. Points connected together to form a polygon define a face. Faces joined together form an object.

**Parenting** — Parenting refers to LightWave's ability to set hierarchical associations between items in a Scene. Generally, the parent item will have some level of influence on its *child*, whether it is position, rotation, size, etc. The child can also be a parent to another item.

**Particles** — Usually refers to animated points used for sparks, fire, and even liquid. LightWave's particle engine is called Particle FX.

**Pivot Point** — The position in an object that acts as the center of rotation and position reference.

**Pixel** — A pixel is the smallest unit of measurement in an image and is used to describe the image's width and height.

**Plane** — A plane refers to a two-dimensional (i.e., flat and level) surface. You might want to think of a plane as a piece of glass that is infinitely large, but has no depth.

**Plug-in** — A plug-in is a program that works with and extends the functionality of LightWave.

**Render** — Rendering is the computer's process of calculating and generating an image based on the values you have selected for the different options in LightWave.

**Scene** — A Scene is a LightWave project defining the objects loaded and their motions, the number of lights and their values/motions, the resolution of the final image, special effects, Camera settings, and so on. This ASCII text file is generally saved from Layout.

**Session** — A session is a single use of an application. A session starts when you first boot the application and ends when you exit.

**Spline (Curves)** — LightWave uses splines or curved paths between keys while moving items about. When modeling, splines refer to open or closed curves.

**SubPatch** — SubPatch is a modeling mode where polygons become a cage that controls an underlying mesh of NURBS.



**Surface** — Essentially, the surface is the skin of an object. A single object can have multiple surface names, each with its own independent attributes (e.g., color), and multiple objects can share the same surface name(s).

**Texture** — A texture is an attribute of a surface, or an entire object, in the case of displacement and clip maps.

**Vertex** — A vertex is the point at which the sides of a polygon intersect.

## Working With The Interface

In no time, you'll find yourself mastering LightWave's intuitive interface. LightWave uses very few graphical icons. Instead, most functions are listed on the interface panels in plain text. Listed below are a few other conventions involved with the LightWave interface:

**Button** — A button refers to an area on the screen that you click on with your mouse to cause some function to occur. Generally, only a single click is required. There are also special types of buttons, like toggle, pop-up menu, and envelope buttons. Some buttons become highlighted, indicating a chosen or active status.

**Contextual Pop-ups** — This refers to *context-sensitive* pop-up menus. Such menus appear when you SHIFT+CTRL+LMB and are aware of the area the mouse pointer is over.

**Dialog** — See *Requester*

**Drag Button** — A drag button is similar in effect to a slider, except the button does not move. To use, just click on it and hold the mouse button, then drag the mouse. Depending on the parameter being adjusted, the direction and mouse button used will have different specific effects.

**Envelope Button** — An Envelope button is a small button marked with an E. Selecting this button will display the Graph Editor where you can create an envelope for the setting. A highlighted envelope button indicates a value has an envelope applied. To turn off (i.e., remove) an envelope, hold the SHIFT key and click on the Envelope button.

**Ghosted Item** — LightWave will *ghost out* parameters that are not available to you. This is usually the result of a certain option not being activated. Selecting a ghosted item will display a message informing you why it cannot be used.

**Information Field/Display** — Information Fields/Displays are text displays found throughout the different panels. These displays cannot be changed directly and simply provide

information and feedback.

**Input Field** — Input fields are areas on the screen where you can enter data.

**Mini-Slider** — A mini-slider is a button with two arrows on it, pointing right and left. Clicking on one and holding the LMB, then dragging to the right or left will raise or lower the value of the parameter next to it. In many cases, however, the slider does not encompass the entire spectrum of possible values.

**Panels** — Panels are any one of the windows that open when you click on a button in LightWave. Many panels have additional tabbed sections that are selected when you click on a tab. Panels may be Modal or Non-modal. A modal panel must be closed before you can continue working with the rest of the application. A non-modal panel lets you shift the focus between it and another part of the application without having to close the panel—you can continue to work elsewhere in the current application while the panel is displayed. Modeler's Numeric Panel is non-modal because you can do other things while it is open. In contrast, Modeler's Display Options Panel is modal because you must close it before you can continue working.

**Pop-up Menu** — Pop-up menu buttons have a downward facing arrow on their right edge. To use, click on it and hold the LMB. The menu will pop up and as you move your pointer over the menu, each item will become highlighted. When the desired selection is highlighted, release the LMB. If you decide not to select an item, simply move the pointer off the menu and release the LMB. Certain pop-up menus contain lists of objects, images, and lights, which are normally listed in the order they were loaded or created.

**Requester** — A requester is also known as a dialog box. These appear on the screen for operations like file loading and saving. This also refers to smaller windows that appear requesting you to input data into various fields.

**Reset Area** — The reset area is a non-active open area (e.g., not a button) on the Modeler toolbar that acts as a reset button, much like you might use the ESC key on other applications.

**Scrollbar** — See *Slider*

**Shaded Display** — See *Solid-Shaded Display*.

**Slider** — A slider allows you to modify a setting by dragging the slider's button along the bar. Alternatively, you can click to the right or left of the button or use the arrow buttons at either end to incrementally change the setting value. (Also known as a scrollbar.)

**Solid Shaded Display** — Solid shaded display refers to a non-wireframe display mode where some level of surface texture detail is visible.

**Texture Button** — A Texture button is a small button marked with a T. Selecting this button will present a texture panel allowing you to define a texture for the chosen parameter. A highlighted texture button indicates a texture in use for the given parameter. To turn off (i.e., remove) a texture, hold the SHIFT key and click on the Texture button.

**Toggle Button** — A Toggle button is a small button that becomes highlighted with a check mark when clicked. This indicates the adjacent feature is active.



## LightWave Panels and Dialogs

LightWave's panels and requesters are generally non-modal and can be left open even while you interact, say, with the Layout window and its controls.



NOTE: LightWave will attempt to display the selected color to the best of your display's capabilities. This will generally result in an approximation of the color if you are using less than a 24-bit display mode.

## Maths in Input Fields

Numeric input fields support basic maths operations (+, -, \*, /, and ^) and algebraic equations. You can even mix units. Here is an example: (12ft+14m)^2+(3/5). The result, 312.3908 m, would appear after you press the Enter key. Note that the result is shown using the default unit; here, it is meters. Also, since we didn't specify the unit of measure for the 3 and 5, the default is used.

## Enter/Tab Keys with Input Fields

When you enter values into dialogs that have multiple input fields, the Tab and Enter keys have special functions to save you time. Pressing the Tab key stores the entered/current value and automatically advances you to the next input field. Shift + Tab moves to the previous field. The Enter key also stores value, dismisses the dialog, if appropriate.

## Yes and No

Whenever a requester asks you a Yes or No question, pressing the Enter key is the same as selecting Yes, while pressing the Esc key is the same as selecting No. Likewise, an Error, Warning or Reminder requester with a choice of Continue or Cancel can use Esc for Cancel and Enter for Continue. For those requesters with only a Continue, pressing either Esc or Enter will close the panel.

## Color Selection

Where you must select a certain color (e.g., surface color), there will be a three-number color component display and a color swatch box. You can also drag your mouse on each color component to change its value. By default, the standard RGB color space is displayed; however, if you right-click on the display, you may also use the Hue, Saturation, and Value (HSV) color space, if that is more familiar to you. HSV is based on the artist's concepts of tint, shade, and tone. There are over 16.7 million possible color combinations.



Top: RGB, Bottom: HSV

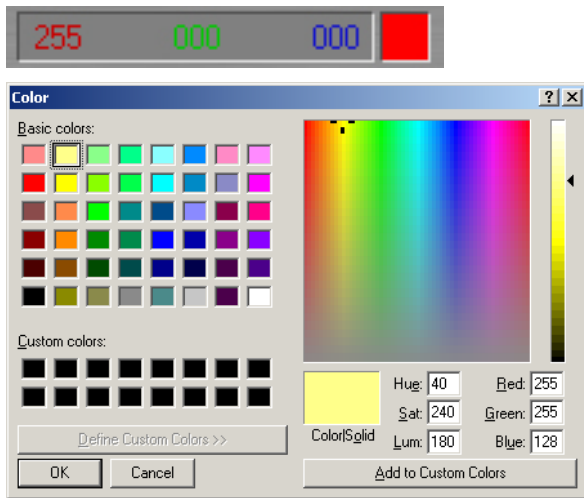


NOTE: Unlike RGB, HSV color components are not independent. For example, if Value is 0 (i.e., brightness is 0), you will not be able to change Hue (color) nor Saturation. Similarly, if Saturation is 0, you will not be able to change Hue. Moreover, reducing one component may cause another component to decrease.



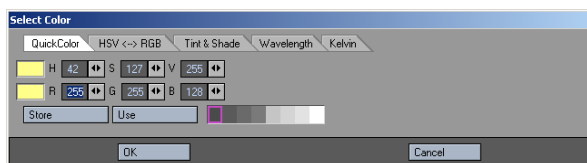
## Color Pickers

Clicking the color swatch box will open a color selection requester that is standard for your specific platform or the built-in color picker.



Windows Color Picker

The LW\_ColrPikr color picker is a color selector with more features, such as a wavelength and temperature-guided layout of color. Select this on the General Options Tab of the Preferences Panel (Edit > General Options).



LightWave Color Picker

## Standard List Windows

Windows that contain lists of items (surfaces, scene items, menu items, etc.) all have a slider bar to scroll through the list. If the list is hierarchical in nature, there will be arrowheads to the left of item names. If the arrowhead is facing to the right, then they are subordinate items to this *parent* item. Clicking the arrowhead will make it point downwards and also reveal the subordinate items.



Left: All lists are collapsed, Right: Camera Entry is exposed

If selection is appropriate, you can click on an item to select it. Ctrl + click will select non-contiguous items and Shift + Click will select a range of items.

List windows that are hierarchical in nature can be reorganised by dragging items around in the list. Where appropriate, multiple selection is supported.

## Reorganizing Lists

In some list windows (e.g., menu configuration), you can reorganize entries by just dragging them in the window. Place your pointer over the entry and drag it up or down. You will see a thin line appear as you drag. This is the insert point. Releasing the mouse button will drop the entry here.

Now, if your pointer is at a position where the dropped item could be, say, a sub-item of an entry, the same level as the entry, or even the same level as the next entry, the length of the line would change as you drag. The position of the left end of the line determines the level of the drop.

## Context Pop-up Menus

Both Layout and Modeler have special “context” pop-up menus that you display by holding Ctrl + Shift and clicking your LMB, MMB, or RMB with your mouse pointer over a viewport — there are different menus for each mouse button.





## Customizing Your Interface



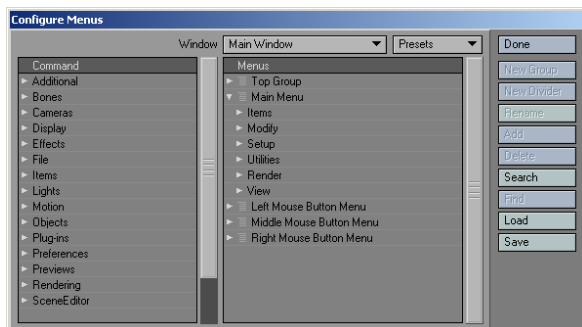
**WARNING:** We strongly suggest that you keep the default menu organisation intact. Otherwise, technical support and using the documentation may become difficult. A better solution is to create a new menu tab and/or groups and place your frequently used tools in them.

## Menus

LightWave menus are customisable. You can add, remove, group, and reorganise commands.

Choose Edit > Edit Menu Layout to open the Configure Menus Panel. In the (left) Command window, a list of available commands will appear. These will be grouped by type.

In the (right) Menus window, several main sections have sub-items, which are indented to show the hierarchical relationships. Top Menu Group contains the items that are always visible no matter which tab is selected. Main Menu items are the main tabs and related buttons for the main interface toolbar. Bottom Edge is for Modeler only and relates to the controls along the bottom edge of the screen. Left, Middle, and Right Mouse Button Menu (when available) are the menus that appear when the Shift + Ctrl keys are held down along with the corresponding mouse button. There may be other menu sections defined.



In the Menus window, dots indicate commands while arrows indicate groups of commands/sub-groups beneath it. If the arrow points to the right, the group's sub-menu items are collapsed and not visible. To reveal the sub-menu items for a collapsed group, simply click on its right-facing arrow. To collapse an open group, click on its downward-facing arrow.

A set of Studio Production menus has been included. These menus are recommended for more advanced users.

## First-Level Menu Items

Menu items on the first-level must always be a group. Commands and sub-groups may exist on the second menu level. On the toolbar, a Main Menu group's name will appear as a heading above the grouped command buttons. If the group name is blank, no heading will appear. On the next menu level, commands or sub-groups can also be used. If a sub-group is used, it will appear as a pop-up menu on the interface.

The hierarchy would look like this:

Section Type (e.g., Main Menu)

Group (Group name appears on row of horizontal tabs on the main interface)

Command/Sub-group (If a group, its name, if any, appears as a header above the next group of commands)

Command/Sub-group (If a group, its name appears on pop-up button)

## Adding New Groups

If you select a collapsed group when you click the New Group button, it will be added at the same level as the group. If you select an un-collapsed group when you click the New Group button, it will be added inside the group.

## Divider Line

Clicking the New Divider button will add a dashed line below the selected menu item. This is displayed as a dividing line between menu items when it appears in a pop-up menu group.

## Renaming Menu Items

You can rename any menu item (including commands) by selecting it with your mouse and then clicking the Rename button.



**NOTE:** The command name is not always what is displayed on a button. If the command uses a shorter name, the button may use the exact or a similar name. However, the button name can be very different in some cases. Of course, you can always rename the displayed name.

### Automatic "More" buttons

If an interface (e.g., Layout or Modeler) is sized too small to fit all Toolbar buttons, a More pop-up button will appear at the end of menu groups and contain the hidden buttons. Note that sometimes the menu names can be different depending on whether they are listed on the toolbar or listed in a pop-up menu.

## Adding Commands

To add a command, select it in the (left) Command window and select the target position in the (right) Menus window. Then click the Add button. The command will be added under the selected target — inside the group if a group was selected. Alternatively, you may drag the command to the Menus window.



## Reorganizing Menus

You can reorganise entries by just dragging them in the window. Place your pointer over the entry and drag it up or down. You will see a thin line appear as you drag. This is the insert point. Releasing the mouse button will drop the entry here.

Now, if your pointer is at a position where the dropped item could be, say, a sub-item of an entry, the same level as the entry, or even the same level as the next entry, you will see the line change its length as you drag. The position of the left end of the line determines the level of the drop.



**HINT:** To place the group/command at a specific point in a group, make sure the group is uncollapsed before you begin dragging.

## Default Menu Locations for Plug-in Commands

The plug-ins that come with LightWave have default (Layout or Modeler) menu locations for their commands. However, when you add plug-ins, the commands will be placed in their default menu locations only if you are using the default menu. (You are using the default menu, if you have never customised the menu configuration or have chosen Default from the Presets pop-up menu.)

Modeler places commands without default menu locations in the Construct > Utility: Additional pop-up menu. For Layout, you must manually assign such commands to menus.

## Finding Assignments and Commands

If you have a command selected in the left window and click Find, the command, if any, on the right will become selected. Selecting a command in the right window and clicking Find will select the matching command in the left window.

Since a command can be assigned to more than one menu location, successive Find clicks will show the next assignment, if more than one exists.

Clicking Search allows you to type in a search phrase to find a command. Note that this will only find the first instance and it is case sensitive.



**HINT:** If you know the keyboard shortcut, but want to find the corresponding menu location, use the Find function on the Configure Keys Panel to determine the command name. Then, use the Find function on the Configure Menus Panel to see where it is assigned.

## Deleting Menu Items

To delete a command/group, select it and click the Delete button.

## Window Pop-up

The Window pop-up menu lets you select different menus to edit (e.g., Graph Editor), if available.

## Maintaining Menu Sets

The Load and Save buttons let you retrieve and store menu sets that you develop. To restore the default menu configuration, choose Default from the Presets pop-up menu. If you'd like to have a menu set for Layout similar to LightWave 7.5, choose 7.5 Style. You can also use LightWave 6.0 menus or even 5.6. In Modeler you have the choice of 7.0 or 6.0.

## Arranging Menu Tabs

You can reorganise the main menu tabs by simply dragging them left or right on the main interface. Before you release the mouse button you will see an insertion point marker appear.

## Middle Mouse Button Menu

The context menu for the middle mouse button exactly mirrors the group structure for commands, as found in the menu and key editing dialogs — including any plug-in commands added to the system. If you know where a command is located in the master groupings, you can find it in the middle button menu.

## Keyboard Shortcuts

Keyboard shortcuts are displayed on the right side of buttons, if applicable. Keystrokes that need the Shift key are displayed in uppercase if they are letters. (e.g., A) Unshifted keystrokes are displayed in lowercase. (e.g., a) For example, M is Shift + M and & would be Shift + 7 on a US keyboard. This manual, however, always explicitly indicates when the Shift key should be used.



**WARNING:** You need to be aware of the state of your Caps Lock. If this is active, it can affect upper and lowercase keyboard shortcuts.

## Panel-Specific Shortcuts

Some panels have their own special keyboard shortcuts. To use them, you must have that panel active. (Click on it, if it isn't already active.) If the panel is not active and that shortcut has another meaning for LightWave in general, the general function will be run. Moreover, if the panel is active, but doesn't use the keyboard shortcut, it will be passed to the main interface.



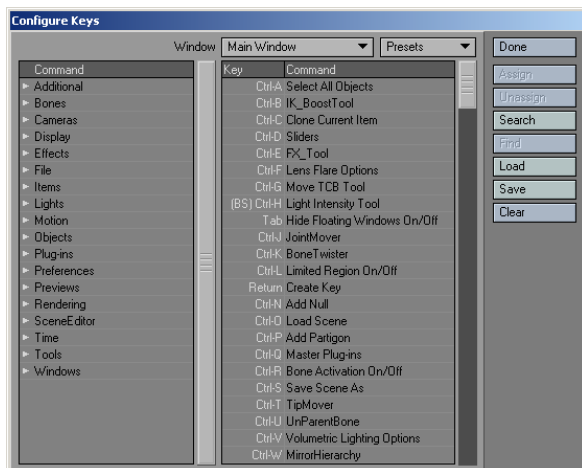
## Customizing Keyboard Shortcuts



**WARNING:** We strongly suggest that you keep the default keyboard mapping assignments and make new assignments only to unmapped keys.

Otherwise, technical support and using the documentation may become difficult.

Like the menus, keyboard shortcuts can be configured to suit your own needs. To display the Configure Keys Panel, choose Edit > Edit Keyboard Shortcuts.



The window on the left contains a complete list of all assignable commands grouped by type. The right window shows a complete list of all keystrokes and assigned commands, if applicable.

### To assign a command to a key:

Select a command in the left window.

Select the target key in the right window. (You can hit the desired key or keystroke combo to quickly select the key.)

Click the Assign button. This will overwrite any existing assignment. Alternatively, you may drag the command to the right window.

### To unassign a command to a key:

Select the target key in the right window.

Click the Unassign button.

### Finding Assignments and Commands

If you have a command selected in the left window and click Find, the command, if any, on the right will become selected. Selecting a command in the right window and clicking Find will select the matching command in the left window.

Clicking Search allows you to type in a search phrase to find a command. Note that this will only find the first instance and it is case sensitive.



**NOTE:** Keyboard shortcuts can be assigned to only one command. Assigning a command that is already assigned to a different key will assign it to the new key, but remove it from the old one.

## Maintaining Key Mapping Sets

The Load and Save buttons let you retrieve and store key mapping sets that you develop. The Clear button clears out all assignments, so use with caution.



**NOTE:** In Modeler, you can use Modeler > Preferences > Revert to Startup Preferences to restore your initial settings; however, any changes since you first ran the application will not be reflected.

Any available preset key mapping sets can be chosen from the Presets pop-up menu. The Window pop-up menu lets you select different menus to edit (e.g., Graph Editor), if available.

## Generic Plug-ins

Generic Layout plug-ins (Scene > Utilities: Generics) will appear in the Plug-in commands group. Thus, they can be added to menus or mapped to keyboard shortcuts.

## Configuration Files

The various LightWave configuration files are stored in your SYSTEM:PREFERENCES folder (with long names) under Macintosh OS X or the DOCUMENTS AND SETTINGS directory under Windows. Generally, you will not need to edit these files directly.

## Custom Configuration Files

Layout (LIGHTWAV.EXE), Modeler (MODELER.EXE) and the Hub (HUB.EXE) all support a -c command line argument that allows you to save your configuration files somewhere other than the system default (e.g., C:\DOCUMENTS AND SETTINGS). To use, add a -c<path> parameter when you launch the appropriate executable.

Example: MODELER.EXE -CD:\MONKEY

For Windows systems, just create shortcuts for Layout and Modeler and add the option to the Target entry in the Properties panel for the shortcut.

For Mac systems the options are implemented using a special text file known as a command line file, located typically inside the Program Files directory. The file must be named the same as the application followed by a space and then the word CMDLINE as follows:

```
LightWave cmdLine
Modeler cmdLine
LWSN cmdLine
```

If they don't exist, a user may create them if necessary. The files must be in plain text format. TextEdit by default saves files in .RTF format — this will NOT work. It is possible to modify the preferences for TextEdit to save files in Plain text versus Rich text. The same command option arguments are available as on the PC, but you will need to be aware of the different directory paths in use on the Mac.





For example, if the file LIGHTWAVE CMDLINE contained the following line, the Hub would be disabled and the path for LightWave's config files would be directed to a folder named SOMEPLACEELSE, on the root of the hard drive:

```
-0 -c"Macintosh HD:SOMEPLACEELSE"
```

The following line would set the location for the plugin preferences file:

```
-p"Macintosh HD:SOMEPLACEELSE"
```

The same holds for Modeler, the Hub and LWSN (LightWave ScreamerNet - the network rendering node). LWSN requires you to path the preferences location and the project directory before the job/ack references. An example of the contents of a LWSN CMDLINE file might be:

```
-2 -c"Macintosh HD:Users:myusername:Library:Preferences:"  
-d"Macintosh  
HD:Users:myusername:projectdirectory:" "Macintosh  
HD:Users:myusername:projectdirectory:job1" "Macintosh  
HD:Users:myusername:projectdirectory:ack1"
```

The -c argument sets up entries that the Hub tracks. So, if you click the Modeler button on the Layout interface, for example, it will run with the proper config file.

You must run Layout and Modeler with this argument at least once to store the entry in the Hub. Moreover, you must continue to use this argument when you run Layout or Modeler from an icon. If you don't, the config path entry in the Hub will change to the default. This can happen, for example, if you run Layout or Modeler directly from their .EXE icons. The path must be valid or the default will be used. Do not add a trailing backslash unless you are using the root directory. If your path includes spaces, enclose the entire path in quotes, like -C"D:\WHY DID I USE SPACES\CFGS".

A -p<path> parameter can also be added to select a different plug-in database file (LWEXT10.CFG). The path can be either a directory or a complete filename.

Using the <path> -r command will create random colors for the menus.

## Layout Startup Command

If the LW10.CFG configuration file contains a line beginning with "StartupCommand," Layout will try to execute the rest of the line as a command after the main interface is first opened. The command can have arguments and it can refer to a generic plug-in or script.

## Customizing Layout Viewport Navigation

The mouse directions used for viewport navigation can be customised by editing the WorldNavigation entry in the LW10.CFG configuration file. It has four boolean values that correspond to panning via icon and ALT key, and rotation via icon and ALT key. Zero values mean that the viewpoint moves in the same direction as the mouse, and ones mean that the world appears to move with the mouse. For example, the default in LightWave 9 is 0 1 0 1. The old 5.6 setting would be 1 1 0 0.

## Customizing Camera Presets

The camera resolution presets are stored in the Lw!0.CFG configuration file using the keyword "ResolutionPreset." Each preset includes a width and height, pixel aspect ratio, mask settings (left, top, width, and height) and a name that will appear in the pop-up menu. They can be modified, deleted, or added to. If no presets are found in the file, the original list is restored. See the configuration file for examples.



## LightWave 3D Basics

### LightWave Scene and Object Statistics

- unlimited images
- unlimited surfaces



NOTE: Because of memory considerations, the available number of cameras, objects and lights per scene is set at 100, 1000, and 1000, respectively, in the LW10.CFG file. The values may be increased — within the limits above, but not decreased below 100; however, this will require more memory, so keep the new values as small as possible. Do not edit the CFG file while LightWave is running.

### Measurement Units

LightWave supports a large number of units of measurement. When entering numeric values into input fields, you may specify the unit of measurement using the following abbreviations:

#### Abbreviation Description

um	Microns (one-millionth meter)
mm	Millimeters (one-thousandth meter)
cm	Centimeters (one-hundredth meter)
m	Meters
km	Kilometers (one thousand meters)
Mm	Megameters (one million meters)
mil	Mils (one-thousandth inch)
in or "	Inches (2.540 centimeters)
ft or '	Feet (.3048 meters)
kft	Kilofeet (One-thousand feet)
mi	Miles (5,280 feet)
nmi	Nautical miles (1.151 miles or 1852 meters)
f	frame
s	seconds



HINT: Since the meter is the default unit of measure for the SI or Metric unit systems, entering the meter equivalent is often easier than typing in the abbreviations. For example, enter .01 for centimeters, .001 for millimeters, 1000 for kilometers, and so on.

### Filenames and Extensions

We recommend that users avoid using spaces in hard drive, directory, and file names for scene, image, and object files: this can possibly cause problems later if you use LightWave's distributed rendering feature.

LightWave uses several filename extensions for the different types of files it uses. The following list gives you some examples of these extensions:

.env	Envelope data
.lwo	LightWave object
.lws	LightWave scene file
.mot	Motion data
.p	Plug-in
.srf	Surface attribute file

Generally, LightWave will automatically add the appropriate extension to filenames when saving files, if one is not provided. However, this is not the case with most plug-ins.

Additionally, there are numerous standard filename extensions (.bmp, .iff, .tga, .wav, etc.) that LightWave uses, but are not specific to this application.



## Object Import/Export

You may *load* foreign object formats supported by LightWave directly into Layout or Modeler. However, only Modeler can be used to *save* an object using a foreign object format. To do this in Modeler, choose File > Export and select one of the Export options.

### Imported Formats

OBJ	M, SA, UV
DXF	M, SA
3DS	M, SA, SSP, T, UV
FACT	M, SA, SSP, T

### Exported Formats

LW5 (5.x LWO format)	M, SSP, T
OBJ	M, SA, UV
DXF	M, SA, T, UV
3DS	M, SSP, T, UV

M = Geometry mesh

SA = Surface assignments (LW default properties)

SSP = Standard surface properties (color, glossy, etc.)

T = Texture maps (non-procedural)

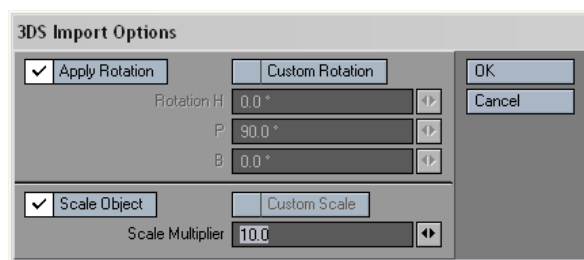
UV = UV texture information

VRML

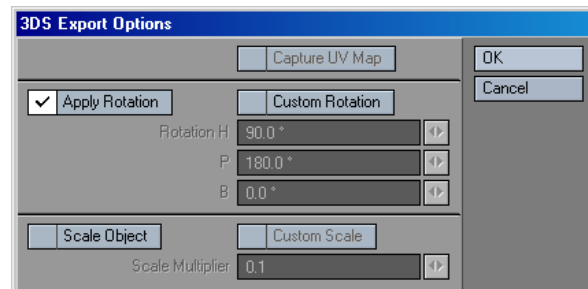


NOTE: Before exporting to a foreign object format, save the object as a LightWave .LWO file first.

When appropriate, an options dialog may appear when you load or save a foreign object format file. Some trial and error may be needed to get acceptable results.



3DS import option dialog



3DS Export option dialog



NOTE: Generally, you should merge points after loading a 3DS object.

## Image and Animation Types

You can load and save various image and animation formats. What is available to you depends on the platform you are using, as well as the plug-ins you have active. Please check the NewTek Web site ([www.newtek.com](http://www.newtek.com)) for additional plug-ins as they become available.

The Flexible Format, TIFF LogLuv, and Radiance image formats contain high dynamic range data (discussed below). In addition, they store the data with floating-point accuracy instead of using integers.



NOTE: When a 32-bit format is selected for saved RGB images, the 8-bit Alpha image is stored along with the 24-bit RGB data. However, when you save alpha images directly, you save only the alpha image, even if the format is 32-bit.



NOTE: The filename extension for Radiance images is .HDR. If you have any of these images using the .PIC extension, you should rename them.

Supported image formats: PSD, Alias, BMP, Cineon FP, Flexible Image Format, IFF, JPEG, PCX, PICT, PNG, Radiance, RLA, RPF, SGI, Sun, TIFF, Targa, VBP, and YUV image formats.

Supported animation formats: AVI (any installed codec), QuickTime, QuickTime Stereo, RTV (VT[3]™), Flexible Image Format, Storyboard, 4X Storyboard, Film Expand.

Supported Web 3D: QuickTimeVR, VRML and Shockwave for export.



## Images and Memory

*Mipmapping* is a process where a series of images with progressively lower resolutions are created from a base (full-resolution) image. Each pre-processed image is a power of two smaller than the previous level. As textures move farther away from the camera, they obviously must be scaled down. However, because the mipmaps are pre-processed and loaded into memory, they can be used immediately for textures instead of continually shrinking down the base image; this saves rendering time, but requires more memory. Eight-bit greyscale and eight-bit index-color images remain in eight-bit form internally. For color images, their mipmaps, however, are 24-bit since they must include colors in between color palette entries.

Greyscale images remain at eight-bit for mipmaps. Floating-point images use 96 bits per pixel and have 96-bit mipmaps.

Here are some examples dealing with a 512 x 512 image in various formats, and how much memory is used for the base image and the first couple mipmaps:

512 x 512 eight-bit greyscale image:

Base image 262,144 bytes (512 x 512 x 1)  
 First mipmap 65,536 bytes (256 x 256 x 1)  
 Second mipmap 16,384 bytes (128 x 128 x 1)  
 Total 344,064 bytes

512 x 512 eight-bit color-mapped image:

Base image 262,144 bytes (512 x 512 x 3)  
 First mipmap 196,608 bytes (256 x 256 x 3)  
 Second mipmap 49,152 bytes (128 x 128 x 3)  
 Total 507,904 bytes

512 x 512 24-bit image:

Base image 786,432 bytes (512 x 512 x 3)  
 First mipmap 196,608 bytes (256 x 256 x 3)  
 Second mipmap 49,152 bytes (128 x 128 x 3)  
 Total 1,032,192 bytes

512 x 512 floating-point image

Base image 3,145,728 bytes (512 x 512 x 12)  
 First mipmap 786,432 bytes (256 x 256 x 12)  
 Second mipmap 196,608 bytes (128 x 128 x 12)  
 Total 4,128,768 bytes

## High Dynamic Range Images (HDRI)

In computer graphics, color is displayed as a *triplet* value: red, green and blue. These values typically range from 0 to 255. Those 256 steps of color represent eight bits and together all three channels make up a 24-bit image. This means the maximum amount of color or luminance variation an image is allowed is merely 256 steps.



NOTE: See the discussion on radiosity for additional information on high dynamic range images.

In the real world, the human eye can perceive a much higher range of brightness and color values. Film can also react to a much wider range. Video cameras, however, are limited to a fixed range that fits closely to the same 256-step limit.

When exposed to *high dynamic range visuals*, such as a sunset or a desert landscape, the lens of a camera will produce some level of artifacting. Some of those artifacts can be seen as blooming areas of brightness, color bleed, luminance spill, lens streaking, and many other *visual cues* that tell the viewer there is a very bright light source in the scene.

These very bright surfaces can also contribute to the overall lighting of a scene. For example, sunlight streaming into a room will bounce off the floor and add a subtle illumination to the walls and ceiling that would otherwise be left dark (this bouncing of light is known as radiosity). All of these effects can be seen in images captured by devices that do not support high dynamic ranges.

Because computer graphics applications were designed to output to devices that would not understand pixel values above RGB 255, 255, 255, most applications do not provide for any value to exceed these limits. LightWave, however, calculates all internal data without limits and with IEEE floating-point accuracy. This means that when LightWave points a light at a surface, while the final rendered pixel may reach only RGB 255, 255, 255 for pure white, internally that pixel may have reached ten times that amount. This may not seem significant at first glance — white is white after all — but, if we look at how LightWave utilises that data, it becomes very exciting.



## Import/Export

LightWave can utilise high dynamic range detail, as it is generated internally (e.g., in the case of a very bright light) or from data in image files. This can be imagery generated from a series of photographs taken at various exposures and composited (see *Recovering High Dynamic Range Radiance Maps from Photographs* by Paul E. Debevec and Jitendra Malik at <http://www.debevec.org/>) or data rendered in LightWave saved in one of the high dynamic range formats.

Once these images are imported into the system, they can be used just like any other image in LightWave 3D (e.g., as a texture, background, etc.). During the rendering process LightWave will respect the extra data in the image to assist in secondary lighting and other calculations.

Imagine using a high dynamic range image as an environment wrap (e.g., using the Image World environment plug-in.), which also illuminates the scene. With the appropriate imagery you can illuminate a scene without any lights and the results will match the look and feel of the original photograph.



Image by Darkside Animation.

Once LightWave finishes rendering, you can export images with the same high dynamic range data. This lets you bring that data back into LightWave or into compositing applications that support such data. Using this extra data in the compositing process is very important as it can more accurately represent imagery as it would look if it were recorded directly to film. For example, compositing applications could use the extra dynamic range data to calculate the amount of diffuse bloom or color bleed from one pixel to the next.

## Internal Compositing

Another area where high dynamic range imagery is supported is in LightWave's own internal compositing through pixel and image filters. Any filter can be designed to take advantage of the high dynamic range data with floating-point accuracy. This way, high dynamic range data can be leveraged in the post-process phase with included filters and by third party additions.

## Preview Compression Codec

The NTCodec.dll adds LightWave's preview compression scheme into the Windows AVI codec. You just need to install it if you want to play/load compressed preview anims into other Windows applications/players. See the README.TXT file located in the NTCODEC directory for installation instructions.

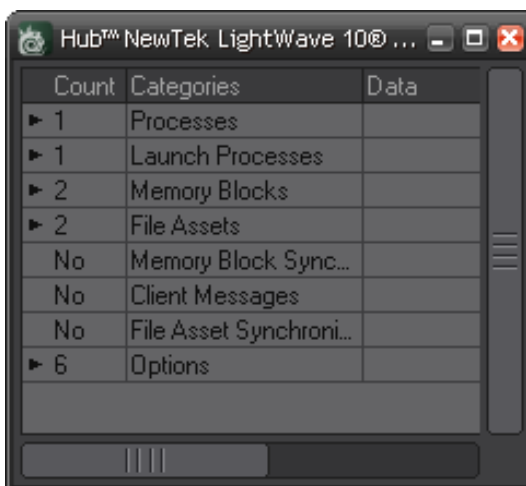
## The Hub

The Hub is essentially a *message board* that LightWave modules use to synchronise information. It contains entries like *synced* object filenames and configuration memory blocks. When the same object is loaded into both Layout and Modeler, changes made to the object are automatically synchronised. If the object appears in Layout, but not Modeler, you can quickly load it into Modeler by selecting it from the object pop-up menu — initially, the name will be ghosted.



NOTE: If you have modified an object in Modeler (without saving) and then load the object file into Layout, the modified version will appear — since it is synced — not the data from the object file.

The Hub runs automatically when you run LightWave. Basically, it is a background process and you shouldn't need to directly interact with it. However, you can bring up its interface by clicking its icon (when running).



If you expand the Launch Processes menu, you should see applications that the HUB is aware of (e.g., Layout, Modeler). You can launch the application by double-clicking it. If you single-click (to select it) and then press the I key, you will also launch the application. If you press the Delete key instead, you will delete the application from the launch list. (This has no effect on the program files on your hard drive.)



NOTE: The Launch Processes menu will be empty initially. However, once you run Layout or Modeler, entries will be added.

Under Windows, the Hub icon will appear in the System tray. You can right-click on it to access a small menu:

Open	Open Hub window
Close	Close Hub window
Launch	Launch applications that the Hub knows about
Properties	Set options to quit the Hub after you exit LightWave
Exit	Close the Hub program





## Properties

You can set the Automatic Shutdown to various time intervals, after which the Hub will shutdown when there is no activity. Note that even when Layout and Modeler are idle, there is still some minimal communication activity which will keep the Hub from shutting down.

When Layout or Modeler are running, the Hub will periodically request that the applications write their recent changes to temporary files. The frequency of this Automatic Save is set by the Hub option. The temporary files are located in a lwhub directory inside your normal system temporary directory. The files have names that include the base name of the object with a numeric prefix. So, if you are editing an object called myObject.lwo and you crash after an editing session, you can still find an auto-saved version of your edits in the temporary directory with a name like 0012.myObject.lwo.



NOTE: If you would like to run Layout and/or Modeler independently, append -0 to the command (e.g., Lightwav.exe -0. That's a zero, by the way.) You will have to load and save objects manually.

## Command IPC

We have implemented an interprocess communications system, CommandIPC, into Layout and Modeler. When LightWave is started without Hub support (the -0 command line option), Layout and Modeler all notify each other when a mesh file has been saved to disc, and the applications that have that file open and have not applied modifications during the session will update from the new version on disc. For example, if both Layout and Modeler have "bob.lwo" open, Modeler has some edits applied to it, and Layout saves it transformed, Modeler will not automatically update from the disc.

If subsequently the object is saved out from Modeler, that will overwrite the changes made by Layout, and Layout will update from the save if no further changes were applied in Layout during the interim. LightWave CORE users should note that the IPC system is only active in the Windows and Mac builds, since LightWave HC code is not built for Linux.

The 'isolate' flag ('-i') will completely disable all IPC mechanisms (Hub and CommandIPC), leaving the application isolated from all other product actions.

## Optimizing RAM Usage

If your computer is accessing virtual memory frequently during rendering, you may find a substantial increase in performance by installing more RAM on your machine or reducing the amount of RAM LightWave needs to store the render information.

One of the best ways to minimise the need for RAM is to reduce the color of the images used for texture mapping effects. Except for those used as a Surface Color, Texture Maps usually need to be only 256-level greyscale images.

Other ways to reduce RAM usage are to render your animation in multiple passes or decrease Segment Memory Limit on Layout's Render Globals Panel.

## LightWave 3D Resources

### Internet Resources

LightWave-specific forums, newsgroups and mailing lists are maintained on the Internet. Here you can find new users asking questions about using LightWave and experts answering them. Also, many topics related to computer animation are discussed in these groups, such as the performance of various graphics cards, CPU speeds, platforms, and many more. NewTek's own forum can be found at <http://vbulletin.newtek.com/> and there are many others out there.

The LightWave-related newsgroup for text-only discussions is at [comp.graphics.apps.lightwave](mailto:comp.graphics.apps.lightwave); and for those who wish to share images [alt.binary.3d.lightwave](mailto:alt.binary.3d.lightwave).

Several Internet mailing lists can be found at [www.groups.yahoo.com](http://www.groups.yahoo.com). The mailing lists are LW3D (support and technique discussions), LW3D-DISC (related discussions for LightWave users), and LW-Plugin (plugin developers discuss programming issues). NewTek also maintains mailing lists - those are located at <http://lists.newtek.com/mailman/listinfo>, and include an LScript mailing list for support of LScript users.

## NewTek Web and FTP Sites

In addition to information about NewTek products, upgrades, and the latest releases of LightWave software, our websites ([www.LightWave3D.com](http://www.LightWave3D.com) and [www.newtek.com](http://www.newtek.com)) have tutorials, LightWave images and animations, technical support FAQs, tech support email links, and links to related sites.

NewTek maintains an FTP site ([ftp.newtek.com](ftp://ftp.newtek.com)) on the Internet. Here you can find objects, scene files, images, and other items of interest to LightWave users.



## Community

A unique attribute of LightWave 3D is the LightWave community. It is rare to find such a passionate group of professionals that are so talented and yet so willing to share their secrets, ideas, and creations. Make certain you also take a look into the available resources such as local users groups and training books and videos.

## Technical Support

The best source for help with installing or configuring software or hardware is the retailer from whom you purchased your NewTek product. While we have made every effort to keep your software and hardware trouble-free and easy to use, you may occasionally need help right from the source. If you have problems with NewTek-supplied hardware or LightWave doesn't seem to be functioning as it should, please contact technical support in one of the following ways:

By email: [tech@newtek.com](mailto:tech@newtek.com)

By fax: 210-370-8001

By telephone: 210-341-8444. Experts are available to answer questions from 8:30AM-6:30 PM Monday-Friday and 10AM-2PM Saturday and Sunday. All times are US Central Time.

In your communication or when calling, please include as much information as possible including:

Your computer's operating system and version;

The version of LightWave you are using;

The amount of RAM in your computer;

Which display card you are using and what version of drivers are installed;

CPU and make and model of computer or motherboard;

Your product serial number



NOTE: Your product must be registered before you can receive support.

## Customer Services

NewTek's Customer Service department provides a variety of services and assistance to LightWave users. These include:

1. Registration of your NewTek product. 24-hour registration is available at [register.newtek.com](http://register.newtek.com)
2. Exchange or replace a parallel or ADB dongle, or a single-platform USB dongle for a dual-platform USB dongle
3. Assist you with recovering your license key in the event you have a system problem and need to re-install LightWave
4. Answer your questions about NewTek policies and procedures and special offers

You may contact NewTek Customer services in the following ways:

By email: [cs@newtek.com](mailto:cs@newtek.com)  
Telephone: 1-800-862-7837  
Fax: 210-370-8001

NewTek's Customer Service Center is open from 9:30 AM-5:30 PM Monday-Friday. All times are US Central Time.

NewTek's postal address is:

NewTek  
5131 Beckwith Blvd  
San Antonio, TX 78249

## NewTek Europe Customer Service Information

For all LightWave customers in Europe, the Middle East, Africa, Russia and India.

## Customer Services and Technical Support

Telephone: +33 557 262 262 (Europe, office hours: 9:30 to 17:30 CET)

Fax: +33 (0)557 262 261

Emails and faxes in English or French preferred.

NewTek Europe postal address:

NewTek Europe  
70 avenue de Capeyron  
33160 Saint Medard en Jalles  
France

Email:

Customer Service: [info@newtek-europe.com](mailto:info@newtek-europe.com)

Technical Support: [tech.lw@newtek-europe.com](mailto:tech.lw@newtek-europe.com)

Emails and faxes in English or French preferred.

## Australia/New Zealand/Indonesia

New Magic Australia Pty Ltd

Phone (02) 9528 4555 / (03) 9752 5200

Fax (02) 9589 2311

International callers replace (02 with +612 / 03 with +613)



---

# Volume i

---

## Modeler

---







---

## Chapter 2: Introduction to Modeling

---



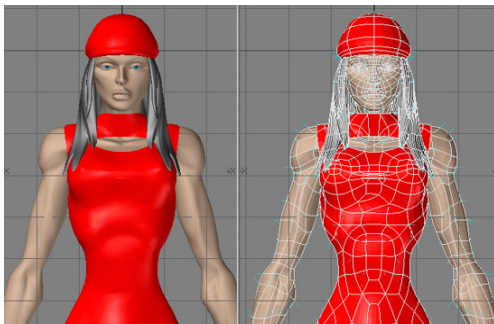
## Introduction to Modeling

LightWave Modeler lets you create objects from scratch or edit existing objects. Object Modeling is the design and creation of wireframe objects from a simple shape, like an apple, to a complex shape, like a finely-detailed sports car. In Layout, you are the interior designer moving furniture and putting up pictures. In Modeler, you are the furniture maker and the builder of the house.



## Components of a 3D Object

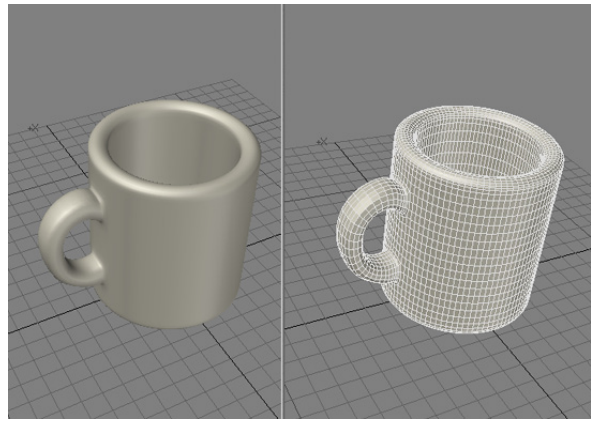
What makes up a 3D object? It has two traits: its physical shape and its visual appearance. The physical shape of an object comprises points joined by lines to form faces that we call polygons. The form that the polygons present to us makes up an object's shape.



The visual appearance of an object consists of color and texture qualities that connote realism to the eye, which are called surface attributes or material properties. When you enter a room and look at your chair you see a certain shape (that of a chair) and a certain surface (maybe wood or metal), so you think, "chair". In LightWave 3D you have the tools to create simple or complex objects and define their surface qualities with great precision and detail.

### Modeling in 3D

How would you make a mug? It depends; if you were a sculptor, you would mould a lump of clay. If you were a designer, you would draft it with paper and a mechanical pencil. If you were an artist, you would probably sketch it. But suppose you were using a computer. How would you make a mug? What tools would you use?

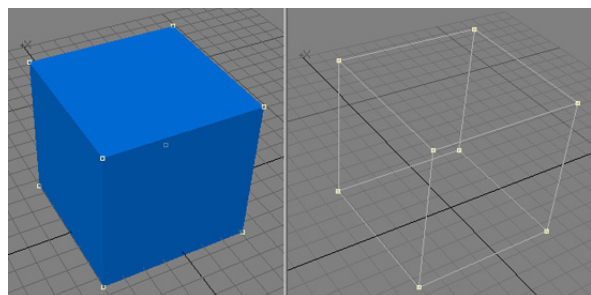


You want to portray the mug with realistic color, depth, and shading. You need drawing tools, yes, but also power tools! With the computer, you can create the mug in a variety of ways. LightWave recreates drafting and workshop tools on screen so that people who are neither sculptors nor traditional artists can use them just as easily. In fact, LightWave's tools do more than their real-life counterparts. There are several bonuses to computer design also. You can undo a mistake, something that is difficult to do in many of the arts. And there's no mess to clean up.

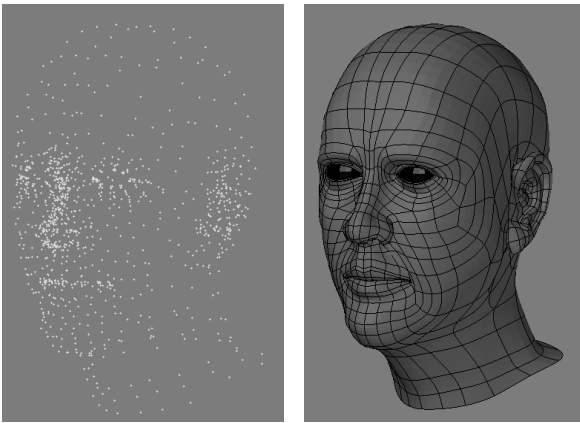
## Points, Polygons, and Edges

Generally, 3D Modeling is based on two simple elements: the point and the polygon. A point is a location in space, just as a point in a dot-to-dot coloring book represents a location on the page. Points are used as anchors to create polygons. Since points alone do not have height, width, or depth, they cannot be seen or rendered. However, you can load them into Layout and use points as invisible influences on other objects: as a center of rotation, or to play an invisible role of influence in an object hierarchy, for example. A third, in-between, mode is the edge mode. Edges are the lines which connect the individual points and form the outline of a polygon.

The process of creating 3D objects is a lot like drawing in dot-to-dot coloring books — of course, with 3D Modeling, the dots can also be placed in a third dimension (i.e., depth). Once placed, the dots can be connected with lines to form polygons. That collection of polygons, also referred to as an object, forms a recognizable shape that the computer can draw and animate. Polygons are usually three-sided triangles or four-sided quadrangles (although they can consist of more sides).



You create or edit all objects using the same XYZ coordinate system used in Layout. The difference is that in Layout you are usually positioning an entire object, whereas in Modeler you are positioning the points that make up an object. Modeler also uses the XYZ coordinates 0, 0, 0 as the Origin.



Left: Points that make up a head, Right: Polygons that make up the same head

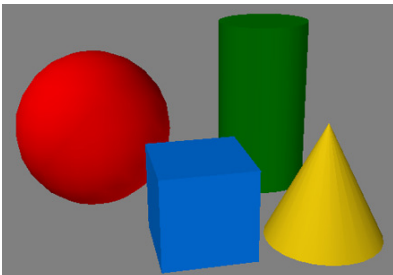


**NOTE:** The Origin is the default rotation center (called the pivot point) for objects in Layout.

Keeping this in mind as you model your objects will make the objects load into Layout properly without having to move the pivot point.

## Editing Objects

Generally, to build a 3D object, you create, combine, and modify simple shapes into more complex shapes. This is known as editing. The basic building blocks (i.e., primitives) include cubes, spheres, cylinders, and other shapes. Using tools not unlike those found in a wood or metal shop, you form a realistic representation of the envisioned object.



**HINT:** Have a Plan!

If you were building a desk, you would work from a sketch, a photograph, plans, or a small model of the desk you intended to build. This approach also works well for Modeling 3D objects. Having the actual shape or a representation of the object in front of you while you design is invaluable, and it will aid you in determining size, shape, angle, color, and more. Whenever possible, work from pictures or small-scale models. Picture books, magazines, blueprints, and visual dictionaries are useful companions to have nearby.

## Selection Mode: Points, Edges and Polygons

The process of object editing involves manipulating points, edges and polygons. It will include selecting some portion of an existing object, choosing the operation to perform, adjusting any parameters for that operation, then carrying out the operation.

The concept of selecting is very important to Modeler. Selection is the act of specifying either the object itself, or a portion of it, that you intend to edit next. The most basic elements that you can modify include points, edges and polygons — the dots and multi-sided shapes that make up all objects. Any number of points, edges, polygons, or objects, in any combination, may be selected for a specific operation.



**NOTE:** Everything in a layer is considered selected when nothing is selected.



**NOTE:** When selecting and unselecting, make sure that no Modeling tool is currently active. You can usually tell if a tool is active based on the mouse pointer. You can quickly deactivate any tool by pressing the Spacebar (or RETURN); however, if no tool was active, you will change the edit mode.



## Point Selection

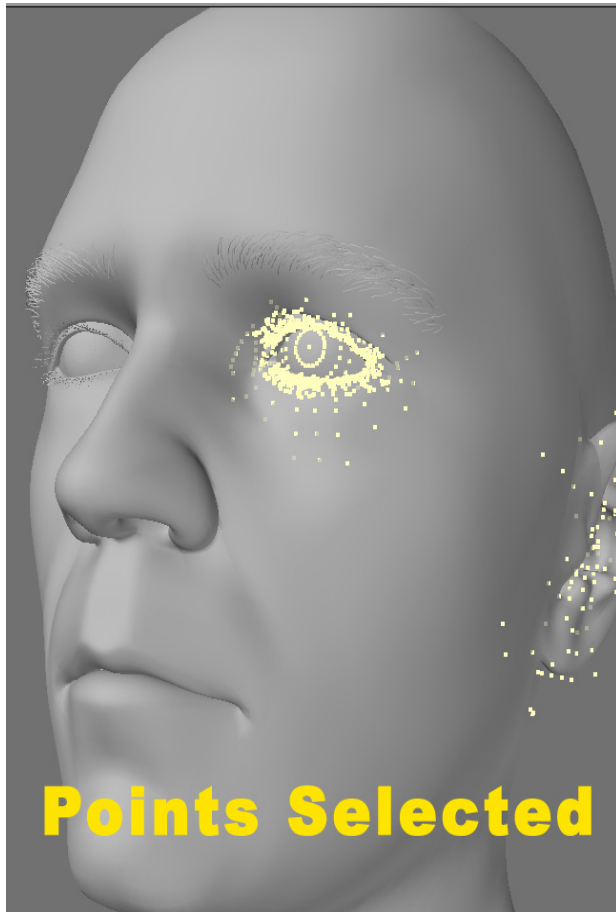
Points generally appear in two ways on-screen, selected and unselected. Selected points are highlighted, while unselected points remain small dots.

### To select points:

**Step 1:** Use **Point Selection** mode (**Point** button or **Ctrl+G**). The **Mode Selection** buttons are located at the bottom-left of the screen.



**Step 2:** Drag your **LMB** over points in an object and then release the mouse button to select those points.



**NOTE:** Once you have selected points and released the mouse button, you cannot select additional points in the same manner.



**HINT:** If you know the number of points you want to select, check the information display above the Point (Selection mode) button. It shows you the number of selected items.

### To add more points to a selection:

Hold the **Shift** key while you select with your **LMB** to add points to the existing selection or just click your **MMB**.

### To lasso a group of points:

With the **RMB**, drag out a circle around points to select a group. As you might expect, you can add more points to a selection by holding the **Shift** key and dragging out a circle around the additional points you wish to select.

### To unselect points:

If selected points exist (and you have released the mouse button), dragging over them with your **LMB** will unselect them. You can also use the lasso by dragging with your **RMB**.

### To unselect all points:

Click in any non-active (i.e., not a button) area on the toolbar to unselect all points. You can also use the **Drop Current Tool** command (**I**). Also, clicking on the number of selected points



in the Selection Counter information display above the **Point** (Selection Mode) button will deselect all selected points.

## Edge Selection

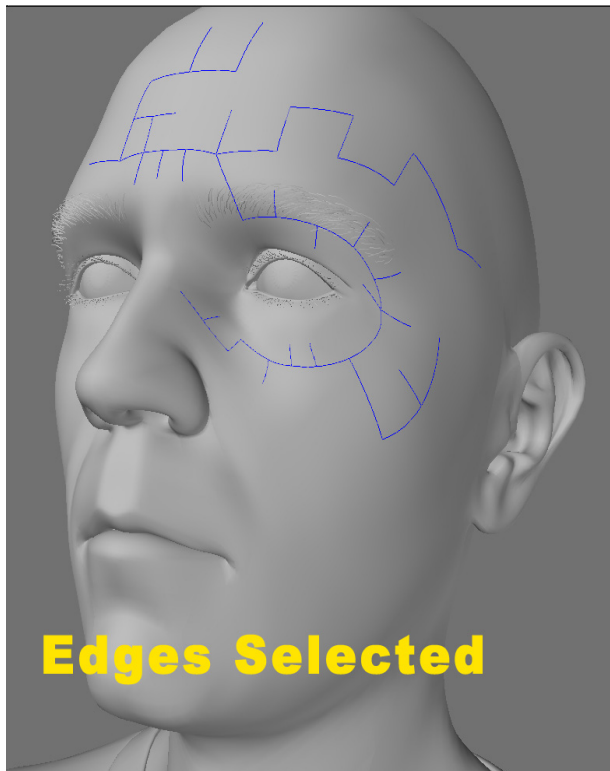
Edges generally appear in two ways on-screen, selected and unselected. A selected edge will appear highlighted with outlines in yellow (default). You will see that selecting edges is very similar to selecting points or polygons.

### To select edges in a viewport:

**Step 1:** Use **Edge** selection mode. The **Mode Selection** buttons are located at the bottom-left of the screen.



**Step 2:** If you are using a viewport set to a non-shaded Rendering Style (Display Options), like wireframe, drag your LMB over the edges and then release the mouse button.



**NOTE:** Once you have selected edges and released the mouse button, you cannot select additional edges in the same manner.

### To add more edges to a selection:

Hold the **Shift** key while you select with **LMB** to add more edges to the existing selection or just click your **MMB**.

### To lasso a group of edges:

With the **RMB**, drag out a circle around edges to select a group. Only the edges located completely within the outline are selected.

Of course, you can add more edges to a selection by holding the **Shift** key and then drag out a new circle around the edges you wish to add to the selection.

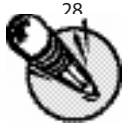
### To unselect edges:

If selected edges exist (and you have released the mouse button), dragging over them with your **LMB** will unselect them. You can also use the lasso by dragging with your **RMB**.

### To unselect all edges:

Click in any non-active (i.e., not a button) area on the toolbar to unselect all edges. You can also use the **Drop Current Tool** command (**I**). Also, clicking on the number of selected edges in the Selection Counter information display above the **Point** (Selection Mode) button will deselect all selected edges.





**NOTE:** Edges are always part of a polygon. Cutting, or otherwise deleting or hiding, an edge will also remove the associated polygon.

## Polygon Selection

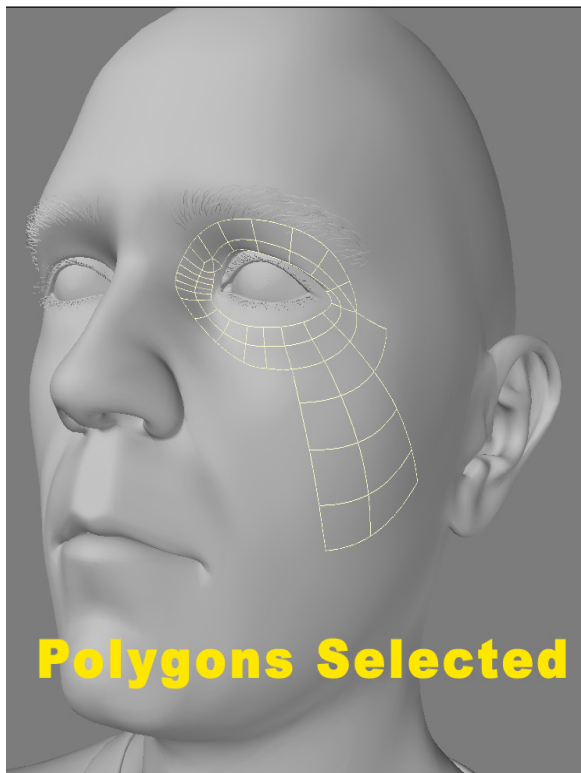
Polygons generally appear in two ways on-screen, selected and unselected. A selected polygon will appear highlighted with outlines in yellow (default). You will see that selecting polygons is very similar to selecting points and edges.

### To select polygons in a viewport:

**Step 1:** Use **Polygon** selection mode (**Ctrl+H**). The **Mode Selection** buttons are located at the bottom-left of the screen.



**Step 2:** If you are using a viewport set to a non-shaded Rendering Style (Display Options), like wireframe, drag your LMB over the edge of polygons and then release the mouse button. This selects polygons that share that edge. With a shaded rendering style like Texture or Smooth Shade, you can click directly on the polygon's face.



As polygons become selected, they will become highlighted and you will see a dotted line extending out of the face of the polygon and perpendicular to that polygon (The **Normal** display option must be active.). This dotted line is also referred to as the normal of the polygon or surface. The normal line indicates which side of a polygon has a face (the surface of the polygon you can see) or which direction the polygon is facing. In LightWave polygons by default have only one face. This means if you rotate a polygon the other side will not be visible because it is not double sided.



**NOTE:** Once you have selected polygons and released the mouse button, you cannot select additional polygons in the same manner.



**HINT:** If you know the number of polygons you want to select, check the information display above the **Point Selection Mode** button. It shows you the number of selected items.

### To add more polygons to a selection:

Hold the **Shift** key while you select with **LMB** to add more polygons to the existing selection or just click your **MMB**.

### To lasso a group of polygons:

With the **RMB**, drag out a circle around polygons to select a group. Only the polygons located completely within the outline are selected. Of course, you can add more polygons to a selection by holding the **Shift** key and then dragging out a new circle around the polygons you wish to add to the selection.

### To unselect polygons:

If selected polygons exist (and you have released the mouse button), dragging over them with your **LMB** will unselect them. You can also use the lasso by dragging with your **RMB**.



**NOTE:** Selecting a group of polygons and then deselecting the ones you don't want is often the only way to select the ones you do want.

### To unselect all polygons:

Click in any non-active (i.e., not a button) area on the toolbar to unselect all polygons. You can also use the **Drop Current Tool** command (**I**). Also, clicking on the number of selected polygons in the Selection Counter information display above the **Point** (Selection Mode) button will deselect all selected polygons.

### Selection with In-line Points/Polygons/Edges

If there are points/polygons/edges in-line (on top of each other) in a viewport using a non-shaded rendering style (i.e., Wireframe), you select all of the points/polygons/edges when you select with your mouse. However, shaded viewports respect polygons facing away from you and those obscured by other polygons. Thus,



you can select by clicking on a polygon's face without worrying about what's behind or on the other side. You can change the **Rendering Style** from the pop-up menu on the viewport titlebar or on the **Viewport Tab** of the **Display Options Panel**.

## Symmetrical Selection

When you select polygons/points/edges on the positive X axis and Symmetry mode (**Symmetry** button) is active, polygons/points/edges on the negative X axis are also selected (or deselected). Polygons/points/edges must be exactly opposite each other on the positive and negative sides of the X axis for this command to work properly.



**NOTE:** Symmetry also affects polygon editing.

**Sel Entire Surface:** Selects all the polygons making up the surfaces of the polygons that are currently selected.

**Sel Entire Part:** Selects all the polygons making up the parts of the polygons that are currently selected.

**Select Path:** Selects a shortest path (there can be multiple) between two elements. The selected elements can be of any type (vertex, edge, polygon).

**Select Outline Points:** Selects the outer points of a selected set of polygons.

**Select Outline Edges:** Selects the outer edges of a selected set of polygons.

## Edge Selection

**Loop Expand:** Expands an edge selection along an edge loop.

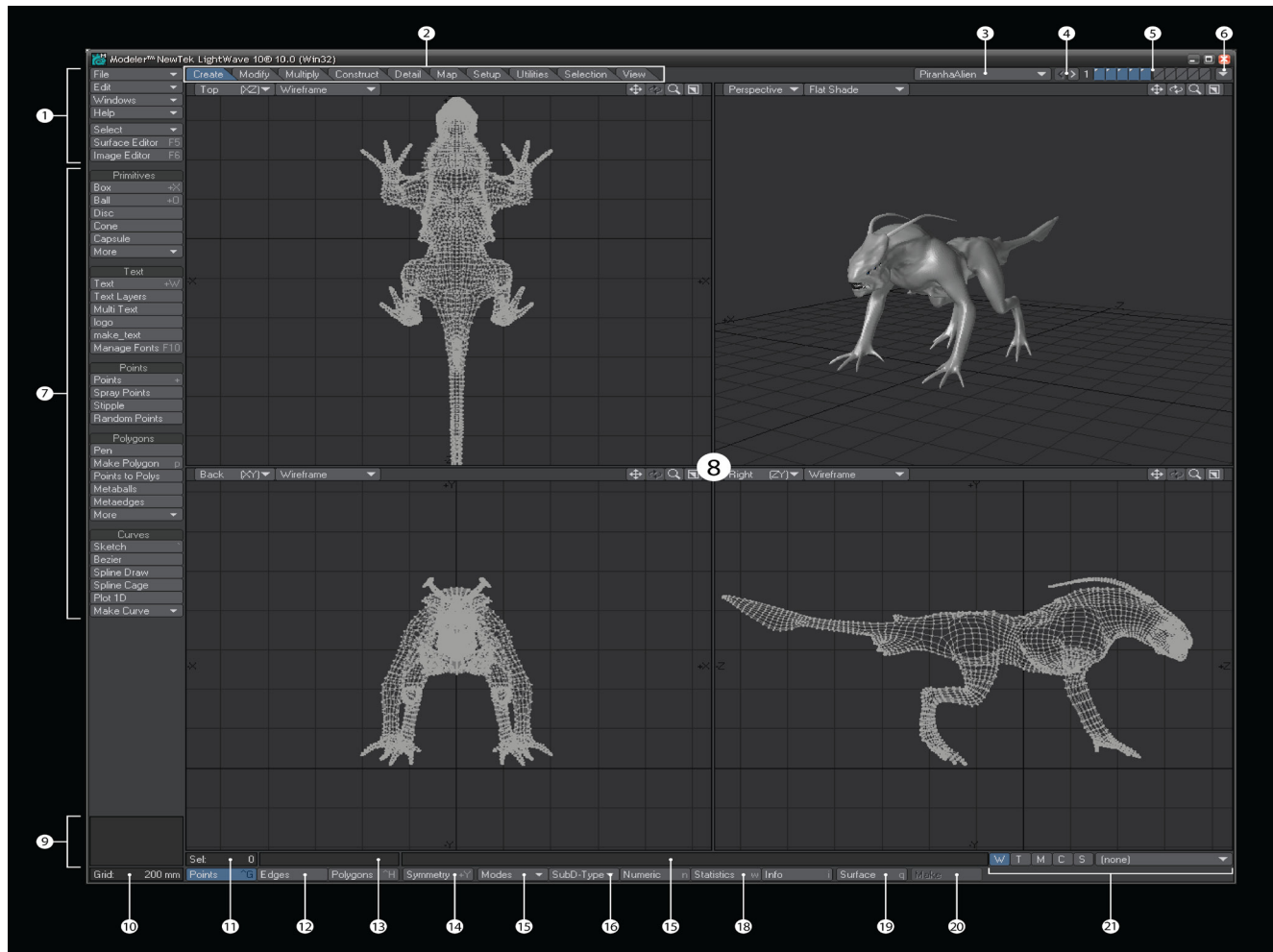
**Loop Contract:** Shrinks an edge selection along an edge loop.





## Interface

Modeler's screen is divided into several areas. By default, the workspace of the screen consists of four viewports. Modeler displays four simultaneous points of view (Top, Back, Right, and a forced-perspective view), each looking at the same portion of the workspace from a different angle. These are not unlike Layout's viewports. Changes made in one view are immediately updated in the others.



1. Top Tool Bar, 2. Menu Tabs, 3. Current Object, 4. Layer Banks, 5. Layers, 6. Layout Access, 7. Main Tool Bar, 8. WorkSpace, 9. Info Display, 10. Grid Display, 11. Selection Info, 12. Edit Mode, 13. Progress Bar, 14. Symmetry, 15. Tool Tips, 16. Mode Menu, 17. Sub-D Type Menu, 18. Statistics Window, 19. Change Surface, 20. Make Tool, 21. Vertex Maps



**NOTE:** Each area of Modeler's Interface is fully customisable in the **Configure Menu Panel (Alt + F10)**.

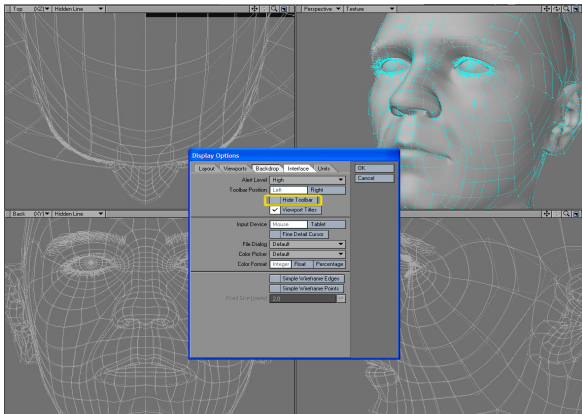


## 1 and 7. ToolBar

The toolbar sits at the side of the screen. The buttons presented will vary depending on which menu tab you select along the top. The **Top Tool Bar** buttons will appear no matter which Tab is selected. You can completely hide (or unhide) the toolbar by pressing **Alt+F2** (or choosing **Edit > Display Options > Hide Toolbar On/Off**).

### Hiding Panels

Since your screen can often get cluttered with open panels, you can quickly hide/show these floating windows by pressing **Alt+F1**.



2.

### Modeler Menus

The **Menu Tabs** located at the top of the interface will determine which tools appear in the Tool Bar. Each Toolbar is divided into menu groups. Generally, menu group names that are verbs contain commands based on the type of action they perform. Menu group names that are nouns contain commands based on the type of object they affect.



**Create** — Primitives, Points, Polygons, Curves and more. All the basic building blocks can be found here - see Chapter 3.

**Modify** — Move, Rotate, Scale, and more - see Chapter 4

**Multiply** — Expand on existing geometry - see Chapter 5

**Construct** — Reduce and combine points and polygons and much more - see Chapter 6.

**Detail** — Tools to detail your object - see Chapter 7.

**Map** — All the tools you need to create and manage **Vertex Maps** - see Chapter 8

**Setup** — **Skelegons** and other tools that can be used inside Layout - see Chapter 9

**Utilities** — LScript and Plugin access - see Chapter 10.

**View** — Display options and selection tools can be found here -

## 3. Current Object

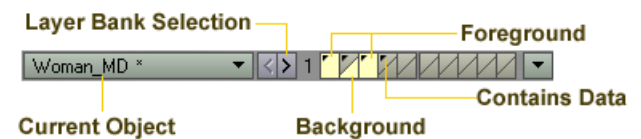
Like files in your wordprocessor, multiple object files can be loaded simultaneously. The **Current Object** pop-up button to the left of the layer buttons switches from object to object. Object names listed with an asterisk (\*) have been changed and may need to be saved. Attempting to close the application triggers a request to save all changed objects. Ghosted objects are ones that have been loaded into Layout, but not into Modeler. Choosing a ghosted object from the pop-up menu will load it.



Use the **Current Object** pop-up menu to select which object you are editing.

## 4. Layer Bank Selection

You can use the layer buttons in the top-right corner of the main interface to work with layers in banks of ten. LightWave gives you unlimited banks of ten which can be accessed here.



## 5. Layers

Every object can be a "MultiMesh", that is, consist of an unlimited number of layers, similar to layers in many paint programs.

The MultiMesh lets you work independently on specific parts of an object. During Modeling operations, you can set layers independently to be in the foreground or background, so you can work on a combination of layers as if they were in the same layer. Background layers can be used as a reference, but are often required when using certain Modeling tools.

## 6. Layout Access

This drop down menu gives you the following options:

**Switch to Layout** switches to your Layout window if it is open or opens up Layout if you do not already have it open.

Changes to objects in Modeler are reflected automatically in Layout when you select the Layout interface. You can force this synchronization by choosing **Synchronize Layout**.

**Send Object to Layout** sends the current object to Layout. Actually, it sends a pointer to the current object's file. As such, the object must be saved to your hard drive first.



**NOTE:** These options assume that you have the Hub running.



## 8. Workspace

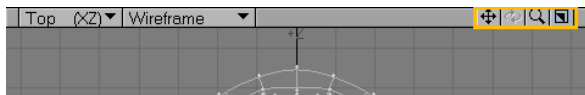
By default, the workspace of the screen consists of four viewports. Modeler displays four simultaneous points of view (Top, Back, Right, and a forced-perspective view), each looking at the same portion of the workspace from a different angle. These are not unlike Layout's viewports. Changes made in one view are immediately updated in the others.

### Viewport Titlebar

You can change the **View Type** and **Rendering Style** of a viewport without going to the **Display Options Panel** by using the pop-up menus on the left side of each viewport titlebar.



The buttons on the right side affect panning, rotating, zooming, and minimising/maximising your view.



## 9. Info Display

Just above the grid display is an information display. Most of the time this gives you feedback on the position of your mouse. However, depending on what you are doing, it can also display a variety of other information.

## 10. Grid Display

In the bottom-left corner of the screen you can see the grid size information display. As in Layout, this tells you how big each square is in the Modeler workspace.

## 11. Selection Info

Above them is the selection info display, which tells you how many points/polygons/edges you have selected.

## 12. Edit Mode

To the right of the grid display are the three mode buttons. These are mutually exclusive buttons, so only one can be active at any one time. These determine when you are editing points, edges or polygons.

## 13. Progress Bar

Displays tools processing.

## 14. Symmetry

The Symmetry mode (**Symmetry** button, located along bottom edge) not only works on selection, but also on editing. Operations

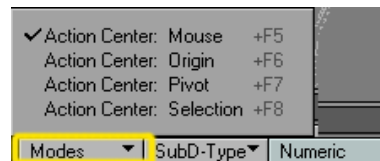
on the positive side of the X axis also inversely affect the negative side of the X axis. When this mode is active, your object is theoretically split in half at X=0.

## 15. Tool Tips

Displays helpful information for current tool.

## 16. Mode Menu

Make particular note of the **Mode** menu along the bottom border. This button contains some very powerful features like the ability to change a tool's Action Center. The Action Center is the point of origin from which the tool's action is applied to modify the geometry.



## 17. SubD-Type

Allows you to switch between two SubDivision modes, Subpatch and Catmull-Clark.

## 18. Statistics Panels

**Numeric Panel** — You can enter numeric values for every tool in Modeler. Access it at the bottom of the interface.

**Statistics Panel** — The **Statistics Panel** includes some tools that makes selecting points and polygons far easier.

**Info Panel** — From here you can get detailed information on points and polygons and even edit the data on a point-by-point basis.

## 19. Change Surface

Activates the **Change Surface Panel** to set the surface name and basic attributes for the selected polygons. See **Surface Editor** for more information.

## 20. Make tool

The **'Make'** tool is a tool that finalises your temporary adjustments to the initial creation of the geometry.

## 21. Vertex Maps

**VMaps** can be created using the buttons in the lower-right corner of Modeler. In a way you can think of Vmaps as "surfaces" for points. It's a way of tagging some information onto points that allow much more control of various Lightwave functions on a point level.

**W** - Weight, **T** - UV Texture, **M** - Morph, **C** - Color, **S** - Selection Set.



## File Menu

### Starting a New Object

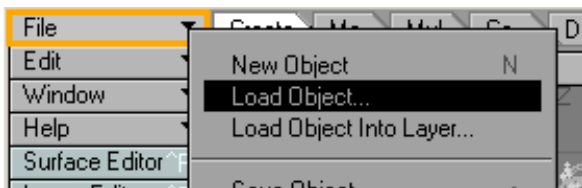
(default keyboard shortcut **Shift N**)

To start editing a new object from scratch, select **File > New Object**. A new object called UNNAMED is created with empty layers.

### Loading an Existing Object from Disk

(default keyboard shortcut **Ctrl O**)

Objects can be created, loaded into Layout, or loaded directly into Modeler.



To load an object into Modeler, select **File > Load Object**. The object becomes the current object and all of its layers are loaded.



**NOTE:** You may select multiple objects to load at once, by selecting more than one object at once in the file requester.

Choose **File > Load Object into Layer** to load an object file into the currently selected layer. If the file contains multiple layers, they will all be loaded into the same layer.

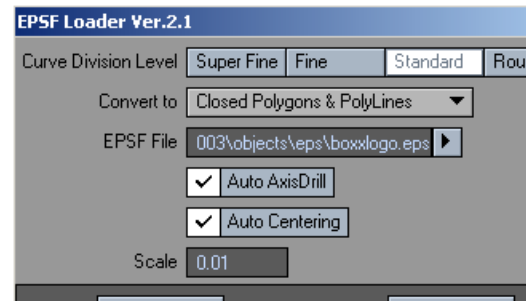
Choose **File > Recent Files** to reload an object that you had recently opened.

Choose **File > Revert Current Object** to put the current object in its last-saved state. Essentially, this is like closing and reloading the object.

## Importing an Object

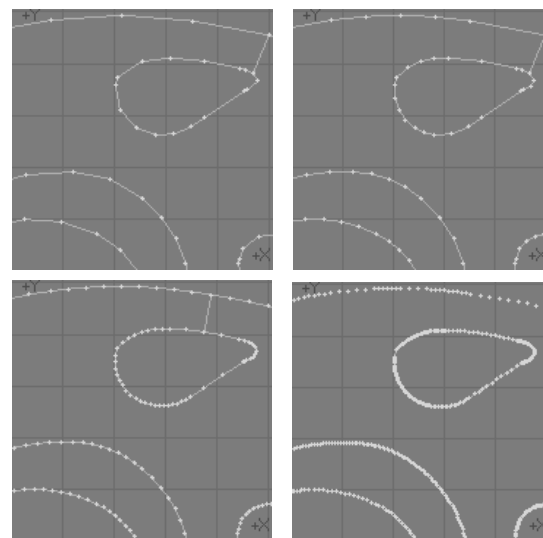
### Encapsulated PostScript Loader

The EPSF Loader command (**File > Import > EPSF Loader**) converts Illustrator (.ai) and encapsulated PostScript (.eps) files into two-dimensional LightWave objects, by converting bezier curves into tessellated polygons.



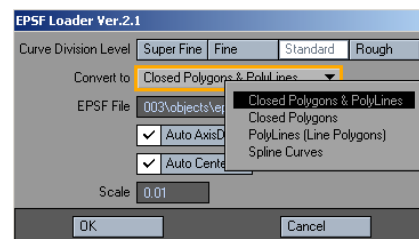
**NOTE:** If the EPS uses a gradient fill, you will need to convert it to a solid fill (e.g., using Illustrator) before using this loader.

The **Curve Division Level** option sets the desired level of detail.



Curve Level — Rough, Standard, Fine, Super Fine

The **Convert to** pop-up menu sets the type of object that will be created.



**Closed Polygons & Poly Lines** — Will create a 2-point polygon chain and Polygons where vector lines are in the file.

**Closed Polygons** — Will create closed polygons where vector lines are in the file.



**PolyLines (Line Polygons)** — Will create a 2-point polygon chain where vector lines are in the file.

**Spline Curves** — Will create splines where vector lines are in the file.

The EPS file is defined by either typing the full path in the **EPSF File** input field or using the requester button to bring up a file requester.

**Auto Axis Drill** will attempt to automatically drill holes in faces, where appropriate. (e.g., the two holes in the number 8).



**NOTE:** If this option is not checked, the geometry will be created and you will have to manually create the hole.

**Auto Centering** will center the resulting geometry.



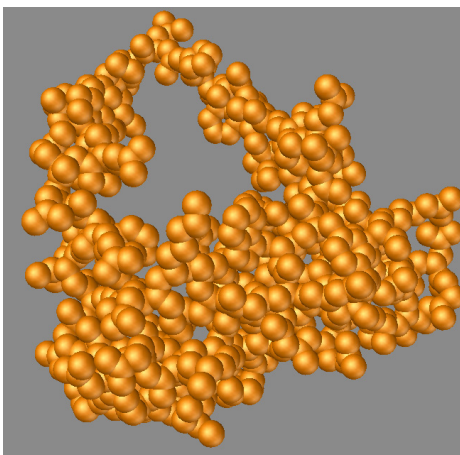
**NOTE:** Due to the inherent differences between 2D structured drawings and 3D objects, there will usually be some amount of cleanup required.



**NOTE:** This command has been tested with Illustrator files up to version 8.

## Pdbreaderv3-

The Protein Data Bank (PDB) is an archive of experimentally determined three-dimensional structures of biological macromolecules, serving a global community of researchers, educators, and students. The archives contain atomic coordinates, bibliographic citations, primary and secondary structure information, as well as crystallographic structure factors and NMR experimental data.



This importer will read the PDB data and convert it to a LightWave Object.

**Sean's PDB Reader**

1359 ATOM Entries 43 HET Entries 1402 Total Entries

Sample ATOM Line: 12 Fields  
 ATOM 1 N GLN 1 2.595 -13.139 61.633 1.00 0.00  
 Enter ATOM Values

Field to read for X:   
 Field to read for Y:   
 Field to read for Z:   
 Field to read Atom Info:

Sample HET Line: 11 Fields  
 HETATM 1361 CHB BLV 1 3.008 0.752 56.860 1.00 0.0  
 Enter HET Values

Field to read for X:   
 Field to read for Y:   
 Field to read for Z:   
 Field to read Atom Info:

Relative Scale:   
 Sphere Tess Level:

More information about the PDB can be found here: <http://www.rcsb.org/pdb>



## Vertex Loader

Use **Vertex Loader** in order to edit **Weight Maps** for bones that were created from scratch in Layout, or bones that were converted from **Skelegons**.

To use **Vertex Loader**, choose **File > Import > Vertex Loader**. First the Load Scene requester opens; select a scene file that has object(s) with bones. After you select a scene file, the following dialog box appears. There you select an object that you want to edit and click **OK**.



The selected object is automatically loaded into Modeler and displayed in **Vertex Paint** with its bones. If the object is already loaded, only the bones are loaded from the scene file. **Weight Maps** set on the **Bone Properties Panel** in Layout are used for each corresponding bone.



**NOTE:** Before launching **Vertex Loader**, please make sure Layout's **Content Directory** setting matches Modeler's.

## Saving Object Files

When you save an object, all layers are saved and preserved. You have several options which are all available from the **File** menu:

**Save Object** (default keyboard shortcut **S**)— This option saves the current object using its filename. If the object has never been saved before, this is the same as using **Save Object As**.

**Save Object As** (default keyboard shortcut **Ctrl S**)— This option opens a file dialog letting you name (or rename) the object before saving.

**Save Layers As Object**— Use this to save a foreground layer to its own object file. If multiple layers are selected, they will be saved in a single layer. Make sure you add the .lwo filename extension to your object name when saving.

**Save All Objects**— This option saves all files using their filenames. If any new objects haven't been saved before, you are alerted and given the option of naming them.

**Save Incremental** (default keyboard shortcut **Shift S**)— This option will give you the ability to save a version of an object by adding incremental numbers to the object's name. This is handy when you want to save your object in various stages of the Modeling process.



**NOTE:** An asterisk will appear next to the names (on the current pop-up menu) of objects that have been modified since their last save.





## Closing Object Files

Once you are finished with an object, you can close it by selecting **Close Object** from the **Files** pop-up menu. You can close all object files by selecting **Close All Objects**. If any objects are unsaved, you are alerted and given the option of saving them.



**WARNING:** Objects revert to their on-disk version in other Hub applications (e.g., Layout) when you close them or exit Modeler and do not save them first. This will cause you to lose unsaved attributes like surface changes.

## Sections

**Save-Object-Section** — You can use this to save a selected polygon as an object

**Save-Object-Section-Cut** — Saves the selected object section, and then cuts it from the object itself.

**Save-Object-Section-Points** — Use this plugin to save selected points of an object.

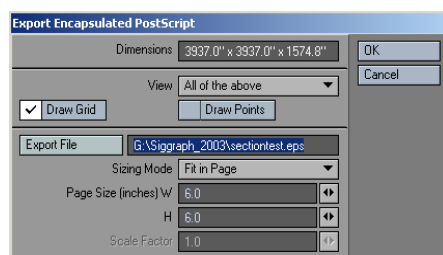
**Save-Object-Section-Points-Cut** — Saves the selected points, then cuts them from the object itself.

## Exporting Objects

The **File > Export** menu has several options to export your LightWave object into different formats.

### Export Encapsulated Postscript

This plugin will allow you to save your object as an encapsulated postscript file (.EPS). In the **View** pop-up menu, select which of your viewports you want to include in the file. The Texture UV View is especially useful for exporting UV projections to use as templates in a paint program. If you want to include the grid or points, activate **Draw Grid** or **Draw Points**.



Enter the path and filename in the file input field or click the **Export File** button to bring up a file dialog. Define the size in inches of the resulting drawing using the **W** (width) and **H** (Height) field.

**Export 3DS** — Use this to Export to AutoDesk 3DS file format. All polygons must be triangulated for this format.

**Export DXF** — Use this to Export to DFX file format. DXF is the standard format for 3D objects and is used by most

programs that support 3D content. DXF files can't contain polygons with more than four points.

**Export LightWave 5** — Use this to Export to LightWave's old Object file format. Anything after LightWave 5.7 uses current LWO file format.

**Export OBJ** — Use this to convert a LightWave object to Wavefront .obj file format specification.

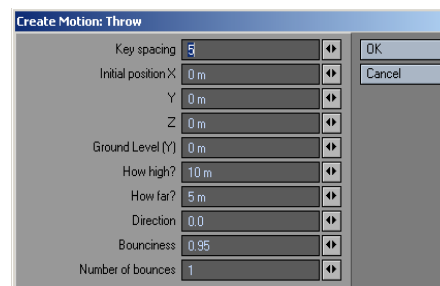


**Note:** UV maps are not preserved in Export OBJ unless you follow these steps:

1. Load the object
2. Give it a UV texture map.;
3. Hit F5 and open the Surface Editor and hit the T next to the Color channel. In the Texture Editor, leave the texture type as Image Map, select UV as Projection and the UV map you created. You do not need to load an image;
4. Export as OBJ and then reload the object. You will see that the UV texture map is preserved, albeit renamed..

**Path to Motion** — takes a spline from Modeler and makes it into a motion path for Layout.

**Throw** — calculates trajectory and bounce on given item and inputs — output is a motion file for Layout.



## Quit

(default keyboard shortcut **Shift Q**)

Quit will shut down LightWave Modeler.



## Edit Menu

### Basic Point, Edge, and Polygon Maintenance

#### Cut, Copy, Paste, Paste in Place, Delete

The basic steps to copy and paste points, edges, and polygons are essentially the same. First, you select the points, edges, or polygons. Then you either click the **Copy** button (**Ctrl C**) to copy the selection into the memory buffer, retaining the original, or click the **Cut** (**Ctrl X**) button to delete the selection. You can also use the **Del** key to run the **Delete** command; it is similar to Cut, but does not copy the data to memory for future pasting.

Once something is in the memory buffer, you can click the **Paste in Place** tool (**Edit > Paste in Place**) (**Ctrl V**) to paste the contents back into the layer. The memory buffer is not reset after pasting. Thus, you can paste the contents multiple times. The buffer contents change only when they are replaced by a subsequent cut or copy. For interactive pasting, use the **Paste** tool (**Edit > Paste**).



**NOTE:** Geometry cut (or copied) to the memory buffer loses its association with geometry it is cut away from, even when it is pasted back in.

## Undoing Operations

LightWave Modeler offers an undo function that lets you take back the last change you made while working on an object. The **Undo** button (**Ctrl Z**) is located in the **Edit** drop down menu. The number of undos is determined by the **Undo Levels** setting found in the **General Options Panel**. The maximum value is 128, but you can also be limited by your computer system RAM. However, minimally-configured systems should be able to go back several steps if necessary. Even power-users should find 10 undo levels sufficient.

## Redo

It's possible to go too far unintentionally and undo too many steps. To correct this error, click the **Redo** (**Z**) button located in the **Edit** drop down menu. As long as you have made no editing changes, you can redo all the way back to your last Modeling operation.

The combination of **Undo** and **Redo** is a powerful tool, allowing you to step backward and forward through the evolution of a model. This is especially useful when you use some of the more advanced Modeling tools, and you may need to make a series of subtle changes to a model as you design it by undoing and redoing as needed until you achieve the desired shape.



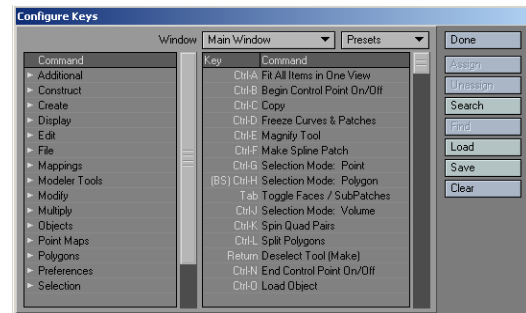
**WARNING:** If you undo back five steps, then use a new tool, you will no longer be able to redo those steps. Undo is like history that unfolds as you model. If you go back and change the model, then you have changed the course of history in a fundamental way. The structure of the model is now different, and redo will be unable to restore it.

## Customizing Keyboard Shortcuts



**WARNING:** We strongly suggest that you keep the default keyboard mapping assignments and make new assignments only to unmapped keys. Otherwise, technical support and using the documentation may become difficult.

Like the menus, keyboard shortcuts can be configured to suit your own needs. To display the **Configure Keys Panel**, choose **Layout / Modeler > Edit > Edit Keyboard Shortcuts** or the default keyboard shortcut of **Alt F9**.



The window on the left contains a complete list of all assignable commands grouped by type. The right window shows a complete list of all keystrokes and assigned commands, if applicable.

#### To assign a command to a key:

Select a command in the left window.

Select the target key in the right window. (You can hit the desired key or keystroke combo to quickly select the key.)

Click the **Assign** button. This will overwrite any existing assignment. Alternatively, you may drag the command to the right window.

Commands can also be dragged and dropped into the proper key slot on the right.

#### To unassign a command from a key:

Select the target key in the right window.

Click the **Unassign** button.

## Finding Assignments and Commands

If you have a command selected in the left window and click **Find**, the assigned key, if any, on the right will become selected. Selecting an assigned key in the right window and clicking **Find** will select the matching command in the left window.

Clicking **Search** allows you to type in a search phrase to find a command. Note that this will only find the first instance and it is case sensitive.



**NOTE:** Keyboard shortcuts can be assigned to only one command. Assigning a command that is already assigned to a different key will assign it to the new key, but remove it from the old one.





## Maintaining Key Mapping Sets

The **Load** and **Save** buttons let you retrieve and store key mapping sets that you develop.

The **Clear** button clears out all assignments, so use with caution.



**NOTE:** In Modeler, you can use **Modeler > Preferences > Revert to Startup Preferences** to restore your initial settings; however, any changes since you first ran the application will not be reflected.

Any available preset key mapping sets can be chosen from the **Presets** pop-up menu. The **Window** pop-up menu lets you select different menus to edit (e.g., **Graph Editor**), if available.

## Edit Menu Layout

(default keyboard shortcut **Alt F10**)

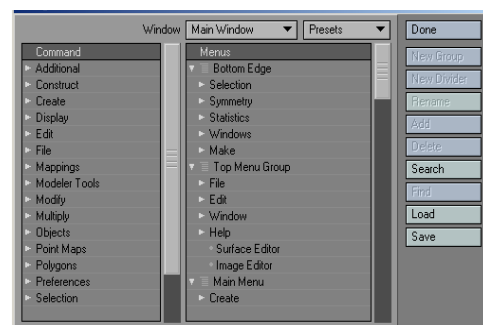


**WARNING:** We strongly suggest that you keep the default menu organisation intact. Otherwise, technical support and using the documentation may become difficult. A better solution is to create a new menu tab and/or groups and place your frequently used tools in them.

LightWave menus are customisable. You can add, remove, group, and reorganise commands.

Choose **Layout/Modeler > Edit > Edit Menu Layout** to open the **Configure Menus Panel**. In the Command list on the left, a list of available commands will appear. These will be grouped by type.

In the Menus list on the right, several main sections have sub-items, which are indented to show the hierarchical relationships. **Top Menu Group** contains the items that are always visible no matter which tab is selected. **Main Menu** items are the main tabs and related buttons for the main interface toolbar. **Bottom Edge** is for Modeler only and relates to the controls along the bottom edge of the screen. **Left, Middle, and Right Mouse** button menu (when available) are the menus that appear when the **Shift + Ctrl** keys are held down along with the corresponding mouse button. There may be other menu sections defined.



In the Menus window, dots indicate commands while arrows indicate groups of commands/sub-groups beneath it. If the arrow points to the right, the group's sub-menu items are collapsed and not visible. To reveal the sub-menu items for a collapsed group, simply click on its right-facing arrow. To collapse an open group, click on its downward-facing arrow.

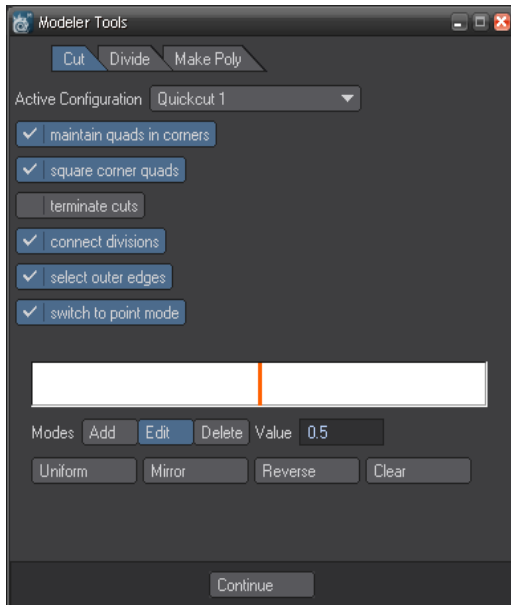


**Note:** LightWave 9.0 introduced the Studio Production Style Layout. These are keyboard shortcuts and menus developed by production studios. If you like the new menus, you can use them, just remember the documentation is geared toward the default menus



## Edit Modeler Tools

This **Options Panel** gives you the ability to set new default settings on **Cut**, **Divide**, and **Make Poly**. Also, this is where you will need to set the settings for the three **Quick Cut** tools. By default the **Quick Cut** tools all have the same setting and you will need to define them in this panel.



## Custom Preferences

You can use **Edit > Import Preferences File** and **Edit > Export Preferences File** to load and save, respectively, **Preference** settings. Use this feature to access special-purpose **Preference** settings you might set up. You can also use **Edit > Revert to Startup Preferences** to go back to the settings that existed when you first started Modeler.



**NOTE:** The **Preference** settings that exist when you shutdown Modeler will become the startup settings for your next session.

General Options — See General Options Section below.

Display Options — See Display Options Section, starting on the next page.

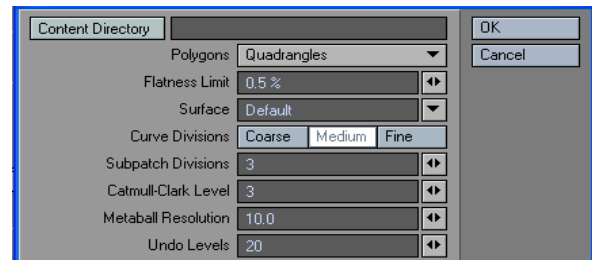
Backdrop Options — Part of the Display Options.

Recent Content Directories — The Content Directory is a central file path for loading objects, images, and scenes. Note that this setting is shared by all LightWave applications. This drop down menu will store recently used Content Directories to help gain access to past projects.

## General Options

(default keyboard shortcut **O**)

You can access the **General Options Panel** by choosing **Edit > General Options**.



### Content Directory

The **Content Directory** is a central file path for loading objects, images, and scenes. Note that this setting is shared by all LightWave applications.

### Default Polygon Type

Some operations in Modeler create many polygons in one operation. As such, the computer must decide whether to generate triangles (three-point polygons) or quadrangles (four-point polygons), depending on the shape's requirements.

To force the computer to use **Triangles** or **Quadrangles**, select either for the **Polygon** setting. Choose **Automatic** to let the computer create the most appropriate polygon type for each given operation.



**HINT:** To create optimal objects, your goal is to keep the polygon count down. They will load and render faster than bloated objects, with no visible difference. As such, Quadrangles is the suggested setting for most cases. This lets you manually choose the areas that require triangles.

### Flatness Limit

The **Flatness Limit** setting determines whether Modeler regards a polygon as planar or not. Note that a non-flat polygon considered planar according to this setting can still cause rendering errors.

### The Default Surface Name

All polygons must have a surface name. When geometry is created, polygons are given the default surface name of **DEFAULT**. You can change the default name by changing the **Surface** field. This will affect only geometry created from that point on, however. Use the pop-up to the right of the field to select from existing surface names.



**NOTE:** When you create a surface using the Change Surface dialog (**Detail > Polygons: Surface**), the option (**Make Default**) automatically sets the default surface name.



## Curve Divisions

The **Curve Divisions** setting determines how smoothly a curve (e.g., spline curves, text, etc.) should be interpolated. The finer the setting, the greater the number of polygons used, and the smoother the resulting curve division.

## Subpatch Divisions

When a SubPatch object is frozen with the Freeze command (**Ctrl D**), it is converted into a polygonal object. The **Patch Division** setting determines the level of detail used in the resulting object. The number entered in the **Patch Division** field must be 1 or greater and is restricted to whole numbers.

The number of polygons per SubPatch surface will be equal to the square of the **Patch Division** number. For example, if set to 4, each SubPatch surface will be converted into 16 polygons arranged in a 4 by 4 array. A setting of 2 would result in 4 polygons arranged in a 2 by 2 array.

The higher the setting, the higher the number of polygons used. Because of LightWave's surface smoothing capabilities, you can often get away with a setting of 2 and sometimes even 1, which will keep the polygon count of your objects to a minimum.



**NOTE:** Setting this value too high with heavy-polygon SubPatch models can cause the performance of Modeler to degrade. If Modeler appears to be running sluggishly when working with SubPatch models, lower this value to help improve performance.

## Sub-D Type

### Subpatch

Subpatches the number of polygons increases linearly with the level.

### Catmull-Clark

Every level of Catmull-Clark subdivision quadruples the number of polygons

## Metaball Resolution

The **Metaball Resolution** setting determines the amount of detail used to display metaballs.

## Symmetry Tolerance

Sets the amount matching elements can differ on opposite sides of the symmetry for symmetry to work.

## Image Cache Maximum

If enabled, sets the amount of caching available for images in megabytes.

## OBJ Tab

The OBJ tab contains options for importing and exporting OBJ file objects.

OBJ One Layer imports the object as a single layer.

OBJ One VMap imports the object with a single vertex map.

OBJ Pivot at Center creates the pivot of the object at the center of the object.

OBJ Write Normals writes the normals associated with the object when saved.

OBJ Merge Points merges points sharing the same space.

OBJ ZBrush Mode exports the object to work with ZBrush.

OBJ Import Scale sets the scale of an object when it is imported.

OBJ Export Scale sets the scale of the object when it is exported.

## Color Space Tab

Color space conversion is now performed in 4 places in LightWave.

1. On loading, an image can be converted from its native color space format to linear.
2. When sent to the Image Viewer, an image can be converted from linear to another color space.
3. When saved from the renderer, an image can be converted from linear to another color space.
4. When picked from the Color Picker, a color can be converted from and then to linear color space.

The color space defaults are set up on the Layout Preferences panel.

When an option is selected in the pop-up it becomes the current item for that selection.

Color tables are added to the selections, as if they were built-in.

A color table can be loaded by using Load Table from the pop-up. The color tables are stored in the project directory, in a directory called Color Tables.

In the Image Editor, there is a check box, Treat as Alpha. This means use the alpha channel from the Preferences Default setting, not the alpha channel from the color space in the Image Editor. This is an override on the alpha color space selected in the Image Editor.

One can set the color correction for the Viewer, files with palettes, 8-bit files, floating point files, alpha channel and the color picker from the preference default panel and the Modeler General Options panel.

When an image is loaded, if the file setting for that image is set to default, then an attempt is made to look at the metadata setting for that image. If the metadata settings have the color space the image was saved in, then it is used. For example, jpegs have metadata setting for sRGB and Adobe 1998 linear format.



The built-in color spaces are:

1. Linear, LightWave linear color space.
2. sRGB, Standard RGB color space.
3. rec709, BT.709, HDTV
4. Cineon, Eastman Kodak Co.

The color lookup tables come in 2 formats:

1. LightWave color table format.
2. 3D LUT format.

LightWave color table format is as follows:

Code:

```
EGA
17, -0.5, 1.5
; Convert on load.
-0.5, -0.5, -0.5, -0.5
-0.375, -0.375, -0.375, -0.375
-0.25, -0.25, -0.25, -0.25
-0.125, -0.125, -0.125, -0.125
0.0, 0.0, 0.0, 0.0
0.125, 0.125, 0.125, 0.125
0.25, 0.25, 0.25, 0.25
0.375, 0.375, 0.375, 0.375
0.5, 0.5, 0.5, 0.5
0.625, 0.625, 0.625, 0.625
0.75, 0.75, 0.75, 0.75
0.875, 0.875, 0.875, 0.875
1.0, 1.0, 1.0, 1.0
1.125, 1.125, 1.125, 1.125
1.25, 1.25, 1.25, 1.25
1.375, 1.375, 1.375, 1.375
1.5, 1.5, 1.5, 1.5
; Convert on save.
-0.5, -0.5, -0.5, -0.5
-0.375, -0.375, -0.375, -0.375
-0.25, -0.25, -0.25, -0.25
-0.125, -0.125, -0.125, -0.125
0.0, 0.0, 0.0, 0.0
0.125, 0.125, 0.125, 0.125
0.25, 0.25, 0.25, 0.25
0.375, 0.375, 0.375, 0.375
0.5, 0.5, 0.5, 0.5
0.625, 0.625, 0.625, 0.625
```

0.75, 0.75, 0.75, 0.75

0.875, 0.875, 0.875, 0.875

1.0, 1.0, 1.0, 1.0

1.125, 1.125, 1.125, 1.125

1.25, 1.25, 1.25, 1.25

1.375, 1.375, 1.375, 1.375

1.5, 1.5, 1.5, 1.5

Where:

EGA is the name of the color space that appears in the pop-up.

17 is the number of entries in the color table.

-0.5 is the lower range of the look-up table ( black ).

1.5 is the upper range of the look-up table ( white ).

-0.5, -0.5, -0.5, -0.5 are the rgba values for the table entry.

Blank lines and lines beginning with comments, are skipped over.

3D LUT format tables are read in and a reverse lookup table is made.  
The name of the color space is the name of the file on the pop-up.

Auto Sense on Load detects the Color Space settings and uses those  
when a scene is loaded.

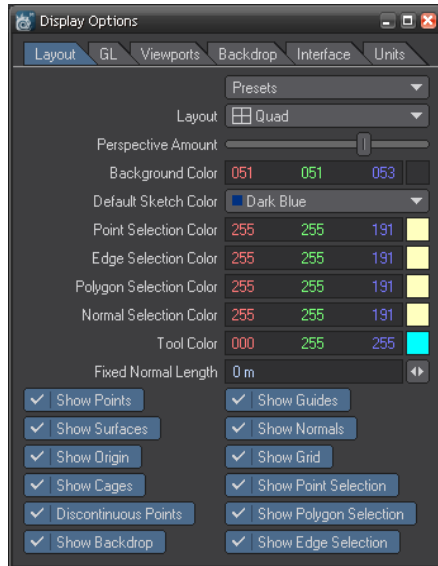
Color Correct OpenGL will color correct the OpenGL viewports.



## Display Options

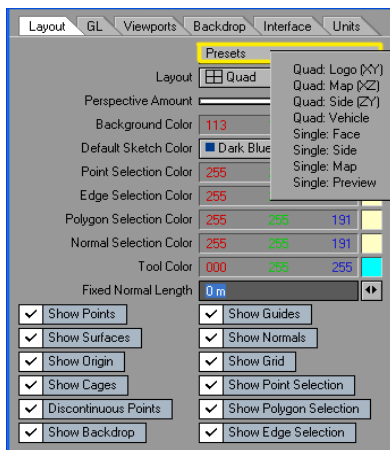
(default keyboard shortcut **D**)

Modeler's interface is highly configurable allowing to tailor it to your specific needs. You can change the arrangement of viewports as well as change the display characteristics for each viewport independently. Choose **Edit > Display Options** to bring up the **Display Options Panel**.

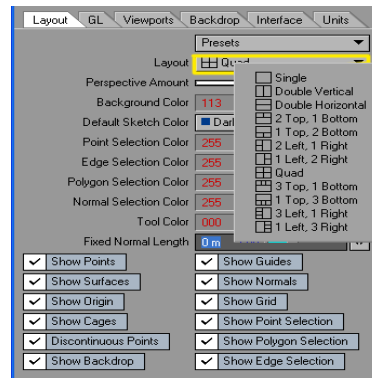


### Layout Tab

Layout options control the global arrangement of the viewports, the number of viewports and some other **Global Display** settings. Use the **Presets** pop-up menu to quickly set common layout configurations (**Quad** is the default setting). Selecting a Preset will affect the **Layout** setting, as well as settings on the **Viewports Tab**.



If you want to set up your own custom layout, make a selection from the **Layout** pop-up menu. The icon to the left of the description will give you an idea of how the viewports will be arranged. Later, you will see that you have total control over what is in each viewport. In fact, each **Layout** option will remember the **Viewport** settings as they were the last time that **Layout** option was used.



### Texture Resolution

The **Texture Resolution** setting determines the resolution to use for displaying textures in viewports that have their **Rendering Style** set to **Texture**. Higher settings increase the detail of image textures displayed in viewports, but increase memory usage and display refreshing time.

### Perspective Amount

The amount of perspective in the perspective view can be varied by changing the **Perspective Amount** slider on the **Layout Tab**. You can go from a very wide-angle to a flat nearly orthogonal view. The setting is global and affects all the perspective views the same way.

### Background Color

Use the **Background Color** selector to change the color that appears in the background for viewports using a shaded display.

### Default Sketch Color

Use the **Default Sketch Color** setting to change the color that the wireframe will appear in the various view modes. You can always change the Wireframe color on polygons using the **Sketch Color** tool located under the **Detail Tab**.

### Selection Color Options

Use the Selection Color Palette for editing Point/Edge/ Polygons when selected. Other options include changing the color for Normals and the color of active Tools.

### Fixed Normal Length

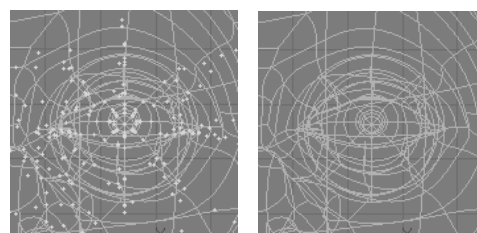
Change the length of the normal here.

### Show Options

The various **Show Visibility** options let you independently set what you want to see globally in your viewports.

### Show Points

Points appear as small dots.

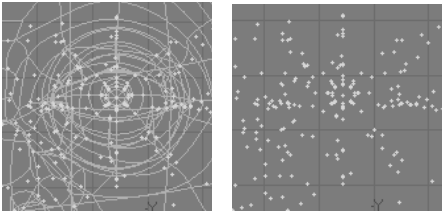


Left: Show Points On, Right: Show Points Off



## Show Surfaces

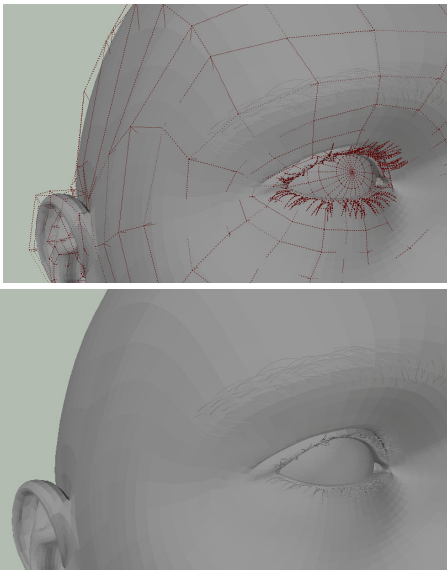
Surfaces are the polygon (or SubPatch) surfaces.



Left: Show Surfaces On, Right: Show Surfaces Off

## Show Cages

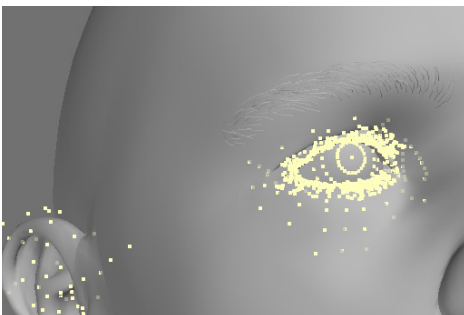
A cage is the outline that connects all of the control points when you edit a SubPatch object.



Top: Show Cages On, Bottom: Show Cages Off

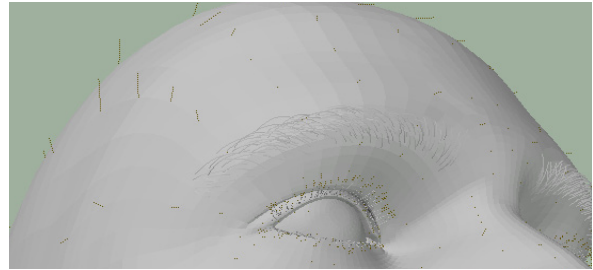
## Show Point Selection

When points are selected they appear highlighted if this option is selected.



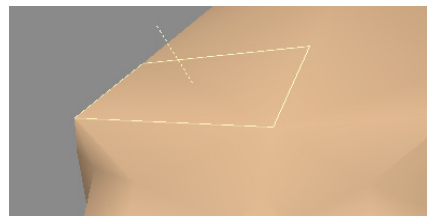
## Show Guides

Guides are the dotted-lines that extend from the surface (patch vertex) to the control points on the cage (if visible) when you edit a SubPatch object.



## Show Normals

Normals are dotted lines that extend perpendicularly from selected (planar) polygons and indicate the direction a polygon is facing.

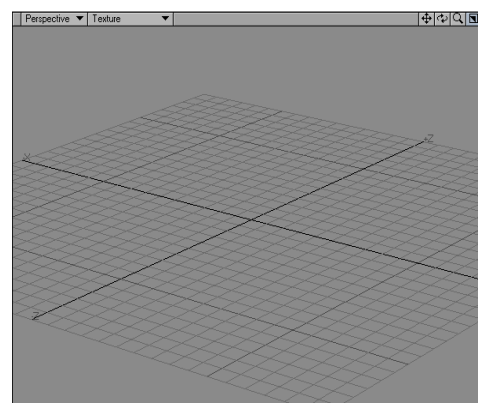


## Show Origin

This option will show the 0 axis in the viewports. For the perspective view, only the X and Z axis are shown.

## Show Grid

The grid is the background reference grid.

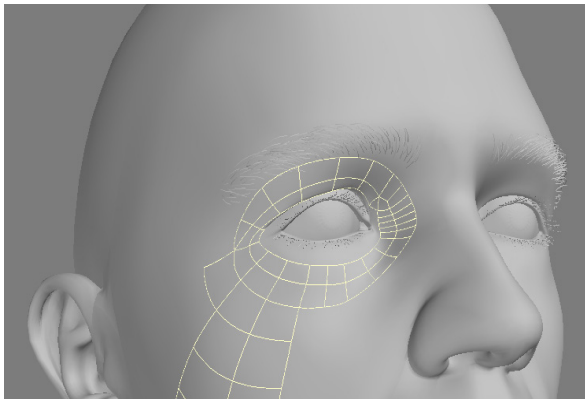






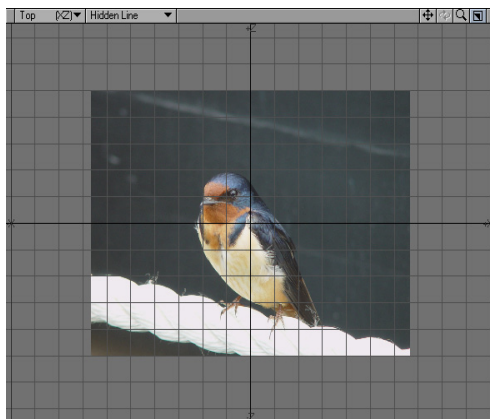
## Show Polygon Selection

When polygons are selected they appear highlighted if this option is selected.



## Show Backdrop

A Backdrop is an image loaded for viewports on the **Backdrop Tab**. These images can be useful as Modeling reference.



1



**NOTE:** These visibility options can be overridden independently for each viewport on the **Viewports Tab**.

## Toggle Background Display

A new command for turning on/off the background image display. Currently, this command must be added either as a menu option or a keyboard shortcut.

## GL Tab

**Texture Resolution** determines the display resolution of images used. Lower settings will update faster and use fewer system resources.

## Shading Method

### OpenGL Multi- Texturing

De- /Activates Multi-texturing – multiple textures layers per polygon in OpenGL.

Depending on the settings that you activate (see below), the following combinations of texture layers are possible:

- A. Two color-layers with one diffuse-layer, one luminosity-layer and one reflection map (5 textures/polygon).
- B. One color-layer with one transparency-layer and one reflection map (3 textures/polygon).

Currently the Multitexturing is made to work with graphics-cards with at least two texture memory units.

## GLSL HW Shading

GLSL HW Shading (OpenGL Hardware Shading) is now among the selections in the Display Options panel. Support for the OpenGL 2.0 hardware shader technology in newer video cards has been added to Layout to provide very close approximations of render functions in the viewport displays. Light falloff, surface blending, gradients, and many procedurals can now be displayed in the OpenGL viewports in Layout when GLSL HW Shading is turned on.

## Geometry Acceleration

Determines how the graphics card displays OpenGL. Streaming renders the mesh immediately to screen, using the lowest amount of memory at the cost of speed. Buffered(VBO) will attempt to store the geometry in graphics card memory, allowing for the highest speed, at the cost of memory. In cases where the mesh or shading changes with every frame no caching is possible, a fallback to the Streaming method will result, for example with animated meshes and reflection maps. Smooth shaded geometry will benefit the most from the Buffered(VBO) mode. If the mesh is buffered in graphics card memory the performance you will get as much performance as your graphics card can give you.

## Mipmap

Mipmapping is similar to what is used in today's games to avoid graininess of textures in a distance or at a flat angle. Basically lower- res versions of the texture are generated in realtime and blended in. This feature is supported in hardware by most of today's graphics cards. This feature also works if Mutitexturing is turned off. Please note however that due to the nature of this filtering method, low-resolution textures may appear a bit blurry.

## Color Channel

De- /Activates the display of textures in the Color Channel if Multi-texturing is on.

## Diffuse Channel

De- /Activates the display of textures in the Diffuse Channel if Multi-texturing is on.





## Transparency Channel

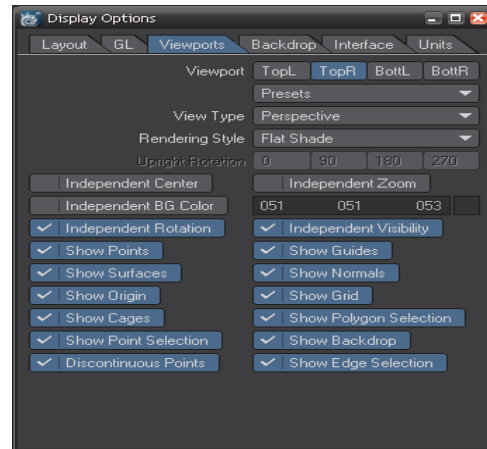
De- /Activates the display of textures in the Transparency Channel if Multi-texturing is on.

## Luminosity Channel

De- /Activates the display of textures in the Luminosity Channel if Multi-texturing is on.

## Viewports Tab

**Viewport** options let you change how objects are displayed in each viewport. To change the settings for a viewport, first select it using the **Viewport** buttons.



Use the **Presets** pop-up menu to quickly set the options on this tab for the selected viewport region. The selections are listed by view types, but will affect all of the settings on this tab.

The **View Type** pop-up menu determines the editing axes you want to use for the selected region. For the **Orthogonal** settings, the names generally indicate the viewing perspective. **Back (XY)**, for example, lets you edit along the X and Y axes. This means you are looking along the Z axis. Since it is called Back, that means your perspective is from the back (i.e., negative side) of the Z axis, looking toward the positive side. **UV Texture** is entirely different — it is not used as a Modeling viewport but for adjusting UV texture maps.



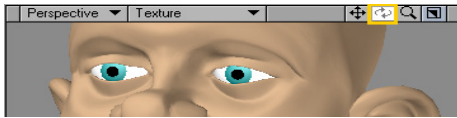
**NOTE:** There are four Display (group) commands assigned by default to certain keyboard shortcuts which affect the viewport beneath your mouse pointer. **Shift F1** toggles the view between Back and Front, **Shift F2** toggles between Top and Bottom, **Shift F3** toggles between Right and Left, and **Shift F4** toggles between Perspective and UV.

## Perspective

You may edit in a Perspective viewport, just as you would in any other, but you may also rotate your view perspective. This is Modeler's virtual trackball, which enables you to rotate the object without affecting its orientation in the other three edit windows. It effectively provides you with three axes of rotation. While holding down the **Alt** key, you can perform these actions:

- Rotate around the X axis (pitch) by dragging up or down directly across the verticle center of the viewport
- Rotate around the Y axis (heading) by dragging left or right directly across the horizontal center of the viewport.
- Rotate around the Z axis (bank) by dragging to the left or right around the perimeter

If the title bars are visible, you can drag the rotate drag button with your **LMB** to rotate. If you use the **RMB**, the viewport will rotate around its perpendicular axis. Holding the **CTRL** key (or using your **MMB**) will cause rotations to snap to 15-degree increments.



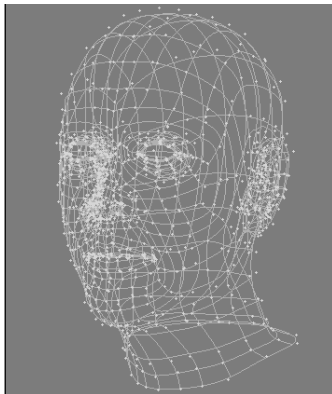
**HINT:** Imagine you are manipulating an imaginary ball with your mouse when you manipulate the Preview window.

## Rendering Style

The **Rendering Style** pop-up menu determines the style of display you want to use for the selected region.

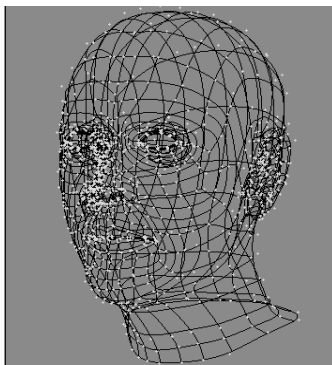
### Wireframe

Although the **Wireframe** is arguably the most limited display mode, it is the most commonly used **Rendering Style** because of its ease in viewing, selecting, and unselecting points and polygons.



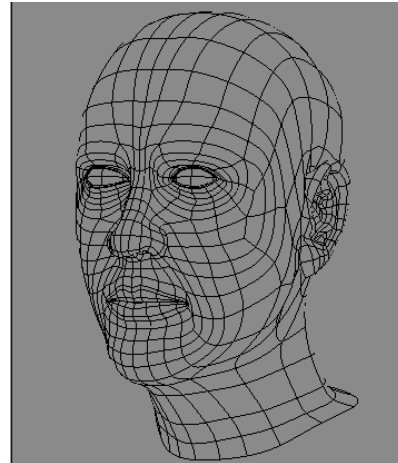
### Color Wireframe

This is nearly the same as **Wireframe**, except the polygon edges are drawn using their Sketch color (**Detail > Polygons: Sketch Color**).



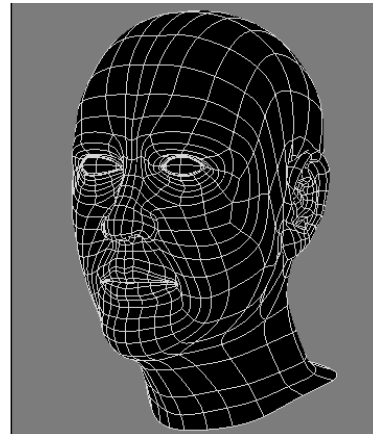
### Hidden Line

**Hidden Line** shows an object's Front Face polygons in wireframe view.



### Sketch

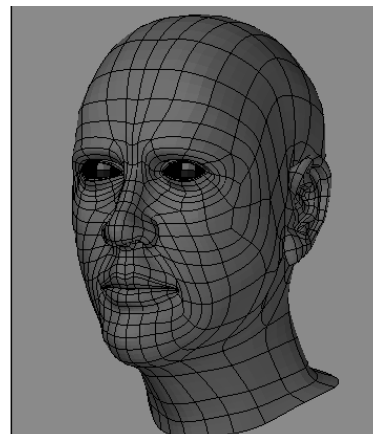
**Sketch** shows an object in a combined wireframe and flat-shaded view. All polygon edges are drawn, but faces are also visible. This mode does not account for **Surface** settings, however. The polygon edges are always drawn in white and faces are grey. Background layers are visible.



You can change the polygon shading color in this mode by choosing **Detail > Polygons: Sketch Color**.

### Wireframe Shade

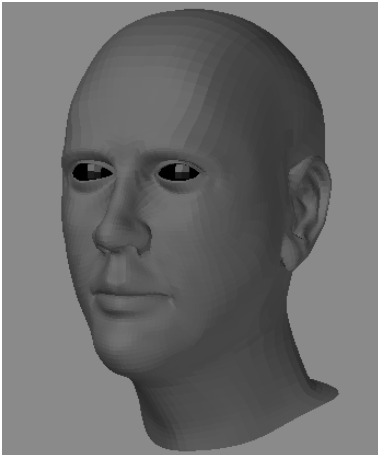
**Wireframe Shade** is a smooth-shaded mode that overlays the wireframe lines.





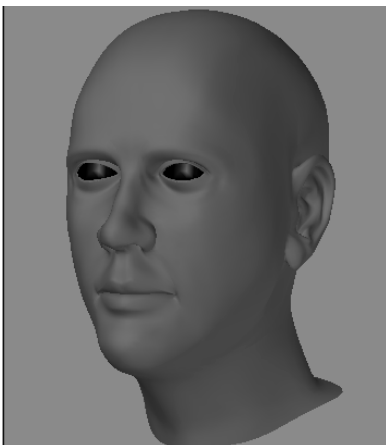
### Flat Shade

In **Flat Shade** mode, the object is shown as a flat-shaded solid. This mode supports some **Surface** settings, but not smoothing.



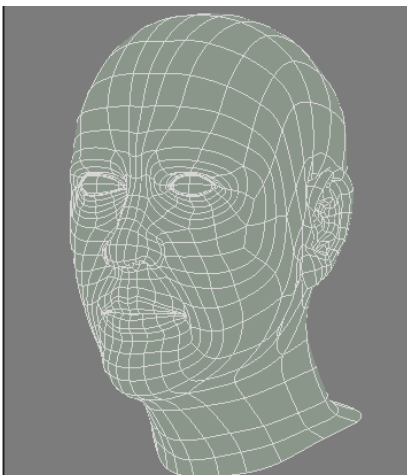
### Smooth Shade

**Smooth Shade** is a smooth-shaded mode that supports some **Surface** settings, such as **Color**, **Diffusion**, **Specularity**, **Glossiness**, **Smoothing**, and **Double-sided**.



### Weight Shade

**Weight Shade** provides visual feedback for editing **Weight Maps**.



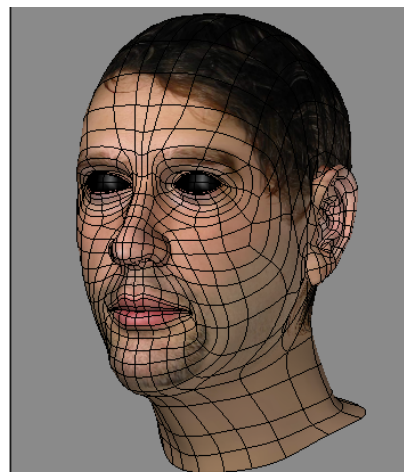
### Texture

**Texture** is similar to **Smooth Shade**, but it also shows images mapped to surfaces.



### Textured Wire

**Textured Wire** is similar to **Texture**, but it also shows wireframes on the surfaces.

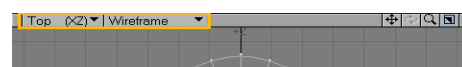


### Upright Rotation

The **Upright Rotation** pop-up menu lets you rotate a viewport clockwise by 0, 90, 180, and 270 degrees to accommodate imported objects from applications that may use other axes as «up», or for exporting to them, for that matter.

### Titlebar Shortcuts

You can change the **View Type** and **Rendering Style** of a viewport without going to the **Display Options Panel** by using the pop-up menus on the title bar. Clicking on the left-most pop-up allows you to choose which View Type you want to be in and will then display it. The pop-up menu just to its right lets you choose the **Rendering Style**.



Note that if you have **UV Texture** selected as the **View Type**, the **Rendering Style** pop-up menu will instead list all loaded images. Select one to load it into the backdrop of that viewport.



## Independent Options

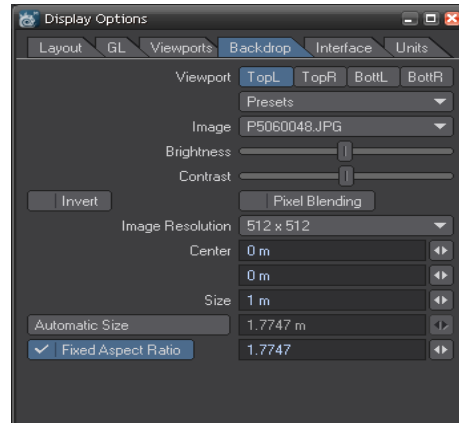
The Independent options let you make certain viewport characteristics independent from other viewports. Changes to the selected characteristic do not affect it in other viewports and vice versa. For example, if a viewport uses **Independent Zoom**, zooming in it does not affect the zoom of other viewports. You must have your pointer over the viewport to affect an independent viewport. **Independent Rotation** is only applicable to perspective views. Normally, it is enabled. If you have multiple perspective views and this option disabled, they will move in unison when you rotate a view.

## Independent Visibility

The lower portion of the panel contains the independent visibility options. You can individually select which visibility options are independent. To use them, you must also activate the **Independent Visibility** option. This option lets you quickly enable/disable this feature without losing the state of each setting.

## Backdrop Tab

On the **Backdrop Tab**, you can add full-color backdrop images that act as reference guides when you build objects. They let you model much like tracing a picture using tracing paper. To use backdrops, select the **Viewport** and the desired image from the **Image** pop-up menu.



**NOTE:** If you are using a UV Texture viewport, you can choose a backdrop using any loaded image from what is normally the Rendering Style pop-up menu on the viewport's title bar.

From the **Presets** pop-up menu, you can save the settings for the selected **Viewport** to a file by choosing **Save Current Backdrop**. You can load this file later using **Load Backdrop** for any selected **Viewport**. You can also save the settings for all viewports by choosing **Save All Backdrops**. When this file is loaded later — using **Load Backdrop** — the individual **Viewport** settings are restored.



**NOTE:** The backdrop images for the current default layout for each view configuration are cached as long as possible. Also, backdrops are not saved in config files or view presets.

Use the **Image** drop down menu to select from the loaded images or choose (**Load Image**) to bypass the **Image Editor** and load a new image into Modeler and automatically select it for the current viewport.

You can also adjust **Brightness** and **Contrast**. Use the sliders to adjust the settings. You can invert the colors by activating the **Invert** option. If you want to blend pixels for close-up work, activate the **Pixel Blend** option. **Image Resolution** determines the accuracy of the displayed image.



**HINT:** Colored background images sometimes make the wireframe hard to see (especially if the image has a bright white background). Setting the Contrast to its lowest value, and the Brightness to a little above the darkest value will produce a reasonable-looking backdrop that works well when Modeling.



## Aspect and Video images

The two **Center** and **Size** input fields are horizontal (top field) and vertical (lower field). Clicking **Automatic Size** will enter values that fit the defined image into an implied bounding box that surrounds visible geometry.

If you activate the **Fixed Aspect Ratio** option, you can enter a frame aspect ratio in the input field. Then, you only need to specify a horizontal size — the vertical size is computed automatically.

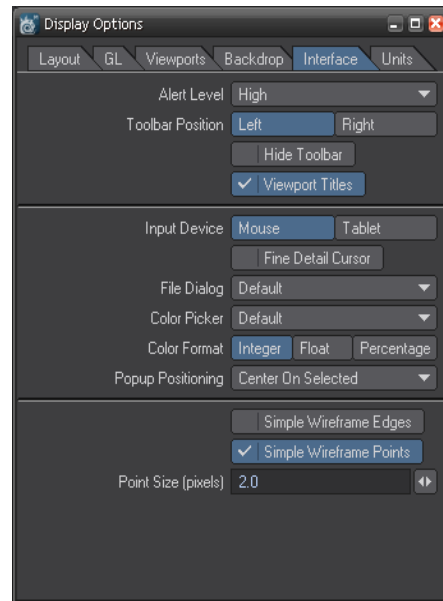
If your images originated from video, you need to account for the pixel aspect ratio, since video pixels are not square like computer pixels. For example, a D1 NTSC image is 720 x 486 pixels; however, the frame aspect ratio is not 1.48 (720/486) as you might expect. The width of a D1 NTSC pixel is 90 percent of the width, that is, a pixel aspect ratio of .9. As such, the true frame aspect ratio is 1.333 (720/486\*.9).



**NOTE:** The frame aspect ratio of a D1 PAL image is also 1.333 (720/576\*1.067).

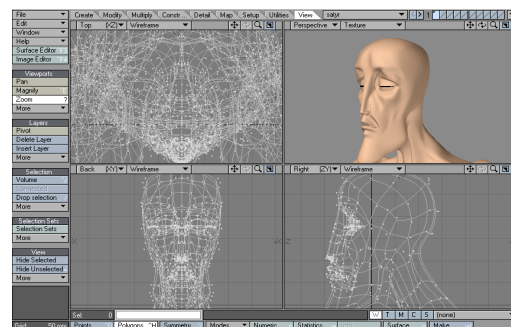
If you do not account for this difference, your models may appear slightly stretched when ultimately viewed on videotape. A way to fix this is to scale the object after the fact.

## Interface Tab

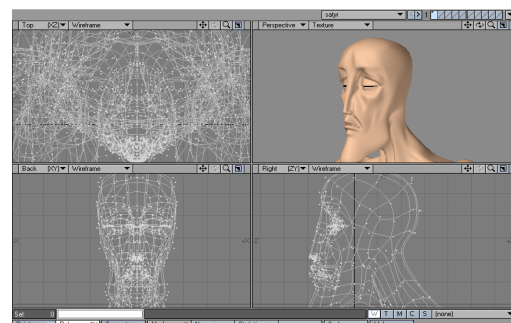


The **Alert Level** affects how error, warning and informational messages are displayed.

The **Toolbar Position** setting determines if the main toolbar appears on the **Left** or **Right** side. Activate **Hide Toolbar** to make it hidden and maximize your screen real estate. **Alt +F2** will hide (and more importantly, Unhide) the Toolbar as well.



Toolbar On

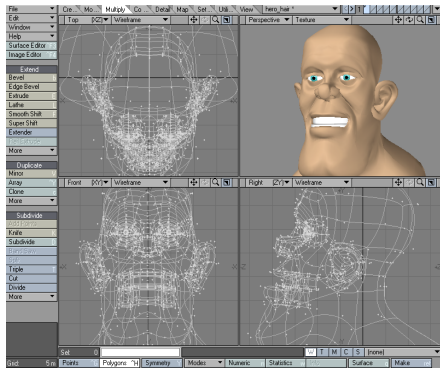


Toolbar Off

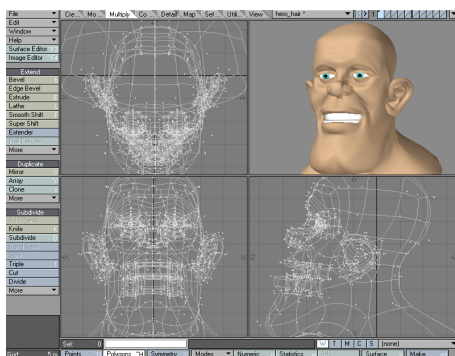




The **Viewport Titles** option turns the titlebars, which appear above each viewport, on or off.



#### Viewport Titles On

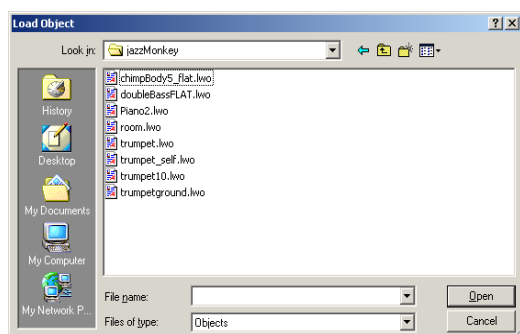


#### Viewport Titles Off

Use the **Input Device** buttons to select the type of input device you are using.

The **Fine Detail Cursor** option, when active, makes your mouse pointer use the main crosshair pointer at all times instead of changing when the various tools and selection functions are used.

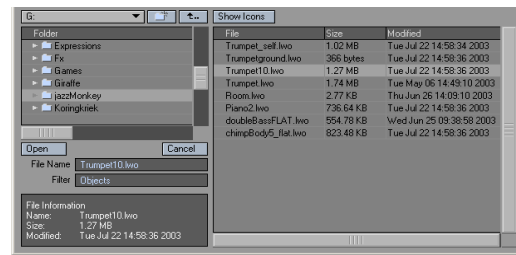
The **File Dialog** pop-up menu lets you use custom LightWave dialogs for file loading/saving. Selecting **Default** will use your standard system dialog.



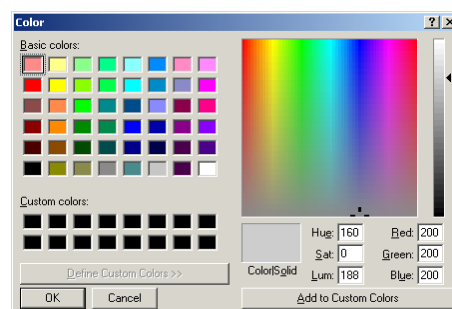
The custom File dialog provides many additional features not available with the standard system dialogs.

## The Visual Browser

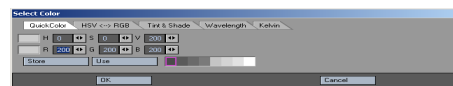
You can make the LightWave visual browser the default file dialog for Layout and/or Modeler by choosing **VBFileRequester** on the **File Dialog** pop-up menu. This appears on the **Interface Tab** of Modeler's **Display Options Panel (Edit > Display Options)**.



The **Color Picker** pop-up menu lets you use custom LightWave dialogs for picking colors.



The **LW\_ColrPikr** color picker is a color selector with more features, such as a wavelength and temperature-guided layout of color.

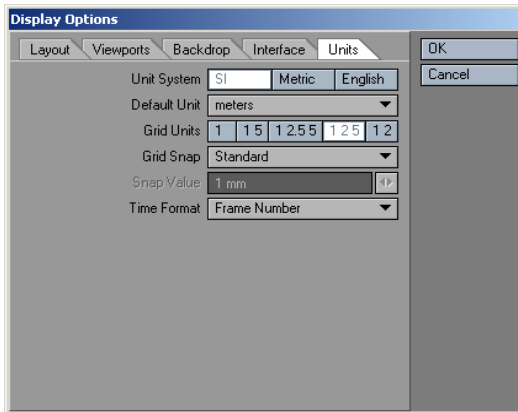


The **Color Format** setting determines the scale used where the color selector appears. **Integer** uses values 000 to 255, **Float** uses .00 to 1.00 and **Percentage** uses 0% to 100%.

The **Simple Wireframe Edges** option turns off polygon offsetting for the sketch-like display modes. This may fix display problems on some video cards. The **Simple Wireframe Points** setting uses OpenGL's own points for wireframe points and selected points at a user-specified size. These points can draw much faster than the standard points.



## Units Tab



The **Unit System** determines the units of measurement that are used and displayed by the Modeler screen.

**SI** — SI is the International System of Units (SI is the abbreviation of the French “Le Système International d’Unités”). Unit measurements in Modeler will now use a base system of meters. Grid sizes and distances can be measured in megameters, kilometers, meters, millimeters, micrometers, and nanometers.

**Metric** — The Metric System is the same as **SI** with the addition of centimeters.

**English** — The English System uses miles, feet, and inches.



**NOTE:** We recommend you use the SI or Metric systems to model objects, since a system using base 10 is often much easier to use for purposes such as applying textures and adjusting object movement.

If you input a value that uses a different unit of measurement than that of the default, LightWave will convert it on the fly. For example, you may be using meters, but typing in “5 ft” will convert to “1.524 m.”

### Default Unit

Use the **Default Unit** pop-up menu to set an assumed measure when none is given for distance values.



**NOTE:** If you use a metric Unit System, you should set the Default Unit to meters.

### Grid Units

**Grid Units** affects the zoom step amount but has no effect on the actual size of the object. You will likely find that **1 2.5 5** and **1 2 5** are your most commonly used **Grid Unit** settings.

**1** — The grid resizes in values that begin with 1, as in 10m, 1m, 100mm, 10mm, etc.

**1 5** — The grid resizes in values that begin with 1 or 5, as in 1m, 500mm, 100mm, 50mm, 10mm, etc.

**1 2.5 5** — The grid resizes in values that begin with 1, 25, or 5, as in 1m, 500mm, 250mm, 100mm, 50mm, etc.

**1 2 5** — The grid resizes in values that begin with 1, 2, or 5, as in 1m, 500mm, 200mm, 100mm, 50mm, etc.

**1 2** — The grid resizes in values that begin with 1 or 2, as in 1m, 200mm, 100mm, 20mm, 10mm, etc.

### Grid Snap

The **Grid Snap** setting forces point creation and item movement to be limited to a specific increment.

#### None

**None** deactivates grid snap, so that items move freely and are not constrained by any grid intersections.

#### Standard

**Standard** sets the grid snap to one-tenth of the current grid size.

#### Fine

**Fine** makes the grid snap as small as possible for the current zoom level. This will be typically two to five times smaller than using **Standard**.

#### Fixed

**Fixed** lets you specify the increment of movement in the **Snap Value** input field. The grid will resize itself to use the value you specify.

#### Time Format

The **Time Format** setting allows you to set the display format for time values, like the **Frame Slider Label** option on Layout’s **General Options Tab** of the **Preferences Panel**. If the **Hub** is active, **Layout** and **Modeler** settings will be kept in sync.

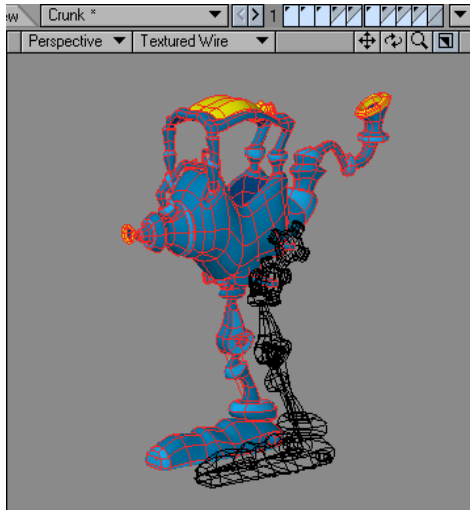




## Windows Menu

### Multi-Layer Object Standard

Every object can be a “MultiMesh,” that is, consist of an unlimited number of layers, similar to layers in many paint programs. The MultiMesh lets you work independently on specific parts of an object. During Modeling operations, you can set layers independently to be in the foreground or background, so you can work on a combination of layers as if they were in the same layer. Background layers can be used as a reference, but are often required when using certain Modeling tools such as boolean or cutting operations.



When you save an object, you save all layers with it. When you load an object, all of the individual layers remain intact; however, you may specify that certain layers be hidden when you use the object in Layout. You can thus save things like scratch objects — which you would not want rendered — in the same object file.



**NOTE:** When loaded into Layout, individual layers of a MultiMesh are treated as individual objects.

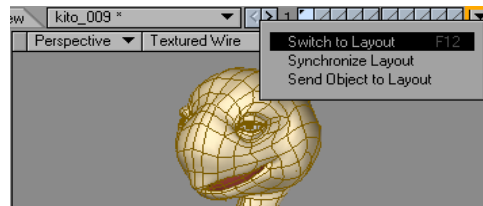
### Multi-document Environment

Like files in your word processor, multiple object files can be loaded simultaneously. The **Current Object** pop-up button to the left of the layer buttons switches from object to object. Object names listed with an asterisk (\*) have been changed and may need to be saved. Attempting to close the application triggers a request to save all changed objects. Ghosted objects are ones that have been loaded into Layout, but not into Modeler. Choosing a ghosted object from the pop-up menu will load it.



### Layout Communication

If the Hub is running, a small pop-up menu button appears in the top-right corner of the interface, just to the right of the layer buttons.



**Switch to Layout** switches to your Layout window if it is open or opens up Layout if you do not already have it open.

Changes to objects in Modeler are reflected automatically in Layout when you select the Layout interface. You can force this synchronisation by choosing **Synchronize Layout**.

**Send Object to Layout** sends the current object to Layout. Actually, it sends a pointer to the current object's file. As such, the object must be saved to your hard drive first.

If you load an object in Layout only, its name is ghosted in the **Current Object** pop-up menu. Selecting a ghosted item will load it into Modeler and initiate the syncing process.

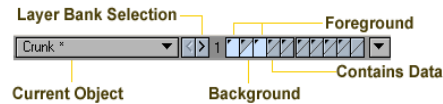
### Layer Navigation

You can use the layer buttons in the top-right corner of the main interface to work with layers in banks of ten. You can apply most Modeling functions across multiple layers simultaneously by placing them in the foreground. You do this by clicking the top part of a layer button; all other layers are unselected, whether they are in the foreground or background. You can add foreground layers by holding the Shift key as you click. You can unselect a foreground layer by holding the Shift key as you click on a



selected foreground layer button.

To place a layer in the background, click on the bottom part of a layer button. Selecting a background layer will unselect all other background layers. You can add background layers by holding the Shift key as you click. You can unselect a background layer by holding the Shift key as you click on a selected background layer button.



**NOTE:** You can quickly swap the state of foreground and background layers by pressing the apostrophe key (').



**NOTE:** If you notice a tool not working when multiple layers are selected, just select the one layer you need to work on and try the tool again.

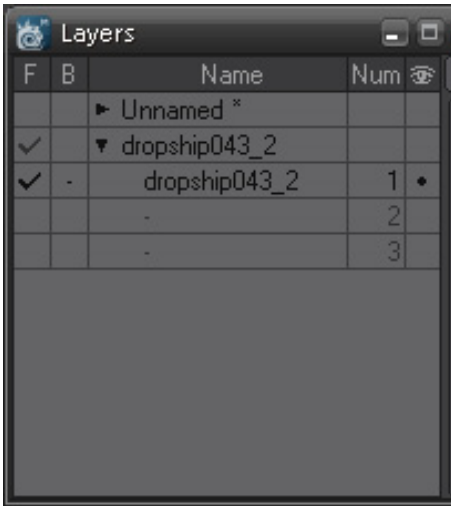
You can change which bank of layers — ten layers per bank — are controlled by these buttons. Just click the < or > buttons. Bank 1 controls layers 1 through 10, bank 2 controls layers 11 through 20, and so on. The **Bank Selection** buttons are mapped to your PgUp and PgDn keys by default. The highest bank number is 99, but you can access higher layers using the **Layer Browser Panel**.

Layer Browser Panel

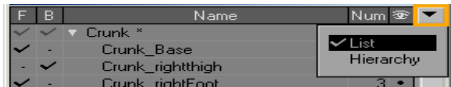
(default keyboard shortcut F7)

You can also use the **Layers Browser** window (**Windows > Layer Browser Open/Close**) to do the following:

- Access more than ten layers at a time
- Set foreground/background state, even between objects
- Navigate between objects
- Name layers
- Set layer visibility flag
- Parent object layers



Layers panel has a pop-up menu that lets you switch between **List** and **Hierarchy** views.

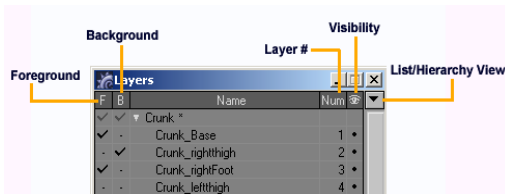


The **List** view lists the layers. In the **Hierarchy** view, the layers of an object are shown in their parental relationship, and that can be altered by dragging and dropping layers. Layers cannot be moved between objects and all children at the same level are shown in numeric order.

When in **Hierarchy** view, the layer list shows layers with nothing in them leftmost in the list, and lists two more than are currently being used. This makes it match list mode behaviour more closely.



The Alt+no. (1 through 0) keyboard shortcut will place a layer into the background. Note that if only one layer is active, it cannot be placed into the background.



All of the currently loaded objects will be listed. Click on the white triangle to expand or contract the layer listing for a particular object. Layers in the foreground will have a checkmark in the corresponding **F** column. Clicking in the **F** column for a different layer will place that layer by itself in the foreground. To add to the current selection, hold the Shift key as you click. The **B** column works similarly on background layers.

The Visibility flag can be toggled on/off for each layer by clicking in the column. A layer is invisible to Layout when the (Visibility flag) dot does not appear. You may want to use this on scratch layers and cutting objects used for booleans. This setting has no effect on Modeler.



**NOTE:** A pair of empty layers are always listed after the last occupied layer.

Double-click on a layer name (unnamed by default) or choose **View > Layers: Layer Settings** to bring up the dialog to set the layer name and parent.

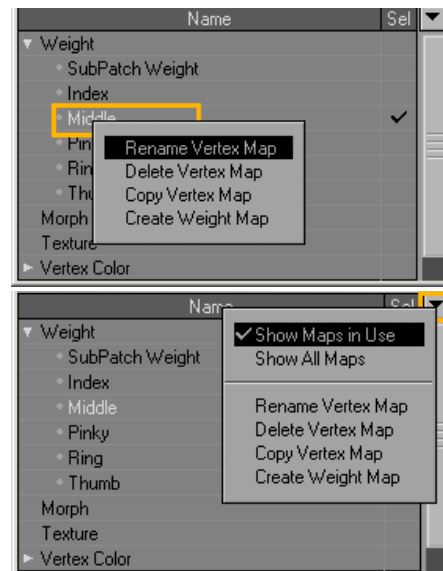
## Vertex Map Panel

(default keyboard shortcut **F8**)

You can display a non-modal window listing all **Vertex Maps** in the current object, grouped by type, by choosing **Windows > Vertex Maps Panel**.



Right-clicking on a map will display a pop-up menu where you can select various maintenance operations. This menu is also available for the **VMap** selected in the list using the pop-up menu button (down arrowhead) in the upper-right corner.



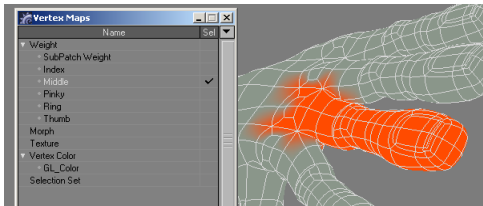
The pop-up menu also has options to show all **Vertex Maps** or to only show those that are in use in the current object, the default.



## VMap Selection

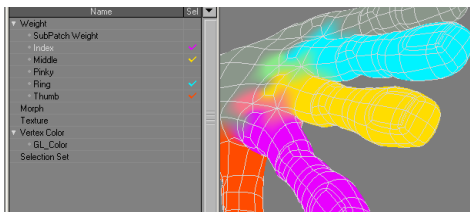
There are two levels of selection. First, one main **VMap**, regardless of type, can be selected by clicking on its name in the window (or selecting it in the pop-up menu in the lower-right corner of the Modeler interface). Its name will become highlighted. This is used by various commands that work on any type of **Vmap** (e.g., **Copy VMap** and **Delete VMap**).

Generally, one **VMap** from each type (e.g., **Weight** and **Morph**) may be selected by clicking in the **Sel** column. A checkmark appears for these. (Note that the main selected **VMap** is always the selected **VMap** for its type.) This selection state is used by tools that only deal with certain **VMaps**, like the **Airbrush** tool.



Changing the **VMap** selected in the pop-up menu on the (main) Modeler interface will change the main selected **VMap** and the **VMap** selected for its type. However, remember that not all **VMap** types (e.g., color **VMaps**) are accessible from this pop-up menu.

For display purposes, you can select up to four **Weight Maps**. They will be visible in **Weight Shade Mode** using different colors.



## Vertex Maps Shortcuts

**VMaps** can also be accessed using the buttons in the lower-right corner of Modeler. These are simply shortcuts to the same options that can be found in the **Vertex Maps Panel**. Choose a **Vmap Mode** to work in by clicking on one of the **Vmap Mode** buttons (**W**, **T**, **M**, **C**, **S**). Select which **Vmap** to edit from the **Vmap** drop down menu located next to the **Modes** buttons.



**W**- Weight, **T**- Texture, **M**- Morph, **C**- Color, **S**- Selection Set.

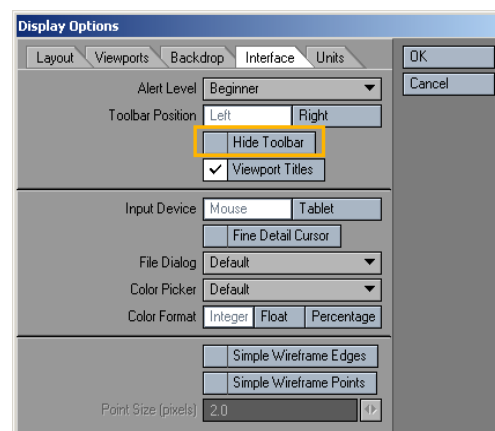
## Hide Floating Windows On/Off

Since your screen can often get cluttered with open panels, you can quickly hide/show these floating windows by pressing the **Alt F1**. This option can also be found under the **Windows** drop down menu.

## Hide Toolbar On/Off

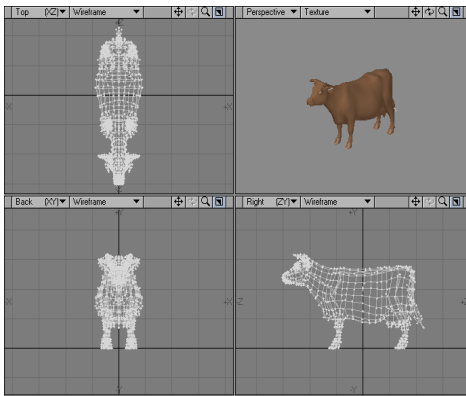
The toolbar sits at the side of the screen. The buttons presented will vary depending on which menu tab you select along the top. The **Top Tool Bar** buttons will appear no matter which Tab is selected. You can completely hide (or unhide) the toolbar by pressing **Alt F2** or by selecting **Hide Toolbar (Window> Hide Toolbar)**.

You can also hide the Toolbar by choosing **Edit> Display Options > Hide Toolbar On/Off**.



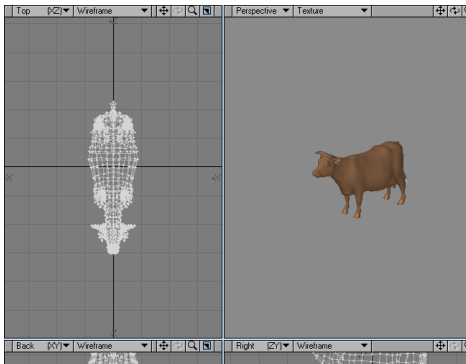
## Viewports

The largest part of the interface consists (by default) of four viewports. You can edit in any viewport and will be spending a lot of time in this area of the interface. The default four viewports starting in the top left and moving counter-clockwise are labelled Top, Back, Right, and Perspective.



The Top view is an overhead point of view looking down on the object. The Back view looks directly at the object from the rear. The Right view is positioned as if you stood to the object's right. Perspective is a forced perspective view that you can rotate by holding the Alt key and dragging your mouse on it.

You can change the relative sizes of the windows by dragging any border between the viewports.



As you grow familiar and gain experience with Modeler, you will find yourself switching back and forth among these viewports, changing their size, position, and function, and even changing the number, to get the best view from which to use a particular tool. You can adjust the screen to see as much or as little detail as you want, by using various options.



**HINT:** You can Maximise a viewport by moving the cursor over it and pressing the Numeric 0 key on the keyboard.

## Save View Presets

You also have the option in Modeler of saving viewport presets. You can place the cursor over a particular viewport and use the **Ctrl** key in combination with any of the number keys on the numeric keypad to map that viewport or the whole window to that hotkey. For example if you place your cursor over the perspective viewport and hit **Ctrl numpad 9** you can assign that viewport to that hotkey. Each time you hit the **numpad 9** key while holding the cursor over ANY viewport it will change to your preset. This is a great way to easily switch viewports or even whole viewport setups instantly for different workflows.

## Resetting Tools

Non-interactive areas (e.g., not a button) on the toolbar and bottom border (including the left-corner information display and grid size) act as a **Reset** button, much like you might use the **Esc** key on other applications. What happens when you click in a reset area depends on the current state of tools. For example, if you have points/polygons/edges selected and you are in the **Point/Polygon/Edge Selection Mode**, they become unselected.



All built-in tools (i.e., primitive and modifying) maintain their state, if they are used, unselected, and then used again. The default state of the tool can be recovered by dropping (clicking in a reset area or choosing **View/ Selection/ Drop Selection** when a tool is selected, but inactive.) If the tool is active, the first drop deactivates the tool, keeping the current settings. The second drop would then reset to the default settings.

You can also choose the **Make** tool. This is just like clicking the **Active Tool** button, making it inactive.



## Bottom Edge Menu

### Info Display

Just above the grid display in the lower left-hand corner of Modeler is the **Information Display**. Most of the time this gives you feedback on the position of your mouse. However, depending on what you are doing, it can also display a variety of other information.

Below are a few examples of what might be displayed with various tools:

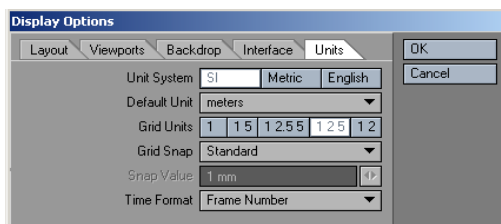
	Position	Angle	Scale	Offset
X:	-315 mm	-7.0	93.0 %	-145 mm
Y:	-120 mm		99.0 %	-25 mm
Z:				0 mm
	<b>Default</b>	<b>Rotate</b>	<b>Stretch</b>	<b>Magnet</b>

### Grid Info Field

A grid of squares — cleverly known as the grid — is visible in any of the orthogonal views, as well as the **Perspective** view. The grid serves as a visual reference when you move items around, but it will never render in a final image. The grid lines are darker every tenth square for visual reference. The Origin is located at the center of the grid.

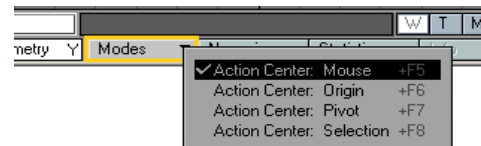
Position	
X:	2.95 m
Y:	
Z:	-3.05 m
Grid:	500 mm
	Points <b>G</b> Edges

You can find the current size of the grid squares in the **Grid Information** field at the very lower-left corner of the screen. Set the Unit System, Default Unit, and other options pertaining to the Grid in the **Display Options** Window.



## Modes: Action Center Control

The **Rotate** and **Stretch** tools (i.e., **Rotate**, **Size**, **Stretch**, **Twist**, **Taper1**, **Taper2**, **Vortex**, **Pole1**, and **Pole2**) can use different centers. This **Action Center** (Located at the bottom of Modeler's Interface) state is set on the **Mode** pop-up menu.



**Mouse** (**Shift F5**) means to use the mouse position.

**Origin** (**Shift F6**) means to use the very center of the universe: XYZ 0, 0, 0.

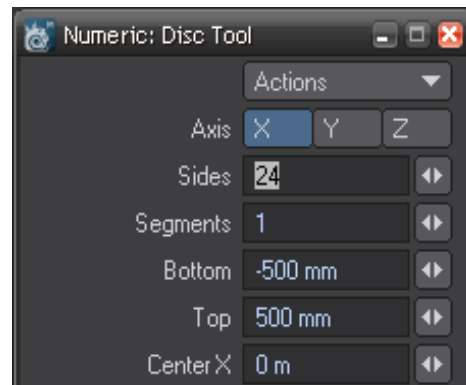
**Pivot** (**Shift F7**) uses the first active layer's pivot point.

**Selection** (**Shift F8**) uses the center of a bounding box around what is currently selected.

## Numeric Panel

(default keyboard shortcut **N**)

The **Numeric Panel** is a multi-functional non-modal window. Its contents and abilities change depending on what tool is selected. When you create primitives, it lets you refine graphically set values. When you use modifying tools, it lets you change various settings that affect how the tool performs. The **Numeric Panel** is located at the bottom of Modeler's Interface, or by just pressing the **N** key. It may be left open continuously.



Use the **Actions** pop-up menu to **Reset** the fields to their default settings or to **Activate** the tool. (You can also press the **N** key to activate the tool.) Generally, activating a tool turns on its interactive handles, if any, in the viewports. This can be for primitive shape, influence range, and so on. It will also activate the numeric fields, if they are ghosted.



**NOTE:** You can also reset to defaults by clicking a reset area when a tool is selected, but not activated.





## Make Command

(default keyboard shortcut **Return**)

The **Make** command (**Located at the bottom of Modelers Interface**) will commit to the changes made with the currently selected tool and will deactivate the current tool.

## Symmetry Mode

(default keyboard shortcut **Shift Y**)

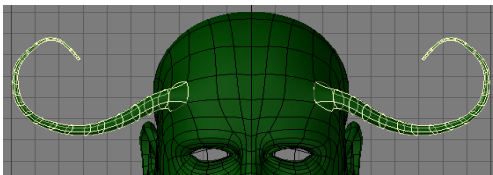
Working in **Symmetry Mode** (**Symmetry** button, located along bottom edge) works on selection, as well as on editing. This mode will allow you to work on one half of your object and **Symmetry** will perform all edits and selection on the opposite side of the geometry.



### Symmetrical Editing

Operations on the positive side of the X axis also inversely affect the negative side of the X axis. When this mode is active, your object is theoretically split in half at X=0.

Generally, you should perform all of your edits on the positive side of the X axis when using **Symmetry**. Using the negative side may lead to unpredictable results.



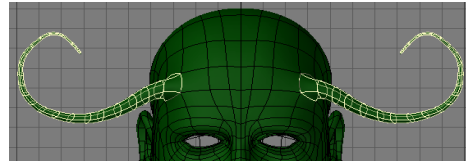
If you get unpredictable results, make sure the negative-side geometry is exactly the mirror of the positive-side geometry with X=0 as the center point. If the negative-side geometry is the slightest bit off, it will not be affected. Generally speaking, this mode was meant to be used where the geometry to the left and the one to the right of X=0 mirror each other. (However, all of the geometry does not need to be a mirror image, only the portion you want to work on symmetrically.)



**NOTE:** The **Fit Selected** command (Shift A) respects symmetry. If both sides of an object are selected and **Symmetry** is active, then only one-half of the selection is used to compute the fit.

### Symmetrical Selection

When you select polygons/points on the positive X axis and **Symmetry Mode** (**Symmetry** button) is active, polygons/points/edges on the negative X axis are also selected (or deselected).



Polygons/points/edges must be exactly opposite each other on the positive and negative sides of the X axis for this command to work properly.



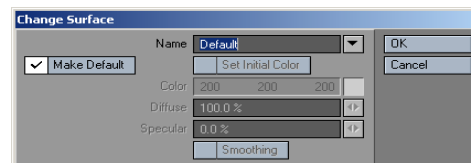
**HINT:** Symmetrical Modeling: Many of the Modeling tools will work in **Symmetry** mode. Most of the **Modify** menu tools

work fine. Others such as bevel or smooth shift also work. Some tools such as **Bandsaw** do not work in **Symmetry** mode. In such a case you may have to use the tool on one side then use the **Mirror** tool to make sure the changes are done on both sides of the X Axis.

### Surface: Change Surface

(default keyboard shortcut **Q**)

The **Change Surface** command (**Located at the Bottom of Modeler's Interface**) will bring up the Change Surface dialog to set the surface name and basic attributes for the selected polygons.



**Name** — This defines the surface name of the selected polygons. The Pop-Up menu will let you choose from existing surfaces.

**Make Default** — Selecting this option will assign this surface to any geometry created after the surface is made.

**Set Initial Color** — This option determines whether or not you can set basic attributes for your surface.

**Color** — Color is probably the most obvious surface parameter. It doesn't take much experience to know that if we want something to look like a banana, we need to make it yellow, right? However, since you are dealing with a 24-bit color palette and, thus, over 16 million colors, there are probably thousands of shades of yellow. Moreover, other settings, such as **Diffuse**, can have a dramatic effect on the final rendered color.





**Diffuse** — **Diffuse** (sometimes called diffusion) is the amount of light scattered by a surface. A high level scatters a lot of light, and therefore, the surface appears bright. A low level absorbs most of the light, and therefore, the surface appears dark and dull. Metal and dirt surfaces are good candidates for a low **Diffuse** level. Common values are 40% to 80%. Surfaces must have some diffusion for shadows cast on them to be visible.

**Specular** — **Specularity** is a kind of reflection or highlight that occurs on the surface of smooth or shiny objects. This highlight is really the reflection of the light source. High Specular levels are commonly used on glass spheres, chrome bumpers, and so on. How the surface reflects this highlight tells the observer if the surface is dull, smooth, shiny, hard, or even metallic.

**Smoothing** — **Smoothing** causes objects to appear to have smoothly rounded surfaces even though the object is composed of flat-faced polygons. To do this, LightWave uses a technique known as phong shading. If the edges of two smooth-shaded polygons share vertices (points), they appear as one continuous, curved surface. The shared edge between them is no longer visible.



**NOTE:** See Chapter 23: **Surface Editor** for more information.

### Vmap Shortcut

**VMaps** can be created using the buttons in the lower-right corner of Modeler. These are simply shortcuts to the same options that can be found in the **Vertex Maps Panel** (**Windows > Vertex Maps Panel**). Choose a **Vmap Mode** to work in by clicking on one of the **Vmap Mode** buttons (**W, T, M, C, S**). Select which **Vmap** to edit from the **Vmap** drop down menu located next to the **Modes** buttons.



**W**- Weight, **T**- Texture, **M**- Morph, **C**- Color, **S**- Selection Set.



**NOTE:** See **Vertex Maps Panel** for more detail.

## Help Menu

The **Help** menu (located in the upper left-hand side of the interface) gives you easy access to documentation, Newtek's website, and LightWave 3D's about box.

**Contents** — Choosing **Contents** will launch LightWave's digital help file. Use this to quickly search for information about a specific tool.

**Web Based Help** — Choosing **Web Based Help** will take you to NewTek's website where you can get the latest news about LightWave, LightWave Patches, tutorials, and more.

**About LightWave Modeler** — Choosing **About LightWave Modeler** will display Modeler's about box with information about the development team, as well as the Build number.





---

## Chapter 3: Create Tab

---



## Create Tab

The tools on the Create Tab allow you to make geometry. A wide range of tools is provided, including pre-made basic shapes known as *primitives*, drawing tools to create lines, curves and polygons, text, metaballs, and point creation tools that let you make points one at a time where you want them, or in groups at a time.

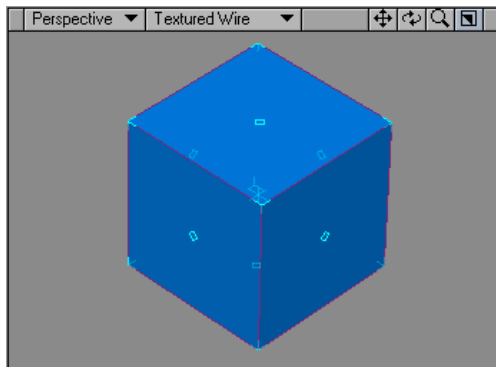
## Primitives

### Tools and the Mouse Buttons

As will be shown in the examples for the primitive tools below, the LMB allows you to activate a tool by clicking on it, interactively adjust the geometry, and finally make the geometry by clicking again on the tool to drop it and complete the operation. Most (interactive) tools that create geometry (**Box**, **Ball**, **Capsule**, etc.) use the **RMB** to complete one operation with the tool — making new geometry, if applicable — and then begin a new operation with it, essentially dropping the tool and then reactivating it in a single step.

### Creating in Perspective

You can create primitives in a perspective viewport in a similar way to how you would do it in a 2D viewport. You will see highlighted control points at the corners, sides, and center. You will also notice that you can see (fainter) hidden sides and control points. You can also drag any of the control points. Dragging a side control point will move the corresponding side along its perpendicular axis.

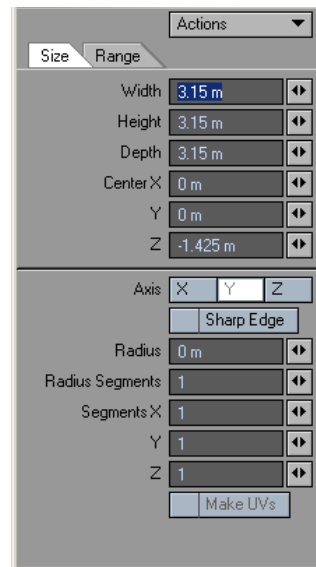


NOTE: Remember, you can rotate your view by holding the **Alt** key as you drag.

## Numeric Panels

For most tools there is also a **Numeric** panel available, which provides for precision when you need it, and which many find a quick and convenient way to control the various characteristics of the object they are creating. The contents of the numeric dialog will change as you select different tools. You may find it handy to just leave the numeric dialog open.

For example with the **Box** tool, the numeric panel offers two different ways to adjust the shape of your box, the **Size** and **Range** modes. Other options allow you to round off the edges of the box (**Radius** and **Axis**) or to make the edges sharp or smooth them (**Sharp Edge**).



## Using the cursor Keys

The cursor keys let you quickly adjust settings for most primitives. For the **Box** tool, for example, they work as follows:

Right/Left cursor

Up/Down cursor

Incr./decr. Horizontal Segments\*

Incr./decr. Vertical Segments\*

\*based on viewport beneath mouse pointer

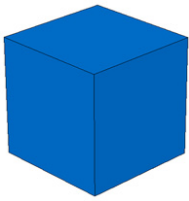
## Make UVs Option

The **Make UVs** option at the bottom of the various primitive **Numeric Panels** assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.



## Box Tool

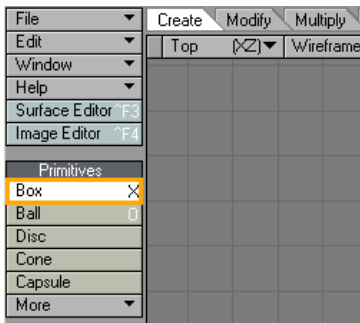
(default keyboard shortcut **Shift** **X**)



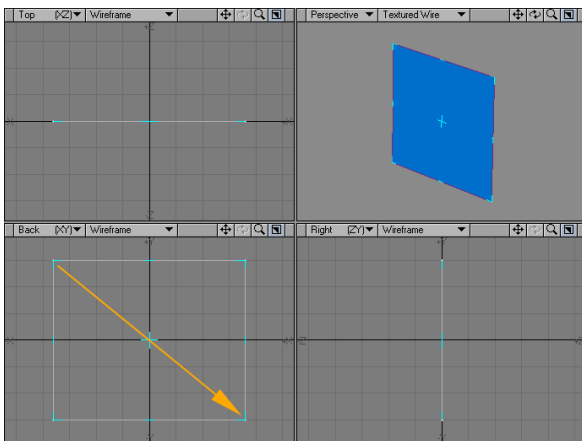
The **Box** tool is located in the **Create Menu Tab** under the **Primitives** group. To create a box simply drag out a two-dimensional shape in one of the viewports and then add depth to your object by expanding it in a different viewport. You'll probably use the **Box** tool most often, because most objects are box-like in general form.

### Creating a Box

**Step 1:** Select the **Box** button located in the **Create Menu Tab** under the **Primitives** group.

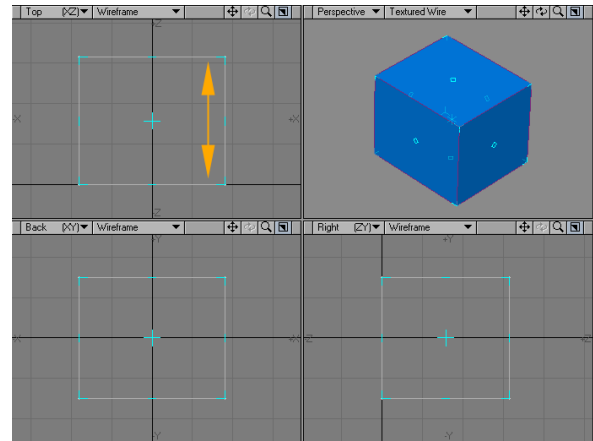


**Step 2:** Drag out the initial 2D shape in any viewport with your **LMB**. This activates the tool. Pressing the **Ctrl** key before you click and holding it while you drag will symmetrically constrain the shape. The dimensions of your outline are shown in the lower left corner.



**Step 3:** After you release the mouse button, you can adjust the shape by dragging any of the sides or corners. You can reposition it by dragging the center.

**Step 4:** Go to another viewport and drag to extend the shape into the third dimension. Release the mouse button.



**Step 5:** You can continue to adjust the box's size and position.

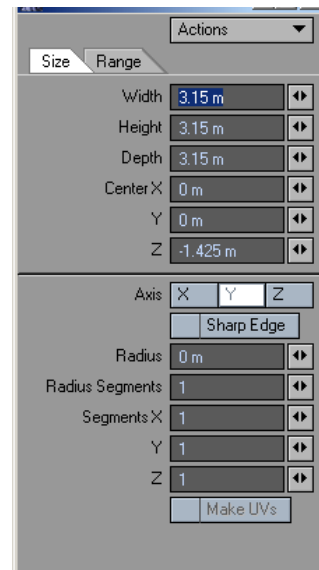


**NOTE:** If you find you need to start over from scratch, just click in an inactive toolbar area or use undo.

**Step 6:** When you are satisfied, deselect the **Box** tool by clicking on it or choosing **Make** to turn your outline into an object. (Alternatively, you can select another tool.)

### Numeric Panel Option: Box Tool

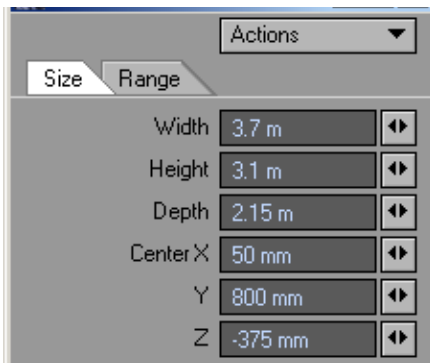
When you click on the Box tool to activate it, you can then bring up the numeric panel using the default keyboard shortcut **N**, or click on the Numeric button on the bottom toolbar of the Modeler window.



You have two different ways to adjust the shape of your box, the



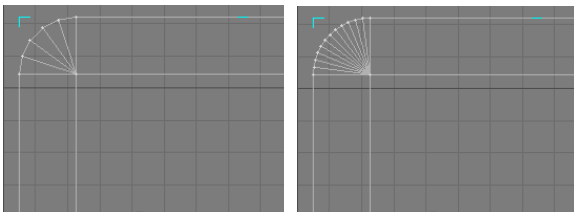
**Size** and **Range** modes. With the **Size Tab** selected, you can adjust the Width, Height, and Depth of your box dimensions. The **Center XYZ** settings set the position of the center of the box.



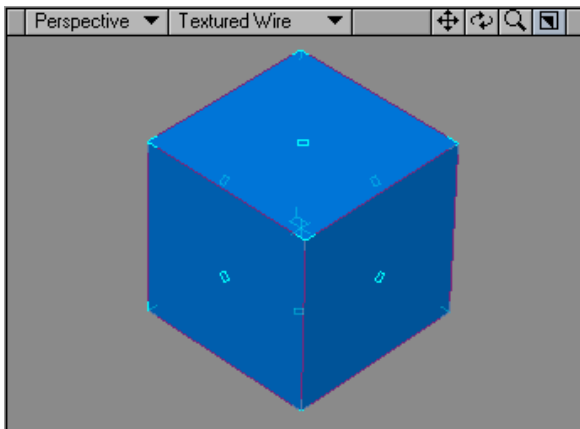
With the **Range Tab** selected, you can adjust the positions of two opposing corners of your box using the **Low XYZ** and **High XYZ** settings.



You can round off the edges of the box by setting the **Radius** to a value greater than 0. **Radius Segments** allows you to specify how many sides are used to generate the smooth corners (larger segments obviously generate smoother ones). The **Axis** sets the direction of the radial polygon pattern used to round the edges. Use the **Sharp Edge** option to smooth or unsmooth the edges.



Left: Radius 176mm, Radius Segments 5, Right: Radius 176mm, Radius Segments 12



Modeling a box in Perspective view.

## Using the cursor Keys

The cursor keys let you quickly adjust settings as follows:

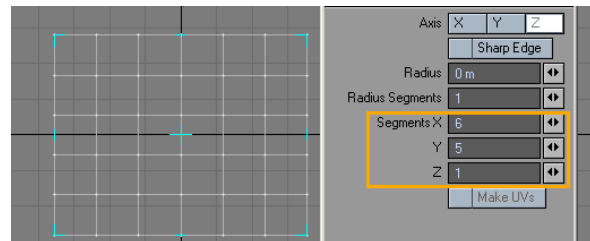
Right/Left cursor

Up/Down cursor

Incr./decr. Horizontal Segments\*

Incr./decr. Vertical Segments\*

\*based on viewport beneath mouse pointer

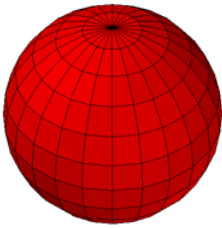






## Ball Tool

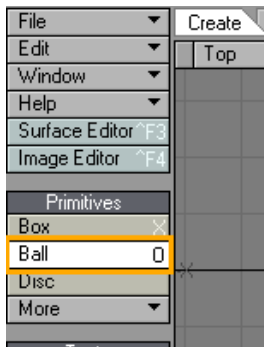
(default keyboard shortcut **Shift** **O**)



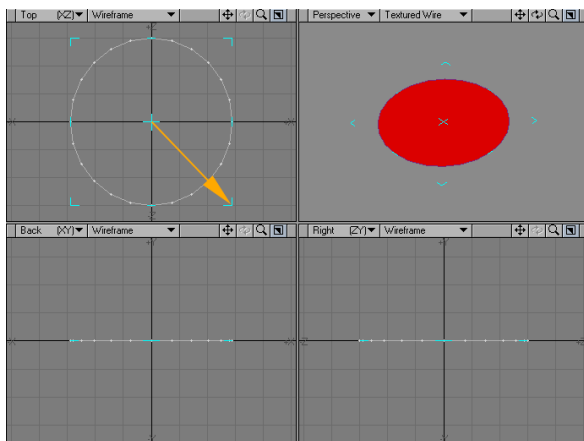
The **Ball** tool is located in the **Create Menu Tab** under the **Primitives** group. To create a ball simply drag out a two-dimensional shape in one of the viewports and then add depth to your object by expanding it in a different viewport. The **Ball** tool is a great starting place for planets, eyeballs, disco globes and more.

### Creating a Ball

**Step 1:** Select the **Ball** button located in the **Create Menu Tab** under the **Primitives** group.

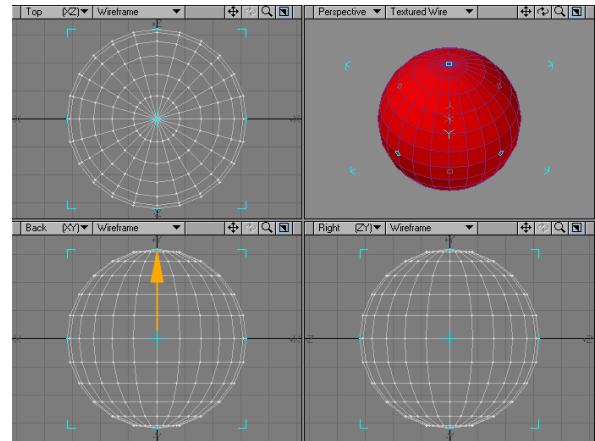


**Step 2:** Drag out the initial 2D shape in any viewport with your **LMB**. This activates the tool. Pressing the **Ctrl** key before you click and holding it while you drag will symmetrically constrain the shape. The dimensions of your outline are shown in the lower left corner.



**Step 3:** After you release the mouse button, you can adjust the shape by dragging any of the sides or corners. You can reposition it by dragging the center.

**Step 4:** Go to another viewport and drag to extend the shape into the third dimension. Release the mouse button.



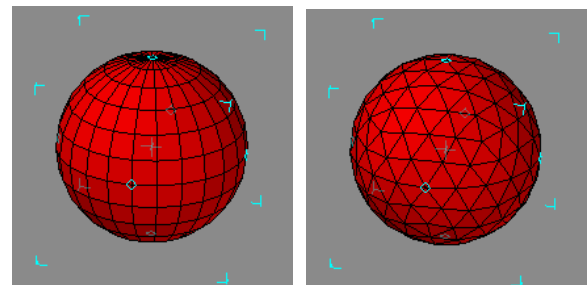
**Step 5:** You can continue to adjust the ball's size and position.



**NOTE:** If you find you need to start over from scratch, just click in an inactive toolbar area or use undo.

**Step 6:** When you are satisfied, deselect the **Ball** tool by clicking on it or choosing **Make** to turn your outline into an object. (Alternatively, you can select another tool.)

Type — The make-up of a Globe is determined by the number of Sides and Segments. Tessellation balls are set using a **Level** parameter; these balls are created entirely of triangles and are, therefore, smoother looking.



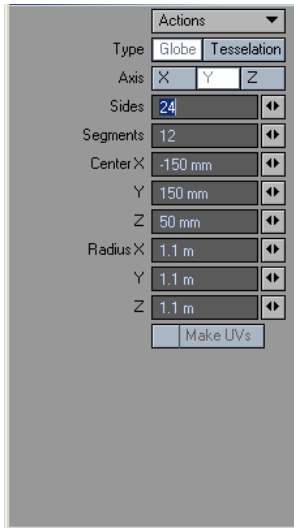
Left: Globe, Right: Tessellation



**HINT:** Use a Tessellation ball if the camera will be close to the object and the surface is smooth.



## Numeric Panel Option: Ball Tool



**Axis** — (Globe) determines the major axis for the ball. This is also set when you click to first create the ball using the axis perpendicular to the (orthogonal) viewport.

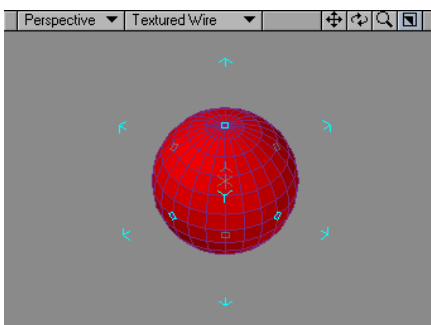
**Sides** — (Globe) determines how many segments should be used around the ball.

**Segments** — (Globe) sets how many vertical segments should be used. Three-sided polygons are always used at the top and bottom. Elsewhere, the polygons used are dependent on the **Polygons** setting on the **General Options Panel (Modeler > Options > General Options)**.

**(Tessellation)** sets the number of segments along the edges between the twelve polyhedral vertices. Higher settings will increase the complexity of the triangle matrix, taking longer to create and requiring more RAM.

**Center** — The XYZ coordinates of the center of the ball.

**Radius** — The radius of the ball along the X, Y, and Z axes.



Modeling a Ball in perspective view.

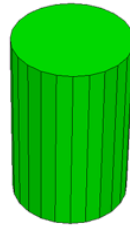
## Using the cursor Keys

The cursor keys let you quickly adjust settings as follows:

Right/Left cursor      Up/Down cursor

Incr./decr. Segments      Incr./decr. Sides

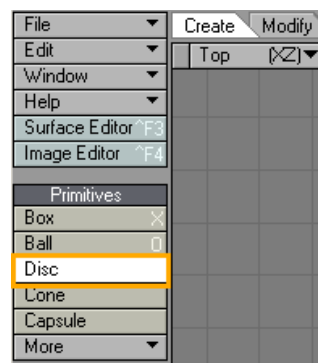
## Disc Tool



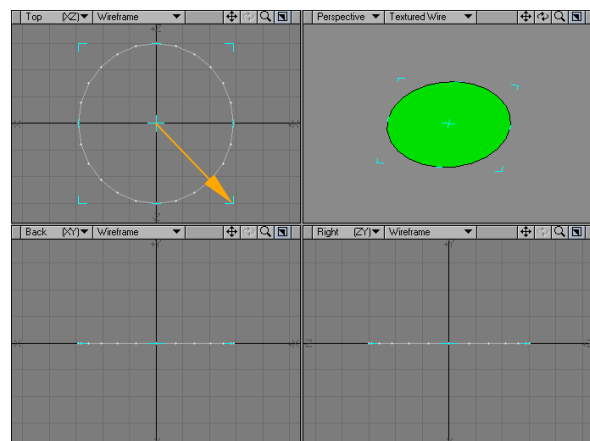
The **Disc** tool is located in the **Create Menu Tab** under the **Primitives** group. To create a disc, simply drag out a two-dimensional shape in one of the viewports and then add depth to your object by expanding it in a different viewport. The **Disc** tool is a great starting place for soda cans, pipes and more.

### Creating a Disc

**Step 1:** To activate the tool, select the **Disc** button located in the **Create Menu Tab** under the **Primitives** group.

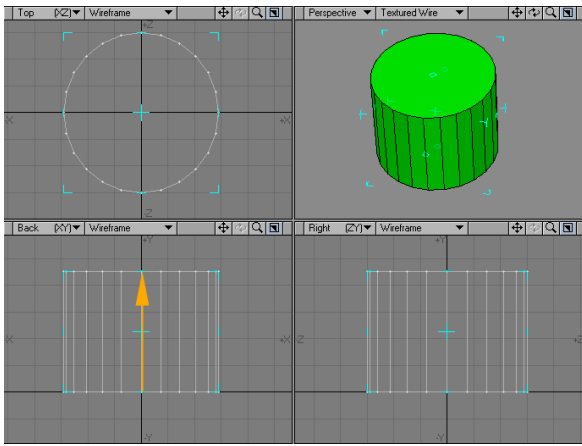


**Step 2:** Drag out the initial 2D shape in any viewport with your LMB. Pressing the **Ctrl** key before you click and holding it while you drag will symmetrically constrain the shape. The dimensions of your outline are shown in the lower left corner.



**Step 3:** After you release the mouse button, you can adjust the shape by dragging any of the sides or corners. You can reposition it by dragging the center. At this point it will look similar to the **Ball** tool until you continue to the next step.

**Step 4:** Go to another viewport and drag to extend the shape into the third dimension. Release the mouse button.



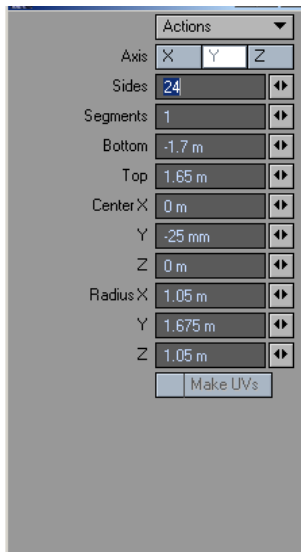
**Step 5:** You can continue to adjust the Disc's size and position.



**NOTE:** If you find you need to start over from scratch, just click in an inactive toolbar area or use undo.

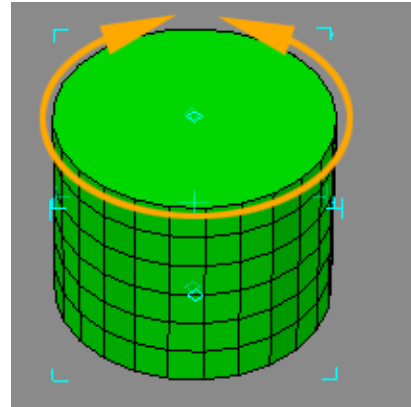
**Step 6:** When you are satisfied, deselect the **Disc** tool by clicking on it or choose **Make** to turn your outline into an object (alternatively, you can select another tool).

### Numeric Panel Option: Disc Tool

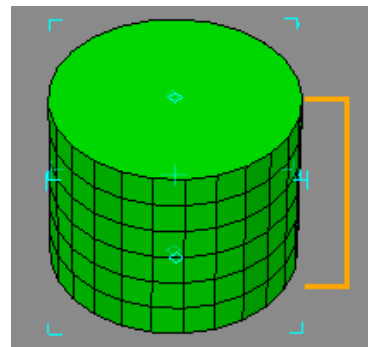


**Axis** — The primitive is aligned with this axis.

**Sides** — Sides determines how many segments should be used around the perimeter.



**Segments** — Segments sets how many vertical segments should be used.

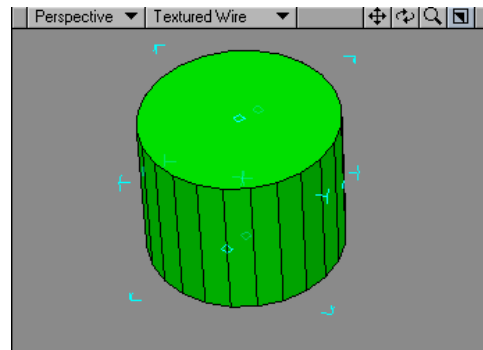


**Bottom** — Bottom is the starting point of the primitive along the selected Axis.

**Top** — Top is the ending point of the primitive along the selected Axis.

**Center** — Center refers to the coordinates for the center of the primitive.

**Radii** — Radii refers to the radius of the primitive along the axes.



Modeling a disc in perspective view.

### Using the cursor Keys

The cursor keys let you quickly adjust settings as follows:

Right/Left cursor

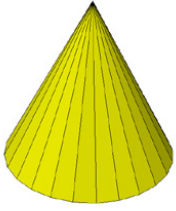
Up/Down cursor

Incr./decr. Segments

Incr./decr. Sides



## Cone Tool



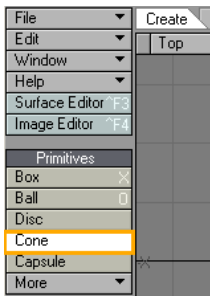
The **Cone** tool is located in the **Create Menu Tab** under the **Primitives** group. To create a cone simply drag out a two-dimensional disc shape in one of the viewports and then add depth to your object by expanding it in a different viewport. The **Cone** tool is a great starting place for an Indian teepee, snow cone, funnel and more.



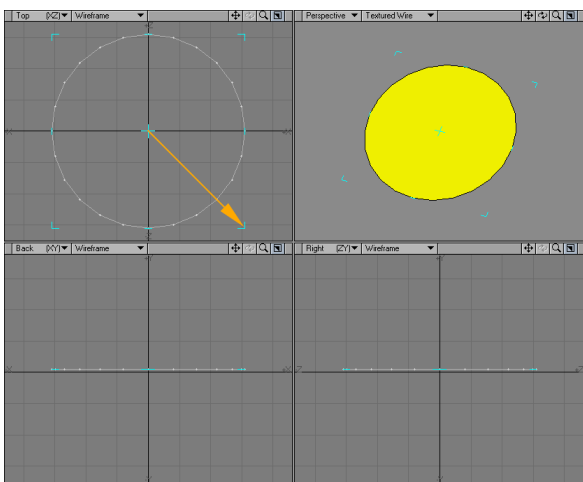
**HINT:** A cone with only four sides is a great way to quickly recreate the pyramids of Egypt!

### Creating a Cone

**Step 1:** Select the **Cone** button located in the **Create** menu tab under the **Primitives** group.

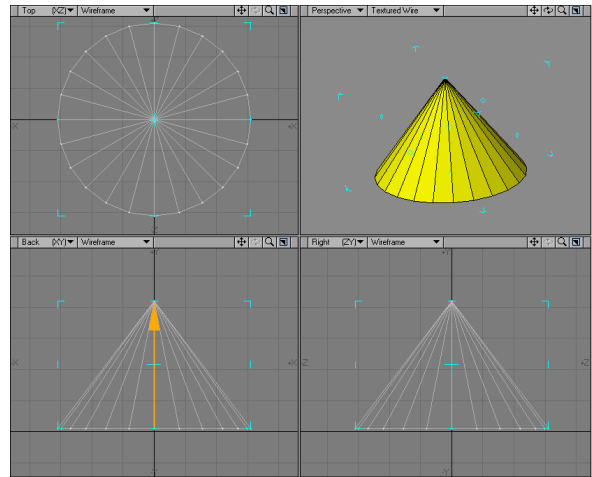


**Step 2:** Drag out the initial 2D shape in any viewport with your **LMB**. This activates the tool. Pressing the **Ctrl** key before you click and holding it while you drag will symmetrically constrain the shape. The dimensions of your outline are shown in the lower left corner.



**Step 3:** After you release the mouse button, you can adjust the shape by dragging any of the sides or corners. You can reposition it by dragging the center. At this point it will look similar to the **Ball** tool until you continue to the next step.

**Step 4:** Go to another viewport and drag to extend the shape into the third dimension. Release the mouse button.



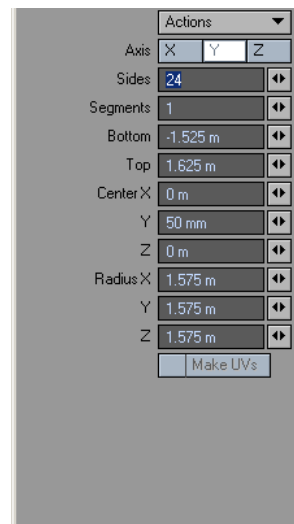
**Step 5:** You can continue to adjust the Cone's size and position.



**NOTE:** If you find you need to start over from scratch, just click in an inactive toolbar area or use undo.

**Step 6:** When you are satisfied, deselect the **Cone** tool by clicking on it or choosing **Make** to turn your outline into an object (alternatively, you can select another tool.)

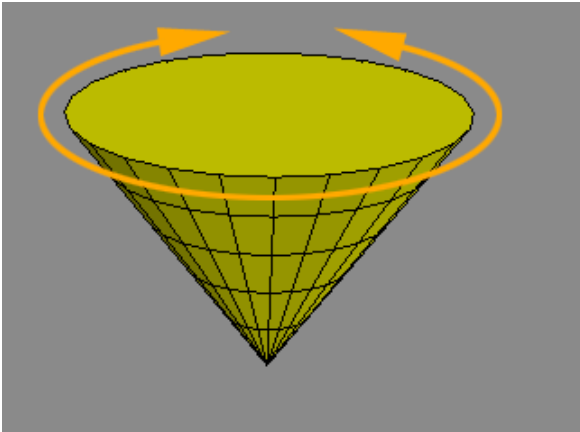
### Numeric Panel Option: Cone Tool



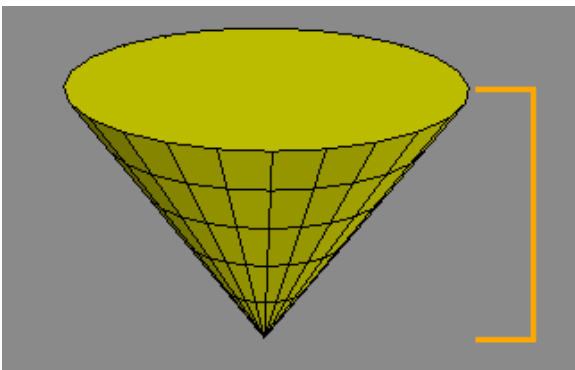


**Axis** — The primitive is aligned with this axis.

**Sides** — Determines how many segments should be used around the perimeter.



**Segments** — Sets how many vertical segments should be used.

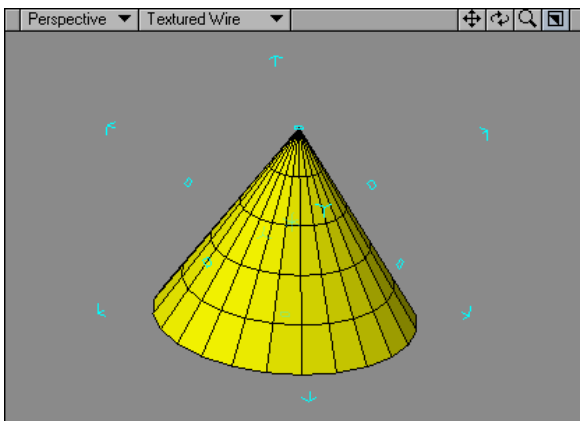


**Bottom** — Bottom is the starting point of the primitive along the selected Axis.

**Top** — Top is the ending point of the primitive along the selected Axis.

**Center** — Refers to the coordinates for the center of the primitive.

**Radius** — Refers to the radius of the primitive (wide end for a cone) along the axes.



Modeling a Cone in Perspective view.

### Using the cursor Keys

The cursor keys let you quickly adjust settings as follows:

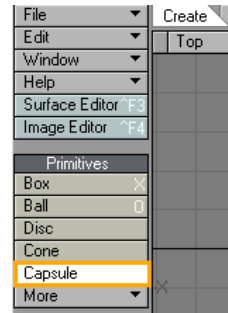
Right/Left cursor      Up/Down cursor

Incr./decr. Segments      Incr./decr. Sides

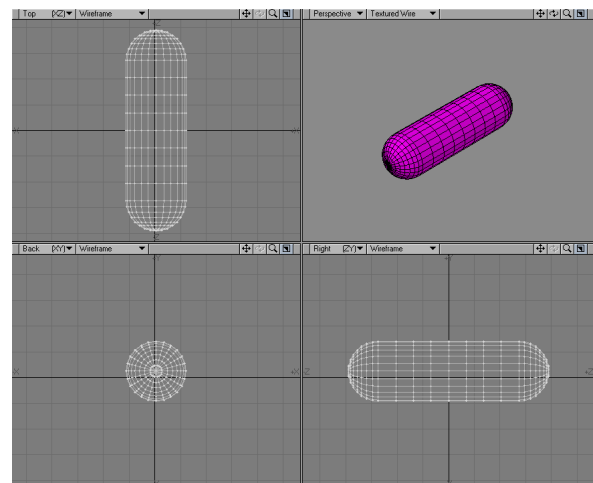
## Capsule Tool

Another primitive tool is the **Capsule** tool (**Create > Primitives: Capsule**). This is used to create a cylindrical object with rounded, closed ends.

### Creating a Capsule

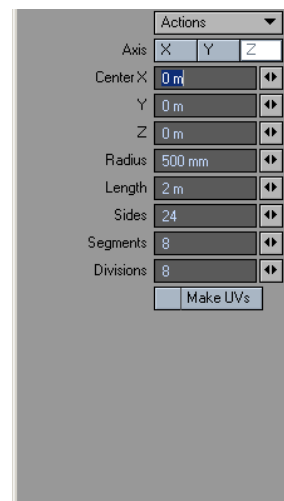


Your initial click defines the perpendicular axis of the object. You can drag the center position handle to move the shape, and you can drag the outer bounding box edges to resize it.



### Numeric Panel Option: Capsule Tool

The numeric settings are similar to those used for the standard primitive tools.

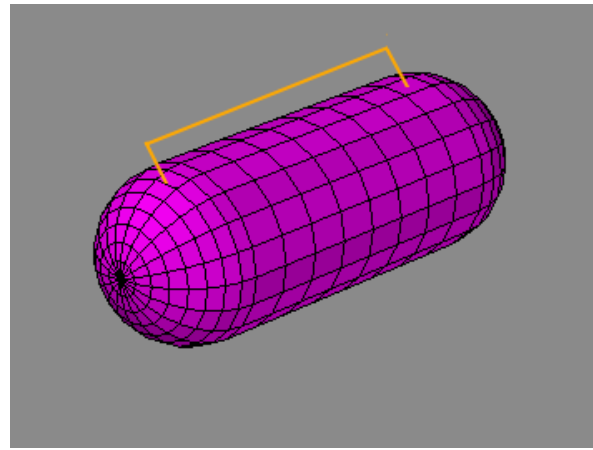
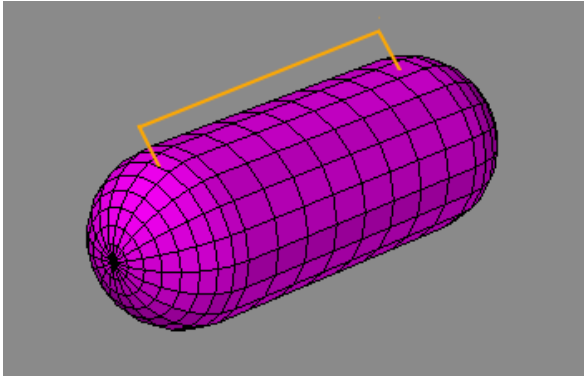




Axis — The primitive is aligned with this axis.

Center — Refers to the coordinates for the center of the primitive.

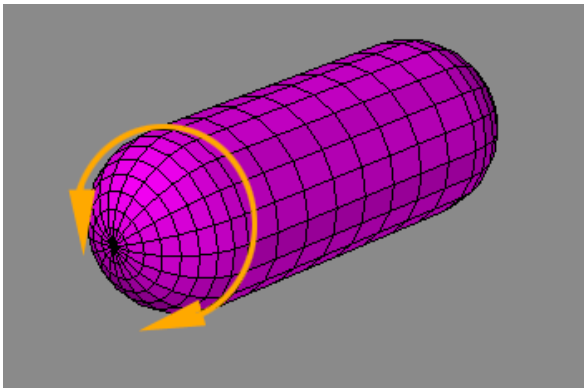
Length — Refers to how long the center of the object will be. Note:  
The ends are unaffected  
by this setting.



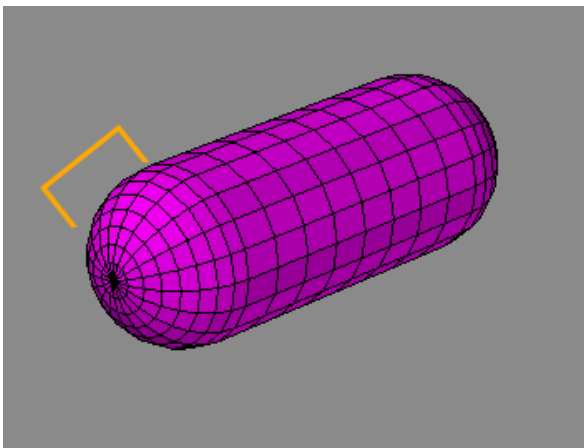
NOTE: Most tools allow numeric input. Thus, the contents of the numeric dialog will change as you select different tools. You may find it handy to just leave the numeric dialog open.

Radius — Refers to the radius of the primitive along the axes.

Sides — Determines how many segments should be used around the perimeter.



Segments — Sets how many vertical segments should be used on the ends.



Divisions — Sets how many segments should be used on the center of the object.





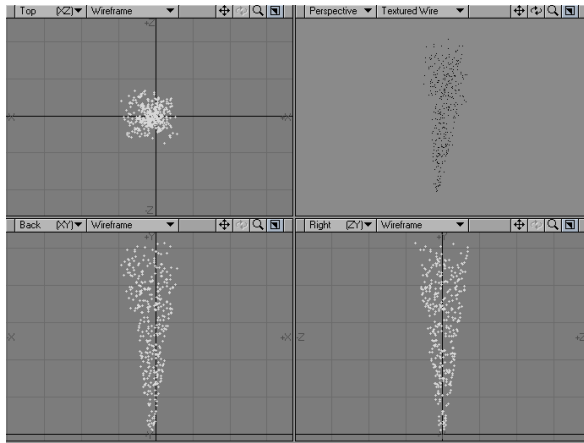
## Bubbles

The **Bubbles** tool (**Create > Primitives > More > Bubbles**) provides a quick and easy way to create bubbles, rocks, stylised clouds, and a variety of other interesting objects. This tool replaces each selected point in your foreground layer with a sphere.

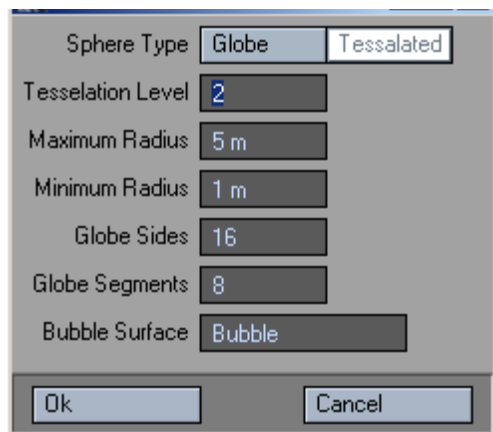


NOTE: If no points are directly selected, the **Bubbles** tool will use every point in the foreground layer. Remember, more points mean longer calculation times.

To begin, create an assortment of points in the foreground layer. The **Spray Points** tool is particularly suited for this.



Once you are satisfied with the number and distribution of your points, select the **Bubbles** tool from the **Create Tab** in the **Primitives** section under **More**.



The options for creating the spheres are similar to those found in the **Ball** tool.

Under **Sphere Type**, **Globes** are composed primarily of four-point polys (also known as quads) and can appear angular if they are created with a low number of sides or segments. **Tessellated** spheres are composed entirely of triangles and, using a comparable number of polys, will often yield smoother results.

If you choose **Tessellated**, the resolution of your bubbles will be determined by the **Tessellation Level**. **Tessellation Level** is similar to the **Ball** tool's "Segments" setting, however, it is important to note that these settings are not identical. A

**Tessellation Level** of 2 (which is the default) will produce significantly more polys than a **Segment** setting of 2 with the **Ball** tool.

The default **Tessellation Level** should work for most bubbles. However, higher numbers (i.e. 4 or 6) can be used for extremely detailed objects. Though a setting of 6 will produce unwieldy individual bubbles with more than 100,000 polygons each.

For **Globes**, the resolution of your bubbles will be determined by the number of sides and segments, which correspond to those in the **Ball** tool.



NOTE: Tessellation Level has no effect when using **Globe** objects and **Sides** and **Segments** have no effect when using **Tessellated** objects.

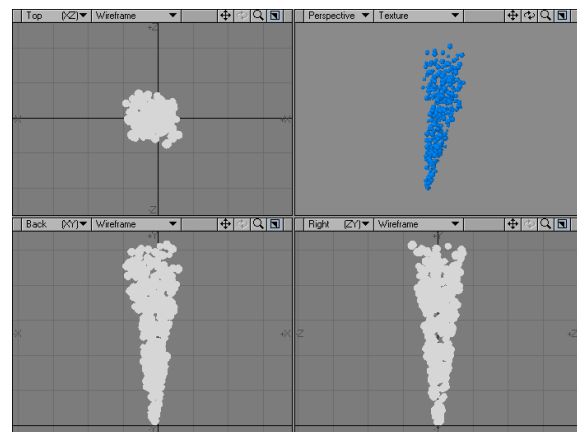
The **Bubbles** tool uses its own internal routines for varying the size of each bubble it creates, however you can determine the largest and smallest bubble size by adjusting the **Maximum** and **Minimum Radius** settings.

For the best results, use the **Fit All** command (a) to show every one of your source points.

Then reference your Grid size before entering the **Radius** settings. If you enter a **Maximum** or **Minimum** radius that exceeds your Grid size, your bubbles will frequently overlap and produce undesirable results.

You name the surface for the bubble objects in the **Bubble Surface** field. The default is **Bubble**, although you can choose any name.

Once you are satisfied with your settings, click on the **OK** button. The position of each point will be recorded, after which the points will be deleted and bubble objects of varying size will be created in their place.



NOTE: The **Bubbles** tool requires one undo for every bubble it generates, which means that running this tool on a large number of points may be impossible to fully undo. If you want to test different **Sizes** and **Sphere** settings, you should copy and paste your points to a new layer before running the **Bubbles** tool.

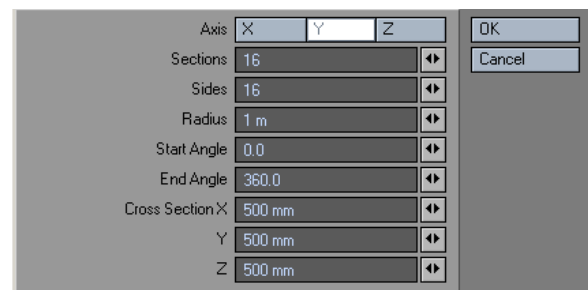
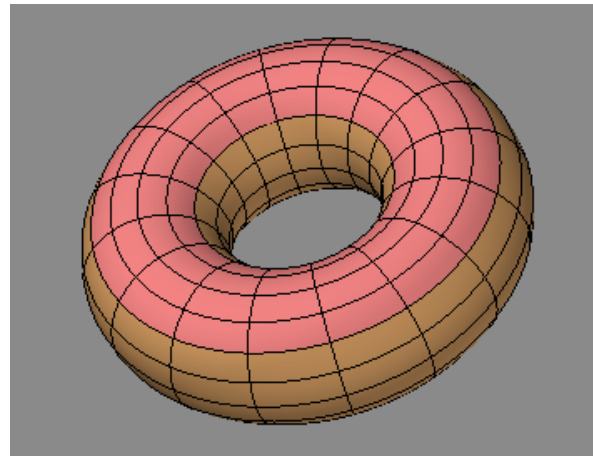


HINT: The Bubbles tool uses the points in the foreground layer, however polygons are ignored. As such, it is possible to achieve interesting results by running the Bubbles tool on one of your existing objects.



## Toroid Function

Choose **Create > Primitives: Toroid** to generate a torus or a section thereof.



Axis — Axis is the axis perpendicular to the doughnut hole.

Sections — Specifies the number of segments used around the doughnut.

Sides — Specifies the number of sides each segment should use.

Radius — Radius is the distance from the center of the doughnut to the center of a cross section.

Start Angle — Start Angle is the starting angle around the selected Axis.

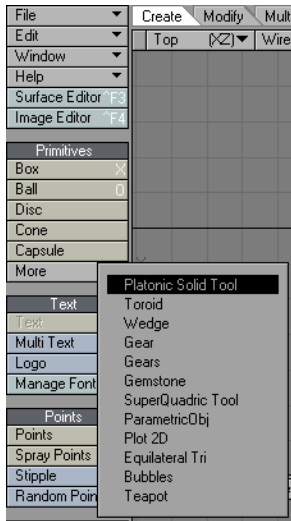
End Angle — End Angle is the ending angle around the selected Axis. (360 for a complete glazed doughnut).

Cross Section — Cross Section is the XYZ size of a cross section. Note that the value corresponding to the selected Axis is not considered.

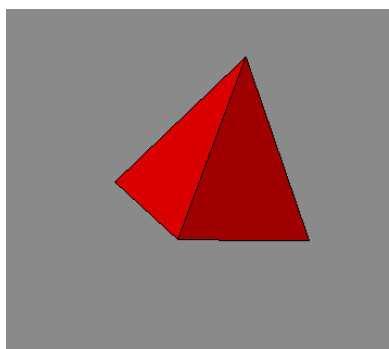
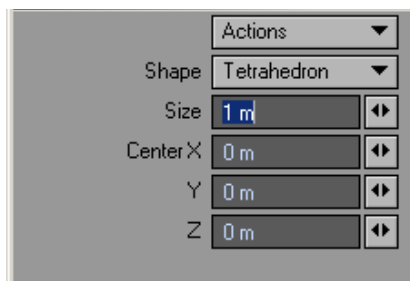


## Platonic Solid Tool

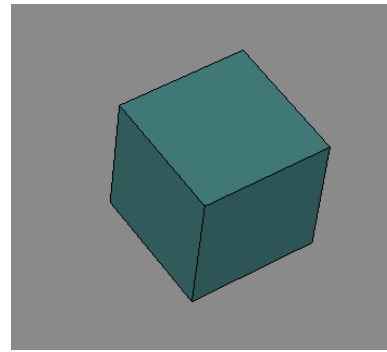
The **Platonic Solid** tool is located in the **Create Menu Tab** under the **Primitives** group. This tool is very easy to use and can really speed up the Modeling process. A platonic solid, as denoted by Euclid in the last proposition of his seminal mathematics textbook *The Elements*, is a polyhedron whose faces are all congruent regular polygons, and where the same number of faces meet at every vertex. The best known example is a cube (or hexahedron) whose faces are six congruent squares. Platonic solids are often used as a dice in role playing games, so if you want to make a D4, D6, D8, D12 and D20 easily, this is the tool for you!



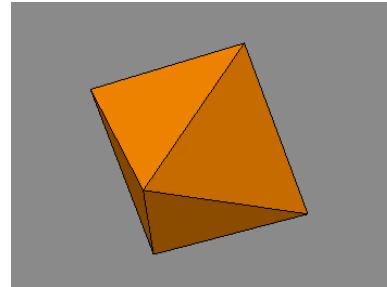
The **Shape** pop-up menu on the **Numeric Panel** controls the shape of the object. It gives you seven primitive shapes to select from.



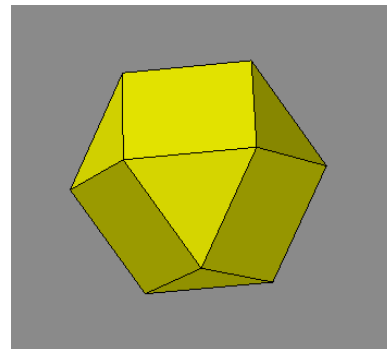
**Tetrahedron** — the simplest of the platonic solids made with just four triangles.



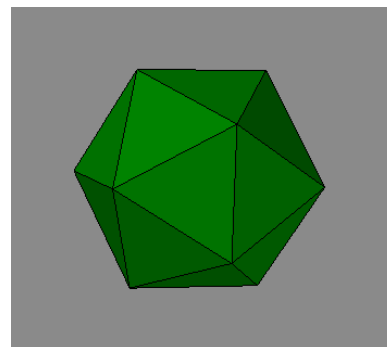
**Cube** — the best known of the platonic solids, otherwise known as a hexahedron.



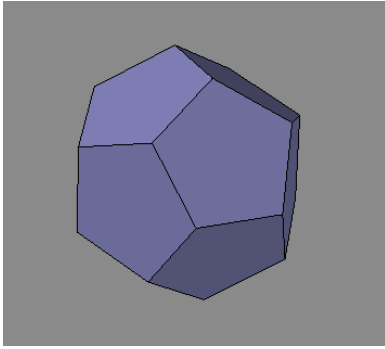
**Octahedron** — an eight-sided object.



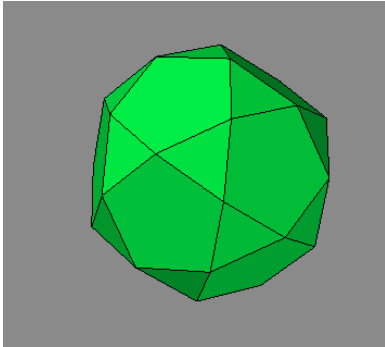
**Cubeoctahedron** — Although this panel is named the **Platonic Solid** tool, a Cubeoctahedron is not actually a platonic solid.



**Icosahedron** a 20-sided object made with triangles.



**Dodecahedron** — a 12-sided object made from pentagons.

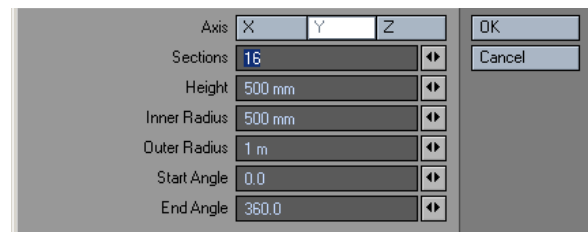
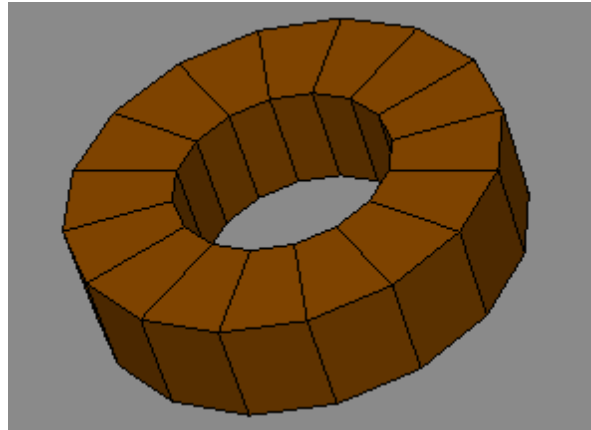


**Icosidodecahedron** — This one isn't a platonic solid either!

You can drag the center position handle to move the shape, and you can drag the outer bounding box edges to resize it. You can also use the numeric window to accomplish these options.

## Wedge Tool

The **Wedge** function (**Create > Primitives > More > Wedge**) will generate an annulus (like a flat donut).



**Axis** — Axis is the axis perpendicular to the doughnut hole.

**Sections** — Refers to the number of radial sections to use for the object.

**Height** — Specifies the width of the object.

**Inner Radius** — Specifies the radius of the inner hole.

**Outer Radius** — Specifies the radius to the outside perimeter.

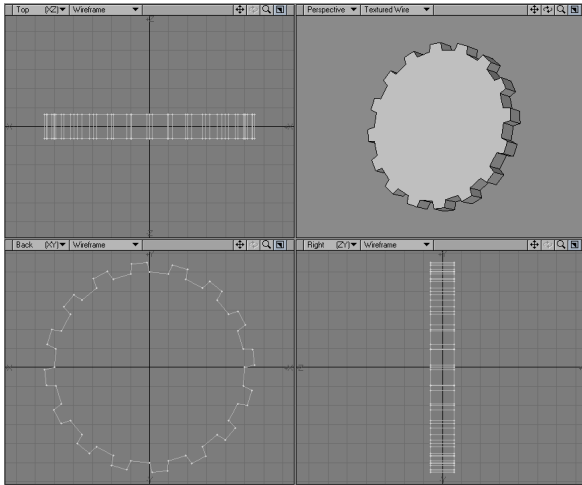
**Start Angle** — Start Angle is the starting angle around the selected Axis.

**End Angle** — End Angle is the ending angle around the selected Axis (360 for a complete doughnut).



## Gear Tool

The **Gear** function (**Create > Primitives: Gear**) creates a gear-shaped object. This tool is very similar to the **Gears** function (discussed next) but has fewer options.



**Teeth** — Refers to the number of teeth in the gear.

**Taper %** — Sets the “pointiness” of the teeth. A higher setting creates sharper teeth.

**Inner Radius** — Refers to the distance from the center of the gear to the bottom of the teeth.

**Outer Radius** — Outer Radius is the distance to the outside edge of the teeth.

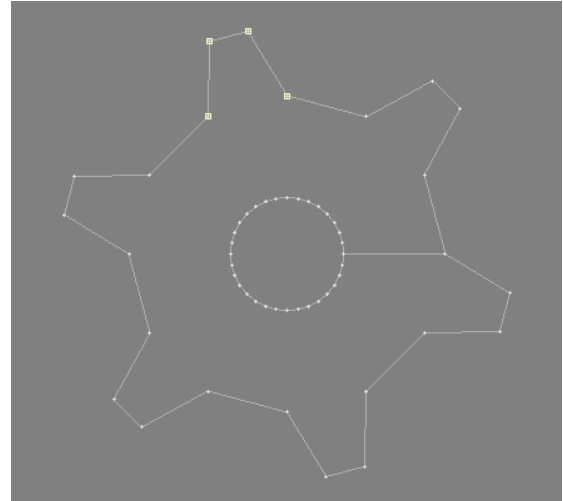
**Thickness** — Thickness is the width of the gear.

**Axis** — Axis is the perpendicular axis of the gear.

## Gears

True to its name, the **Gears** tool (**Create > Primitives > More**) quickly creates gear-shaped objects. Before describing the controls, let’s first take a look at what’s happening behind the scenes.

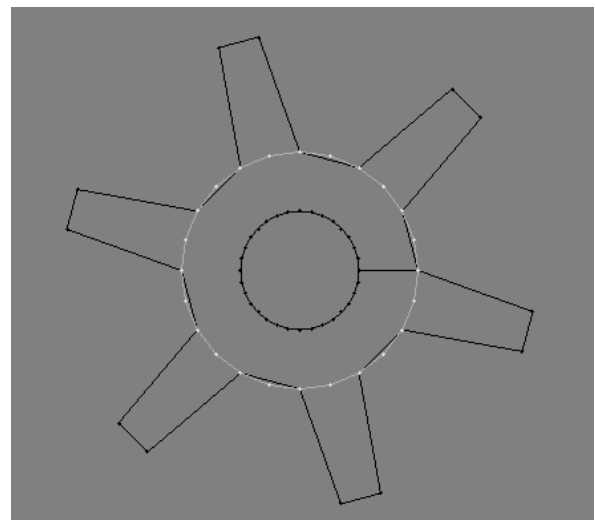
Gear objects are composed of a disc shaped object with protrusions called “teeth.” Each tooth is made up of four points.



Two points comprise the base of each tooth, and two the top. Therefore, a gear with two teeth would be made up of 8 points, a gear with four teeth would consist of 16 points, and so on.

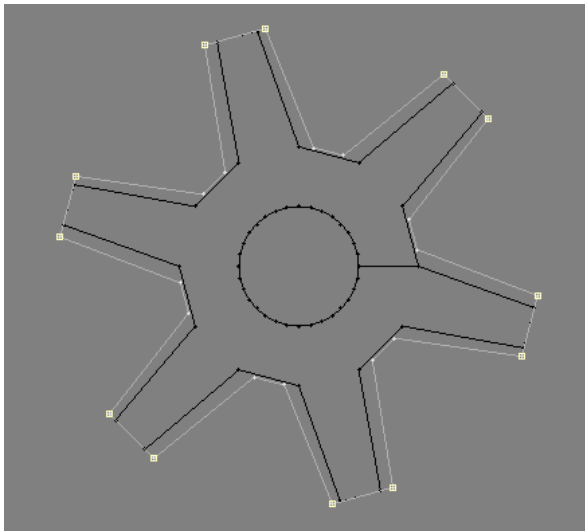
When you run the **Gears** tool, it first looks at the number of teeth you requested. It then multiplies the number of teeth by four (the number of points required for each tooth) and creates a disc using this total number of points.

The diameter of the disc is determined by you, and is called the “Inner Radius.”

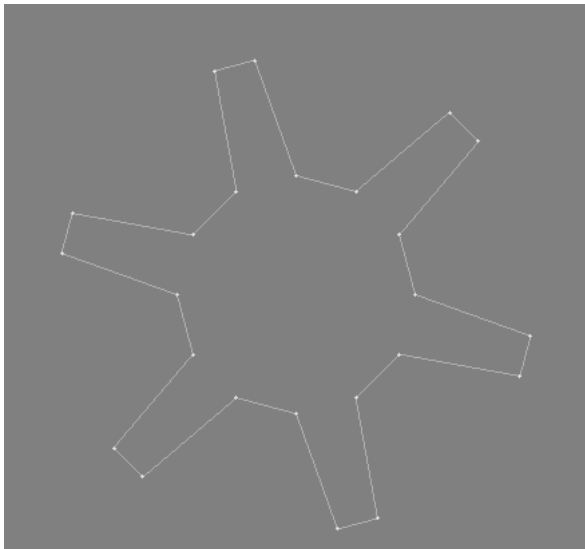


Here you can see that a disc with 24 points has been created. At four points per tooth, this will yield a gear with six teeth. An existing gear with six teeth can be seen in the background.

Once the disc has been created, pairs of points are selected and moved outward. The position these points are moved towards is determined by you, and is called the “Outer Radius.”



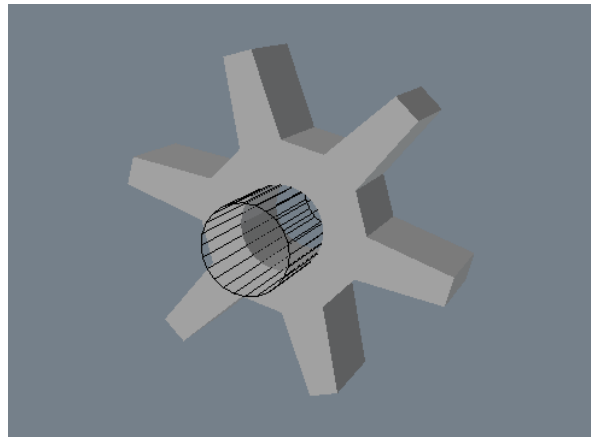
Finally, the four points comprising each tooth are tapered by approximately 65%.



Once the basic gear shape has been created, it is extruded to a **Thickness** specified by you.

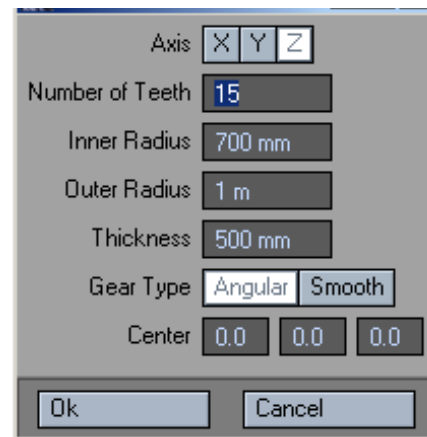


To complete the gear, a hole is punched out of the center. This is accomplished by the creation of a disc whose diameter is exactly  $\frac{1}{2}$  the size of the **Inner Radius**. It is extruded, placed in the background, and used as a cutting template for a Boolean Subtract operation.



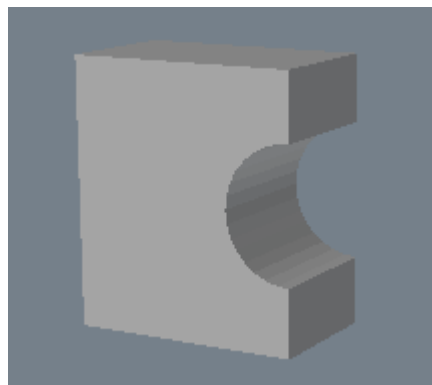
This entire process happens internally, and is transparent to you.

Now that we've looked at how gears are created, let's take a look at the controls offered by this tool.



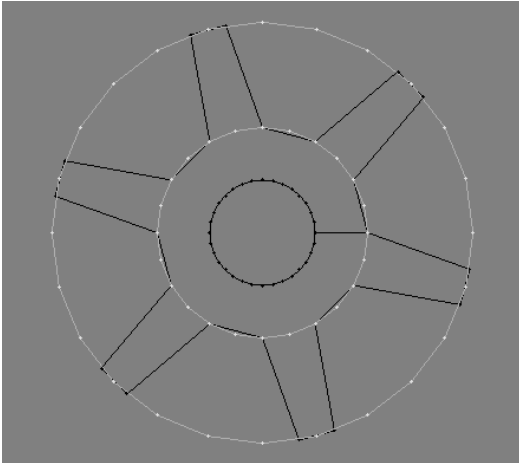
The first row of buttons determines the axis the base gear object will face once it has been created. It also determines the axis along which the gear object will be extruded using the **Thickness** setting (mentioned below).

The **Number of Teeth** enables you to set the number of teeth your gear will have. Note that at least three teeth are needed for a usable gear, and that entering 1 for the number of teeth will not result in a gear at all, as shown below.

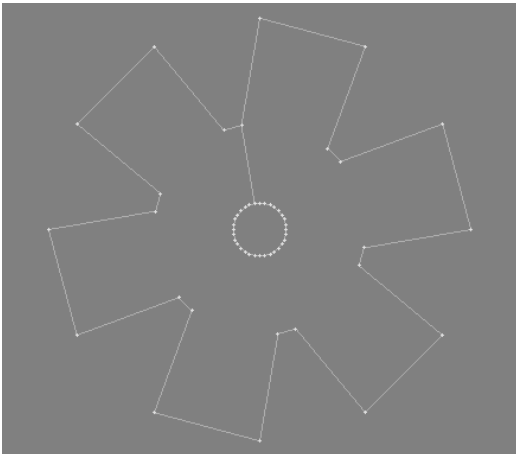


The **Inner Radius** determines the diameter of the gear's base. This is the location from which each of the teeth will project outward.



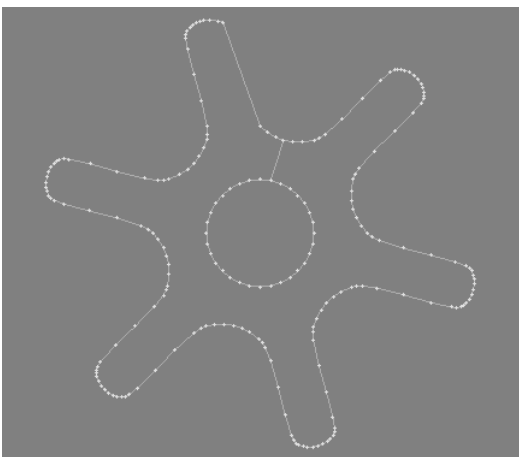


The **Outer Radius** determines the length of the gear's teeth. This is the location that the two points forming the top of each tooth will move towards. Note that it is possible to make the **Outer Radius** smaller than the **Inner Radius**. Doing this will still create a gear shape, although the teeth will appear to taper outward rather than inward.



The **Thickness** field determines the depth of the gear, which is the amount that the gear will be extruded. Note that the extrusion occurs along the Axis set by you at the top of the window.

Under **Gear Type**, **Angular** is the default option, and creates gears using the method described above. **Smooth** uses the Inner and Outer Radius to interpolate a smooth arc between the top and bottom of each tooth.



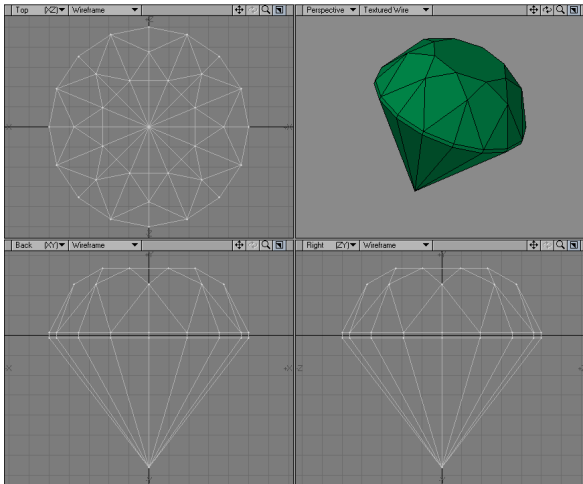
Bear in mind that smoothing is computationally intensive, and will take longer to calculate. Also, it will create an object with a higher number of points and polygons.

Finally, the **Center** option determines the exact center for your gear, which can be useful for positioning gears that are offset precisely from one another. The three fields correspond to the X, Y, and Z axis respectively.

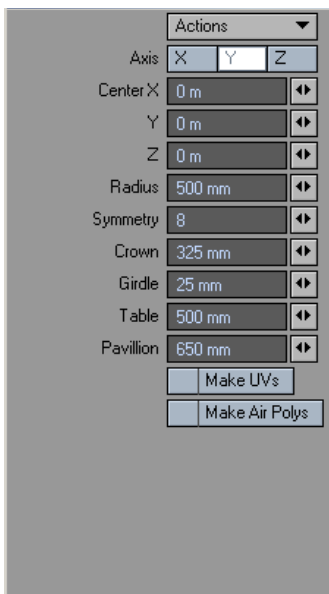


## Gemstone Tool

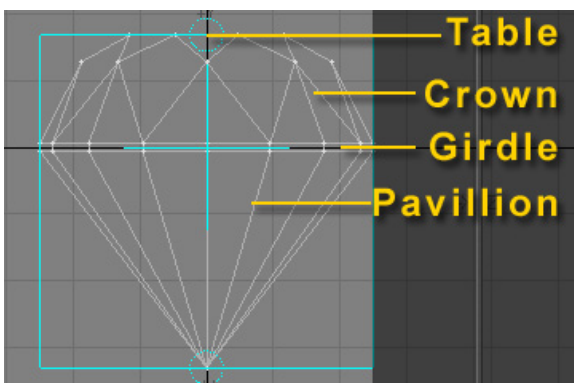
You can quickly make a nice round-cut diamond with the **Gemstone** tool (**Create > Primitives: Gemstone Tool**).



On the **Numeric Panel** you can control various aspects of your diamond. **Center** sets the center position and **Radius** the overall radius of the object.



**Higher Symmetry** values increase the number of polygons used on the crown and lower values decrease the number. The **Crown**, **Girdle**, **Table** and **Pavillion** settings control the size of those areas on the diamond. (See illustration below for reference.)



## Make UVs Option

The **Make UVs** option at the bottom of the various primitive **Numeric Panels** assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.

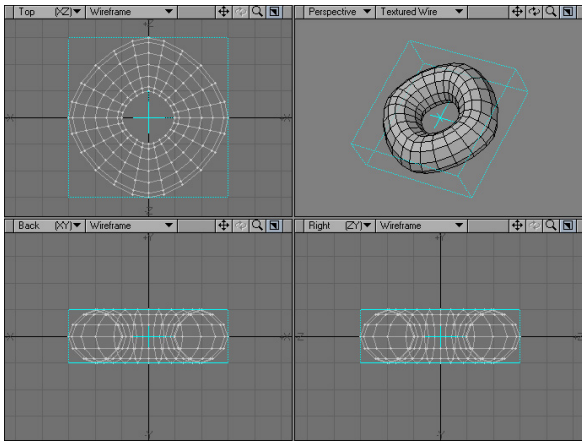
## Make Air Polys Option

**Make Air Polys** will create duplicate polygons facing inside the Gemstone with the surface name of "Air". These polygons should be used when surfacing glass or any kind of refractive material to give a more realistic refraction to your object.

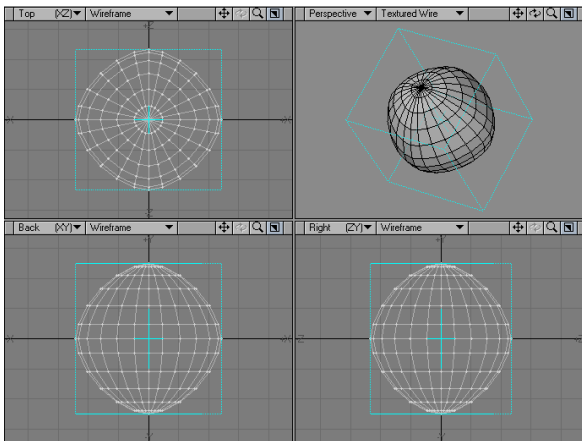


## SuperQuadric Tool

Use the **SuperQuadric** tool (**Create > Primitives: SuperQuadric**) to create quadrics objects interactively. You can choose between two shapes: **Ellipsoid** (spherical) and **Toroid** (doughnut).

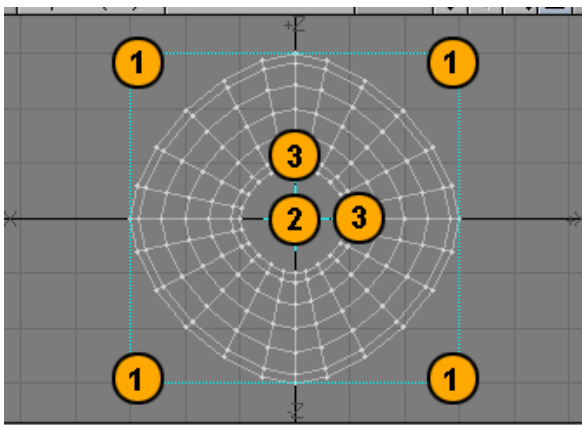


Toroid



Ellipsoid

You can use standard drag handles to manipulate the shape and position interactively.

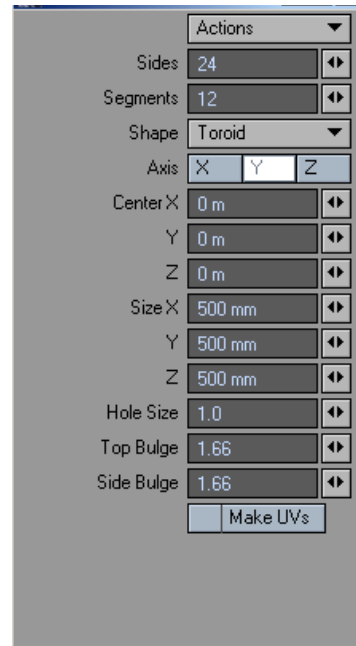


1 — Interactively sizes the object by clicking and dragging here with the LMB.

2 — Interactively Center the object by clicking and dragging here with the LMB.

3 — Interactively sizes the Hole in the object by clicking and dragging here with the LMB.

The **Numeric Panel** provides more detailed control.



**Axis** — determines the major axis for the object. This is also set when you click to first create the ball using the axis perpendicular to the (orthogonal) viewport.

**Sides** — determines how many horizontal segments should be used around the Object.

**Segments** — sets how many vertical segments should be used.

**Center** — The XYZ coordinates of the center of the Object.

**Size** — Overall dimensions of the object.

**Hole Size** — This option is only available for the Toroid Shape. This setting defines the inner radius of the toroid.

**Top Bulge** — This setting defines the vertical curvature of the geometry.

**Side Bulge** — This setting defines the horizontal curvature of the geometry.

A perfect circle — looking along the set Axis — can be made by setting the **Side Bulge** to 2. A value of about 5 yields a rounded square shape.

### Make UVs Option

The **Make UVs** option at the bottom of the various primitive **Numeric Panels** assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.

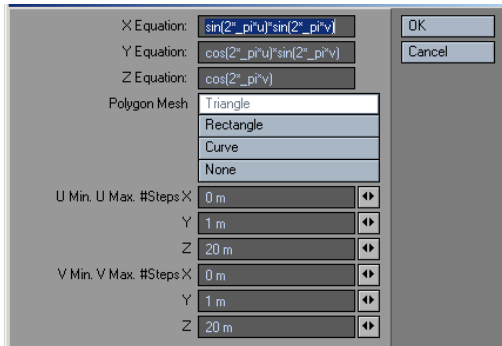


NOTE: A quadric is a shape made of the squares of the coordinates, a generalization of a sphere, which is  $x^2 + y^2 + z^2$ .



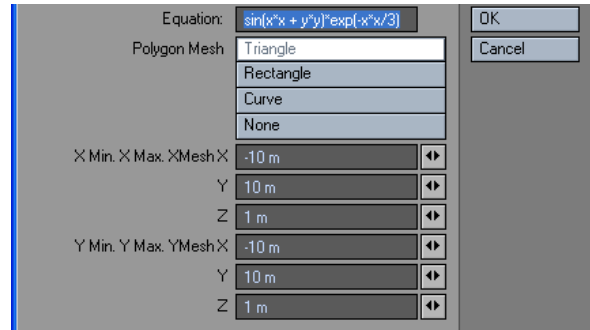
## The Parametric Surface Object Function

The Parametric Surface Object function (**Create > Primitives: ParametricObj**) creates parametric surfaces based on the equations entered for X,Y, and Z in terms of UV coordinates. The function lets you create the object using Triangles, Rectangles, Curves, or just points (None).

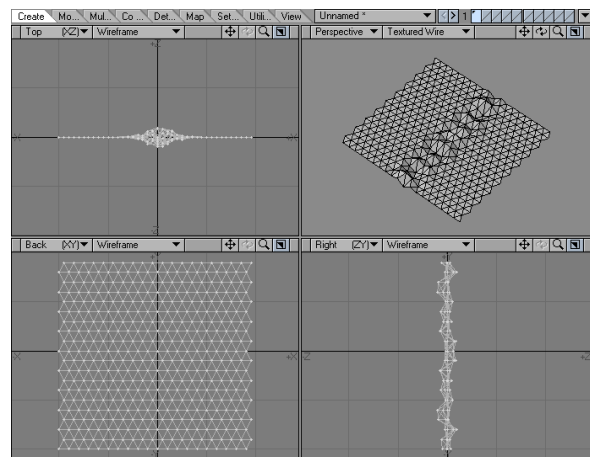


## Plot 2D Tool

The **Plot2D** function (**Create > Primitives: Plot2D**) generates a two-dimensional surface in the XY plane consisting of Triangles, Rectangles, Curves, or points (None). The height (in the Z direction) is determined by the value of the expression in the Equation field at each division in the X and Y axes.



In the fields labelled X Min, X Max, Y Min, and Y Max, enter the boundaries for the surface. In the XMesh and YMesh fields, enter the number of divisions along that axis.





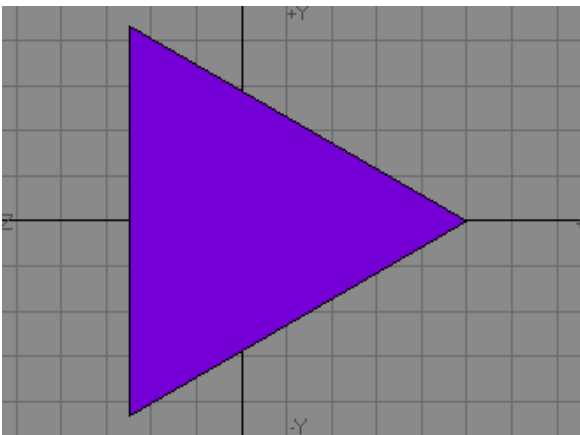
## Equilateral Triangle

Equilateral triangles are three-sided polygons whose sides are all equal in length. The **Equilateral Triangle** tool is found under **Create > Primitives > More**.



**Side Length** determines the size for each of the triangle's sides.

The **Axis** buttons determine the direction along which the polygon's normal will run, or simply, which way the triangle will be facing.



**HINT:** You also can create an equilateral triangle using either the **Ball** or **Disc** tool. If you hold down the **Ctrl** key or use the middle mouse button to drag out a circle in one viewport, you will constrain the width and height, ensuring a perfect circle. Now click **N** to bring up the **Numeric Panel** and set **Sides** to 3.

Using this technique, you can quickly and interactively create perfect triangles, diamonds, pentagons, hexagons, etc.

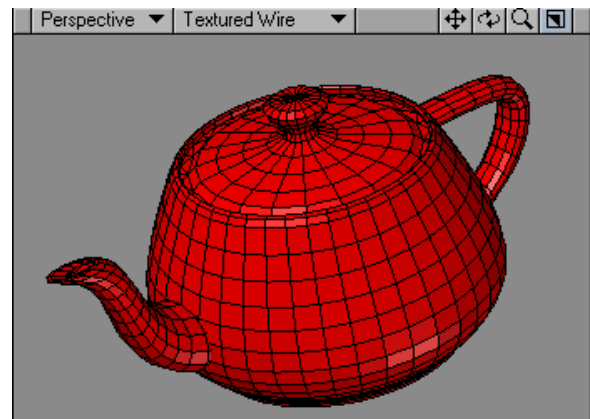
## Teapot

You've probably never heard of Martin Newell, but his work is as ubiquitous as the Mona Lisa. In 1974, the world of computer graphics imagery was being advanced at the University of Utah. When Mr. Newell built a 3D replica of his wife's tea set, he unwittingly fashioned an icon for the industry that would persist for more than three decades.

Dubbed the Utah Teapot, it was both beautiful in form and utilitarian in function. Its smooth curves and simple design made it an ideal candidate for testing new surface shading algorithms. By the time 3D Modeling and animation software reached the personal computer, the Utah Teapot had already become an industry standard.

The attributes that made this object so useful in the 70s still stand today. The teapot is the perfect stand-in object and is useful for testing lights and textures in your scene. Best of all, its low polygon count means that it will render quickly, even on older computers.

You can generate the Utah Teapot by simply clicking the **Teapot** button (**Create > Primitives > More**).



**NOTE:** The object will be generated with two principal surfaces: **TeapotChrome** and **TeapotGaps**.

Take pride when you invoke the **Teapot** primitive. For when you do so, you become a part of CGI history and join the ranks of those whose use of 3D has changed the world.

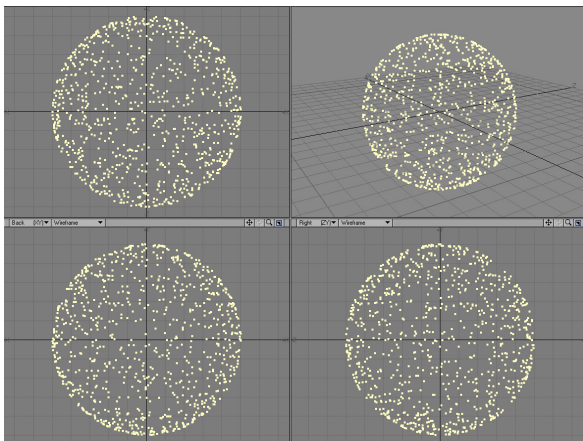


## StarSphere Tool

When you want to create a starfield with parallax, you can sprinkle some points around, convert them to polygons and give them surface names, but why bother? NewTek engineers have done all the hard work for you with **StarSphere**. This generates up to nine layers of single point polygons which all have different levels of magnitude allowing you to create convincing starfield models quickly and easily.



The first parameters to set are how big you want the **StarSphere** to be and whether you want it spread over nine layers, or have it compressed down to one. The remaining settings are how many of each sort of star you wish to have in your **StarSphere**. Actual magnitude distributions are hard to come by and vary from location to location, so the default settings have been left at 100 stars of each magnitude, but it's likely that you will have far fewer bright stars than faint ones, so your lower magnitude stars (Magnitude 2 - 6) should probably vastly outnumber your brighter stars (Magnitude -2 - 1).



NOTE: Using the **StarSphere Generator** will create a new object.

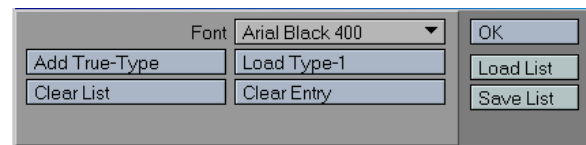
## Text Tools

The Text tools provide several ways to create 3D text objects, from a single word to multiple lines of text. A variety of controls over the look of the final text are provided, including beveling and extrusion depth, as well as how smooth the curved portions of letters will be.

Please note that before you can use any of the text tools you will need to have at least one font loaded. For this reason, though it is the last item on the Text tools list, we begin this section by covering the **Manage Fonts** panel first.

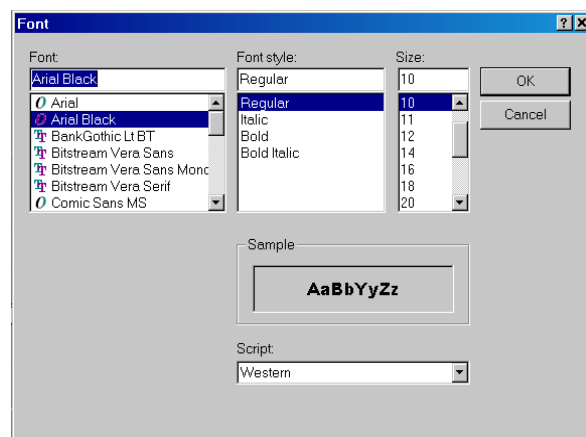
## Manage Fonts

(default keyboard shortcut **F10**)

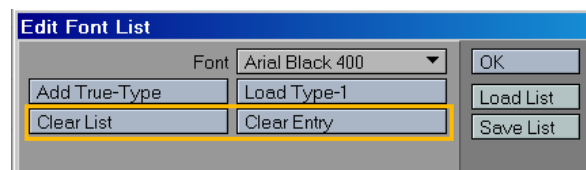


Before you can use any of the **Text** tools, you must have previously loaded one or more fonts into the font list. This is done using the **Manage Fonts Panel (Create > Text: Manage Fonts)**.

Use the **Font** pop-up menu to select the font for your object. Clicking the **Load Type-1** button brings up a file requester where you can load a PostScript font. If you have PostScript fonts, the files often have a .pfb filename extension. Clicking the **Add True-Type** button brings up a font selection requester you can use to select a font. (Note: the point size is irrelevant.)



Click **Clear Entry** to remove the selected font from the font list. **Clear List** will purge the entire list.



**Load List** brings up a file requester where you can load a previously saved font list file. This can be the Lwm.cfg file. **Save List** brings up a file requester for saving the current font list to a file. Note that the font list is automatically saved when you exit Modeler, so it is not mandatory that you save the list. Use this option to create custom file lists for special purposes.





NOTE: The clearing operations do not affect the actual files stored on your hard drive. Also fonts in the list utilise no RAM until used.

## Text Tool

(default keyboard shortcut **Shift W**)

The **Text** tool (**Create > Text: Text**) lets you interactively create type objects using True Type and Adobe PostScript Type 1 fonts. Once you create them, these objects can be bevelled, extruded, drilled, and more.

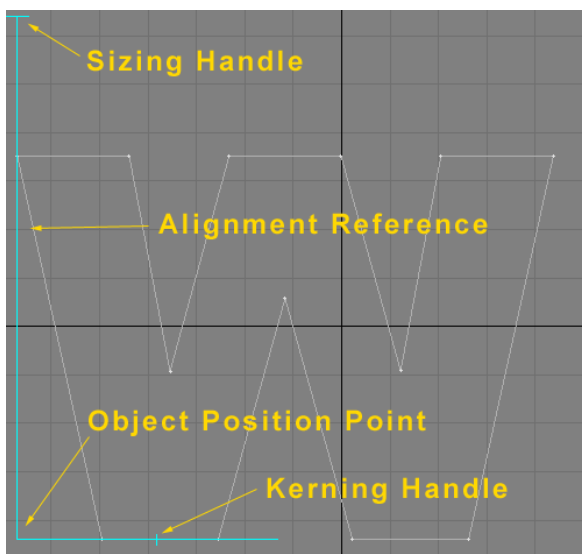


NOTE: If the **Text** tool button is ghosted, make sure you have added some fonts, as discussed on page 106.

### Interactively Creating Text

When you first click in a viewport after selecting the **Text** tool (**Create > Text: Text**), the text insertion pointer appears. You can just start typing to create the **Text** template.

### The Text Insertion Pointer



You can use the **Backspace** or **Delete** keys to clear the last character typed. To clear the entire text line, press **Shift + Backspace** or **Shift + Delete**. If you need to use a Modeler keyboard shortcut, like the **N** key, while using the **Text** tool, first press the **Esc** key to exit the **Text** tool, then press the desired shortcut key.

## Modifying the Template

The position of the text is referenced to the bottom left of the pointer. You can reposition the text in any viewport. Dragging the sizing handle up or down will scale the text larger or smaller. Dragging the kerning handle will adjust the amount of space between characters.

Text will be aligned with reference to the vertical bar. Pressing the **Tab** key will cycle through the alignment options.

You can cycle through the font list using the Up and Down cursor keys or you can use the numeric requester. If you have already started a template, the font will change.

You must deselect the **Text** tool to actually create the object. This can be done by clicking on any other tool button, mode button, the Make button or hitting the Return key. The Text tool cannot be dropped with the **/** key (Drop tool) because it is interpreted as text input. Alternatively, clicking your **RMB** will create the object and also move the template to your mouse pointer position. You can quickly make unlimited copies of the text in this manner.

The number of points used to approximate curves in characters depends on the setting for **Curve Division** on the **General Options Panel** (**Modeler > Edit > General Options**).



HINT: If you realise you are creating your text in the wrong viewport, click in the grey empty area of the interface — the text you have created in the viewport will disappear, but you will notice that the Text tool button remains highlighted and active. Click in the correct viewport, and your text will reappear in that viewport.

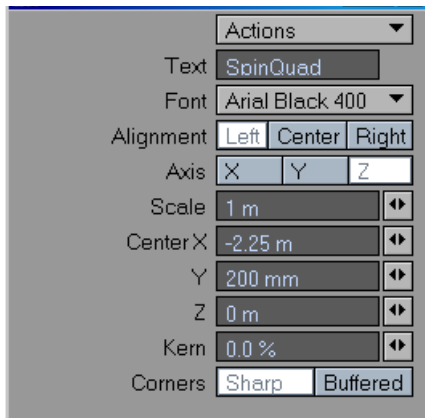


NOTE: The characters that are available within the selected font may be limited. You may or may not have all possible characters allowed by the font type depending on whether those characters were created in the font originally.



## The Text Numeric Panel

The related **Numeric Panel** provides interactive feedback and also lets you set options numerically. Axis is the perpendicular axis used in creating the text; however, it is usually set interactively by clicking the mouse in a view window. The Text input field lets you enter the text to generate into a 3D object.



From the **Font** pop-up menu, you can select any of the fonts defined in the previously discussed Fonts List.

Some fonts, when converted into 3D objects, have additional points near their corner vertices. When these polygons are beveled, these additional points can cause beveling inaccuracies. Use the default setting, **Sharp**, for the majority of your text generation, as this will avoid the creation of such points. The **Buffered** setting allows the additional points to be created.

The **Alignment** setting determines how the text is aligned around the text insertion pointer. **Center** is the XYZ position of the insertion point; however, it is usually set interactively by clicking the mouse in a view window. The **Scale** value sets the general height of the text; however, the actual size may be less and will vary from font to font. **Kern** sets the spacing between fonts and depends on the font's own characteristics. Negative numbers are allowed to bring characters closer together.

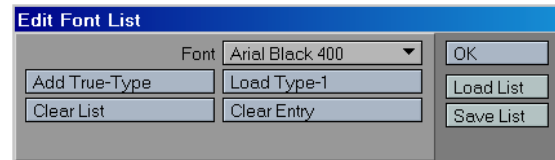


**NOTE:** The **Scale** setting (**Numeric Panel**) cannot be set to zero or a negative number.

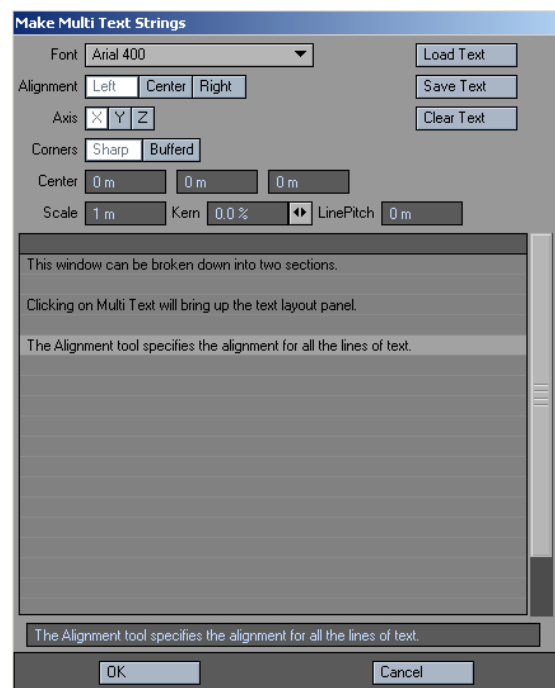
## Multi Text

The **Multi Text** tool (**Create > Text > Multi Text**) generates multiple lines of text and gives you control over size, kerning, and leading. In addition, it can load ASCII text files, making it an extremely versatile text generation tool.

As with the other text tools, you must have at least one font loaded into Modeler before you can use this utility. Fonts in both TrueType and Postscript formats can be loaded using the **Edit Fonts** tool (**Create > Text > Manage Fonts**).



Clicking on **Multi Text** will bring up the text layout panel.



This window can be broken down into two sections. The top section contains text formatting options, while the bottom section contains text entry options.

You begin by choosing your typeface from the **Font** pulldown menu. Remember that only fonts which have been loaded with the **Edit Fonts** utility will be displayed in this list.

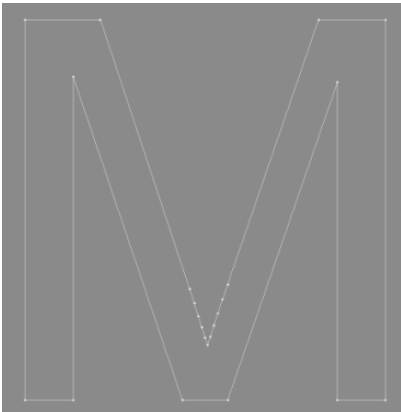
The **Alignment** tool specifies the alignment for all the lines of text. The choices of Left, Center, and Right operate just like their counterparts in word processing programs.

The **Axis** buttons determine which direction the text object will face once it has been created.

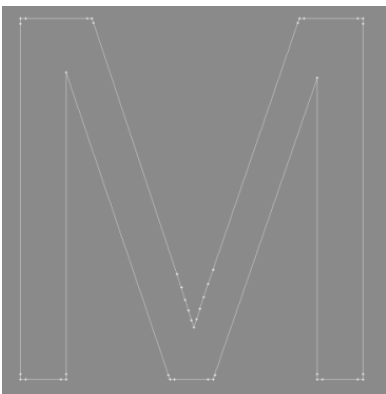
The **Corner** buttons determine whether or not extra points will be created around the hard angles of the characters in your font. Many letters, like an M, have sharp angles. When you extrude these letters, you wind up with sharp edges, which look unrealistic when rendered because they lack the tiny bevels needed to catch the specular highlights. These bevels require extra points in the letter's polygons; and by selecting **Buffered Corners**, the extra points needed to round and refine your polygon's sharp edges are added.



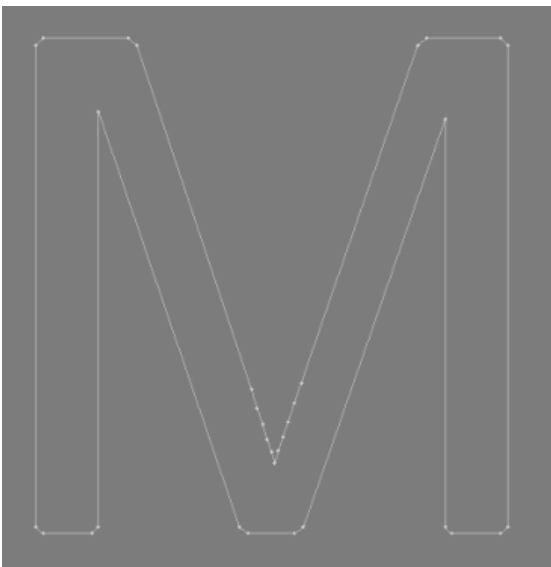
The following picture shows the letter M generated with **Sharp Corners**.



Notice that almost every corner forms an angle of 90-degrees or more. Now take a look at the same character generated with **Buffered Corners**:

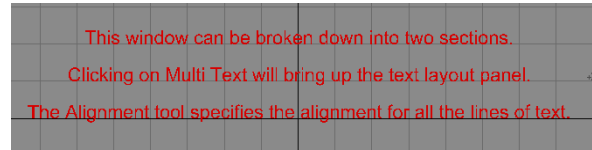


Extra points have been created around the sharp corners. These points act as a buffer, allowing you to move or delete the points at the edges to create a bevelled shape (as in the example below).

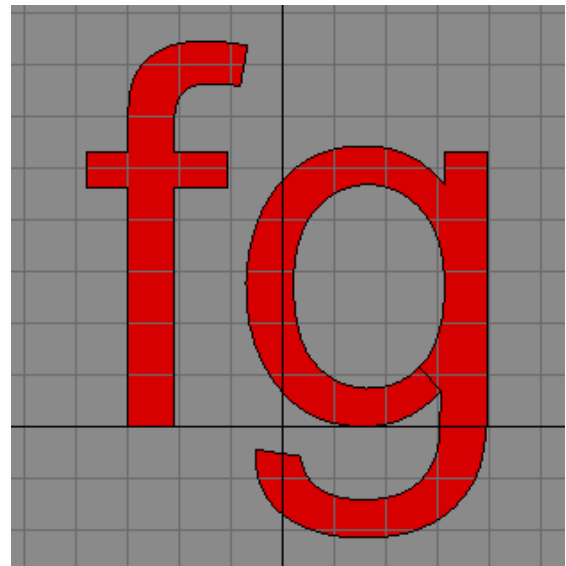


Occasionally, when making a relatively large bevel with the **Bevel** tool, buffer points can cross over each other and produce render errors. If this is the case, try regenerating your text using **Sharp Corners** and using the **Edge Bevel** tool to compensate for any sharp edges.

The **Center** input fields correspond to the X, Y, and Z coordinates, which determine both the left margin and base for the text object. More often than not, this can simply be left at its default, of 0, 0, 0 that used on a left-aligned text block, will produce an object which rests on the ground and is situated to the right of the origin. The image below uses center alignment.



The **Scale** setting determines the height of the font from its lowest point to its highest point. It's important to note that the **Scale** setting does not measure the height of your font from the baseline up. Certain characters, such as the lower case p, q, and g, all fall below the baseline. And characters like f, h, and k rise higher than the more common a, s, and e. The **Scale** setting looks at the lowest and highest points in your typeface and scales the font so that the range between them matches your target size.



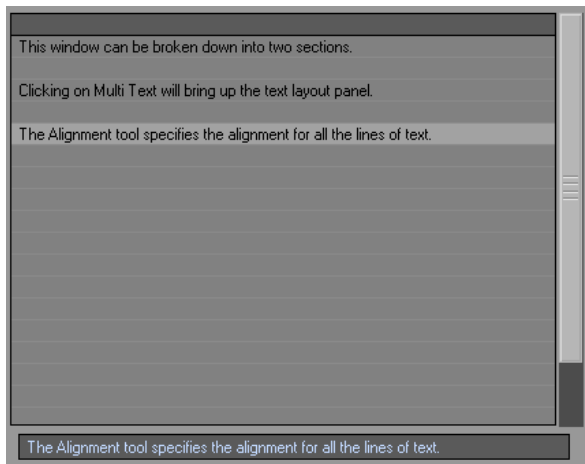
The **Kerning** setting determines the spacing between letters. Kerning is determined by "em" space and is measured in percentages, with 100% being equal to the width of the capital M in your chosen font. Negative values will bring your letters closer together. Positive values will spread your letters farther apart.

**Line Pitch** refers to the spacing between lines of text (also known as leading). A value equal to the Scale of your text will usually produce good results. Increasing this value will push the lines further apart. Decreasing this value will pull the lines closer together.

The **Load Text** and **Save Text** buttons enable you to import and export ASCII text files for use with the **Multi Text** tool. ASCII character codes are respected in these files, allowing you to utilise the extended character set of most fonts.

The **Clear Text** button will erase all text in the composition window, leaving you with a clean slate.

The lower half of the **Multi Text** tool enables you to enter and compose blocks of text.



Text is entered one line at a time, and text lines can be entered in any order you desire. Before entering text, you must select the line onto which the text will be placed. The selected line will highlight as in the example above.

Once a line has been selected, text can be entered into the field at the bottom of the window. Pressing the **Enter** or **Tab** key will place the text into the selected line.

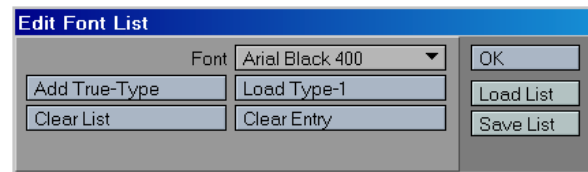
To edit a line of text, simply select it with your mouse. The text from that line will appear in the entry field at the bottom of the window. Proceed to edit the text, then press **Enter** or **Tab** when you are finished.

To clear a single line of text, select it with your mouse, then delete the text from the entry field at the bottom of the window. Pressing **Enter** or **Tab** will accept your changes and wipe the line clean.

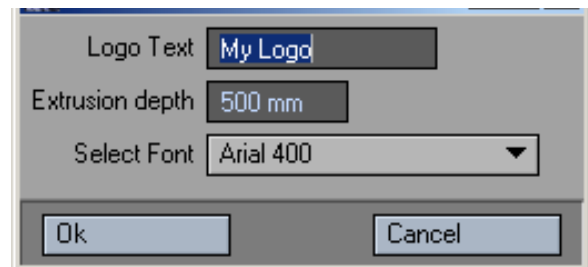
When you are satisfied with your settings and text layout, press the **OK** button. A new text object will be generated. Should you find that a particular setting was incorrect (such as the Kerning or Line Pitch), simply delete your text object and run the **Multi Text** tool again. **Multi Text** retains the settings from its last use through the duration of your Modeling session. Note that once Modeler has been shut down, however, your **Multi Text** settings will be lost.

## Logo

The **Logo** tool (**Create > Text > Logo**) generates 3D text. Before using this utility, you must have at least one font loaded into Modeler. Both TrueType and Postscript fonts can be loaded using the **Edit Fonts** tool (**Create > Text > Edit Fonts**).



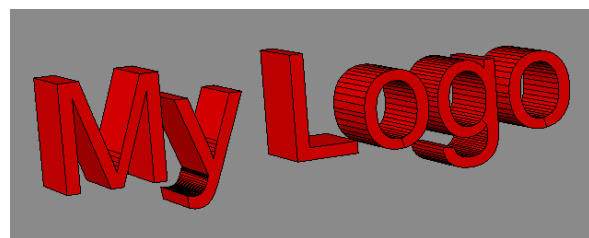
The **Logo Maker Panel** is straightforward.



In the **Logo Text** field, you enter the word or phrase that you want extruded. Multiple lines of text are not supported. (Multiple lines of text can be created with the **Make Text** or **Multi Text** tools.) ASCII characters may be entered by pressing the **Alt** key and the four-digit ASCII code. ASCII characters may not display properly in the **Logo Text** field, however they will be properly generated in the final object.

The **Logo** tool will create your text object in Modeler's **Back** Viewport and extrude it along the positive Z axis. The **Extrusion Depth** setting determines the size of the extrusion.

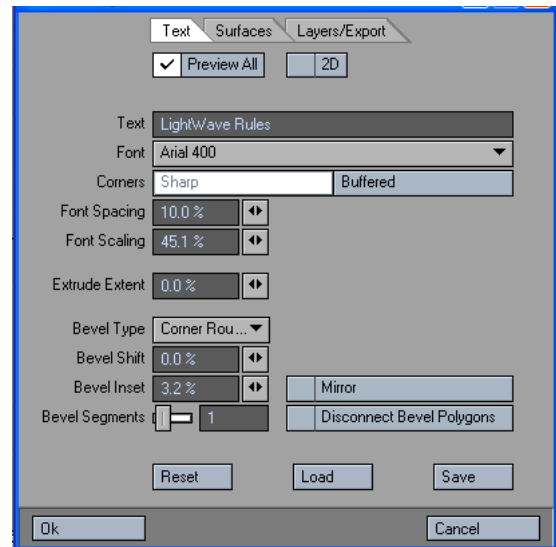
You choose the typeface in the **Select Font** popup list. Remember that only fonts that have been loaded with the **Edit Fonts** tool will be displayed in this menu.



**HINT:** If you use a Dingbat font with the **Logo** tool, you can quickly generate 3D arrows, starbursts, and other common design elements.



## Text Layers



The Text Layers tool allows you to type letters, numbers, and other symbols and automatically have them beveled and extruded. Text Layers also has multiple layers and settings options.



menu.

Note: If Text Layers does not show up in your Utilities>Plugins>Additional Menu, you might need to run Scan Directory in the Edit Plugins



Hint: You will need to have fonts loaded in the Manage Fonts panel (F10) before you start using Text Layers.

### Text Tab

**Preview All:** Checked, this previews all text. Unchecked, this previews only the first character.

**2D:** Checked, previews the text in 2D only.

**Text:** This is where you enter all characters. You can even copy and paste from a text editor and all characters will be copied over until a carriage return is found.

**Font:** Select the font you want to use here.

**Corners:** Some fonts, when converted into 3D objects, have additional points near their corner vertices. When these polygons are beveled, these additional points can cause beveling inaccuracies. Use the default setting, Sharp, for the majority of your text generation, as this will avoid the creation of such points. The Buffered setting allows the additional points to be created, should you decide that you wish it.

**Font Spacing:** Increases/Decreases the amount of spacing between characters. Negative numbers will start spacing the



letters in reverse.

Font Scaling: Increases/Decreases the size of the characters.

Extrude Extent: The amount the characters will be extruded.

Bevel Type:

Single- The bevel is made with only one segment

Round- The bevel is made with a convex edge

Corner Round- The bevel edge is rounded at the corner of the segment

Hollow- The bevel edge is made with a concave edge

Corner Hollow- The bevel edge is inset with a Hollow edge

Bevel Shift: The distance the characters will be beveled out.

Bevel Inset: Affects the size of the bevel.

Bevel Segments: The number of segments created in the bevel.

Mirror: Mirrors the effects on the characters along the axis.

Disconnect Bevel Polygons: Unwelds the beveled edge from the rest of the model.

Surfaces Tab

Separate Surfaces: Checked, allows you to save a surface name for Face, Side, and Bevel. Unchecked, a single surface name will be given to all of the characters surfaces.

Face: The front and back surface of the characters

Side: The side surface of the characters

Bevel: The beveled edge of the character mesh

Layers/Export Tab

Determines how the characters are laid out in Layers for Modeler.

Layers: Per Letter, creates a new layer for each character. Per Word, creates a new layer for each word. Single, creates one layer for entire text.

Center All: Checked, will center all text at 0,0,0.

Pivot Positions: Determines the pivot point for each layer. Center will create the pivot point at the center of the text in each layer. Bottom Left will create the pivot point at the bottom left of the text.

Create Object: Create an object file for the text.

Object File: Determines the name and location of the object file.

Create Scene: Creates a scene file for the text.

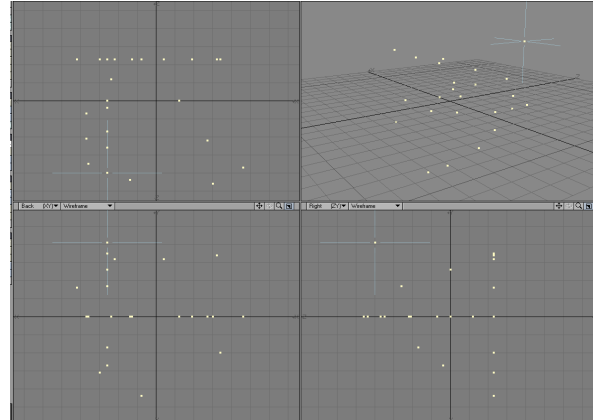
Scene File: Determines the name and location of the scene file.

## Points

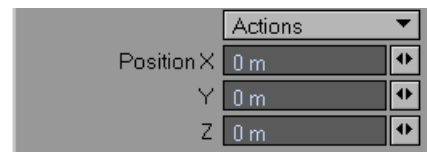
Points are the fundamental building block of 3D objects and have an XYZ location. LightWave provides a variety of tools for point creation, ranging from one at a time with precision to mass creation of points with random locations.

## Point Tool

(default keyboard shortcut **+**)



Point coordinates are the minimum information from which the geometry of an object can be calculated.



Remember a point needs to have X, Y, and Z coordinates. You can create points in real-time or by positioning first and then creating. As with every tool in Modeler you can choose to create a point with the **Numeric Panel**.

### Creating Points

**Step 1:** Choose (**Create > Points: Points**) from the menu or use the **+** key to activate the **Point** tool.

**Step 2:** In any viewport, with your **LMB** drag the large crosshairs to the desired position (you can reposition in any viewport). The information display (bottom-left corner) will give you position feedback as you move your mouse.

**Step 3:** Click the **RMB** to create the point. Be careful not to move your mouse pointer. Deselecting the **Points** tool will also create the point, if you don't have a steady hand. Deselecting and then reselecting the **Points** button will create the point and keep you in the create point mode.

**Step 4:** You can also position the point by dragging with your **RMB**. However, the points will all be created along the same plane (when you release the **RMB**).

### Alternative Method to Create Points

**Step 1:** Choose (**Create > Points: Points**) from the menu or use the **+** key to activate the **Point** tool.



**Step 2:** Move the mouse to the location where you wish to create a point — the information display (bottom-left corner) will give you feedback as you move your mouse — and click the **RMB** to immediately create the point.

You can click down a point with the **RMB** and then, if you need to adjust the location, immediately click the **LMB** and move the point. Alternatively you can simply click the **RMB** to quickly lay down a string of points in succession and move them later using something like the **Drag** tool.



NOTE: You can convert the point you've just made into a polygon by pressing **P**.

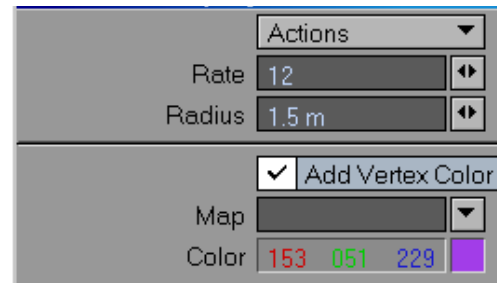
## Spray Points Tool

Creating one or two points is not hard, but what happens when you need to create hundreds or even thousands of random points for objects such as stars, gasses or bubbles? Enter the **Spray Points** tool, (**Create > Points: Spray Points**), which works like a splatter gun to create a random assortment of single point polygons.

After selecting the tool, click in any viewport (including the perspective viewport!). A blue crosshair appears surrounded by a larger blue circle. The crosshair represents the center position of the **Spray Points** tool. The circle represents the radius within which points will be created.



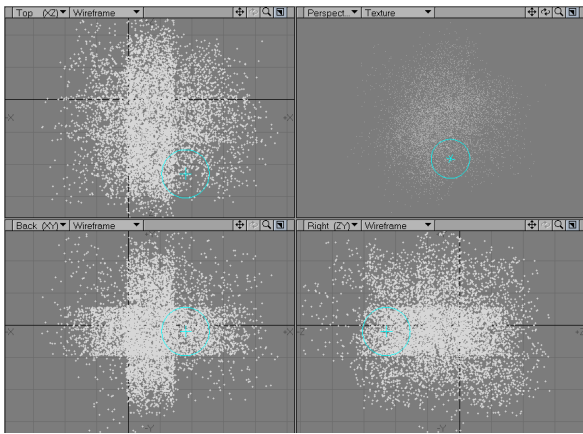
The radius can be adjusted either numerically or interactively. To adjust the radius numerically, open the **Numeric** window (**N**).



Click in the field entitled **Radius**, type a new number and press **Enter**. (You can also use the slider to the right of the numeric field to make the radius larger or smaller).

To adjust the radius interactively, click the **RMB** in any viewport and drag. The radius circle will jump to the position of your mouse cursor. To make the radius smaller, drag your mouse towards the crosshair. To make the radius bigger, drag it away from the crosshair.





To create points, drag your mouse while the **LMB** is held down. Simply clicking the **LMB** will not result in the permanent creation of new points. To see this in action, try clicking once, then moving your mouse to a new location and clicking again. You'll notice that the points from your previous click disappeared and a new set of points was created under your mouse pointer. For the tool to generate permanent points, you must continue to hold down the **LMB** while dragging in a viewport.

The number of points created as you drag can be adjusted by the **Rate** setting on the **Numeric Panel**. A low **rate** setting will cause fewer points to be created as you move your mouse. A high rate setting will cause more points to be created.

The **Spray Points** tool doesn't just create points, it creates single point polygons (also known as Particles), which means they can have **Vertex Color Maps** applied to them and will be visible in a render with those assigned characteristics. Points themselves can have vertex maps assigned to them regardless of whether they have polygons attached to them or not. In the case of color maps those cannot be seen unless the points are attached to polygons.

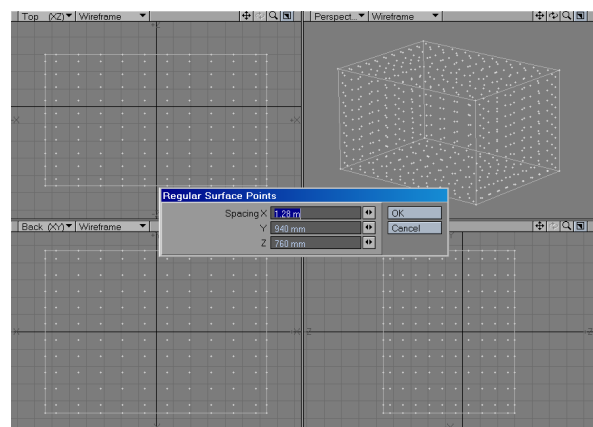
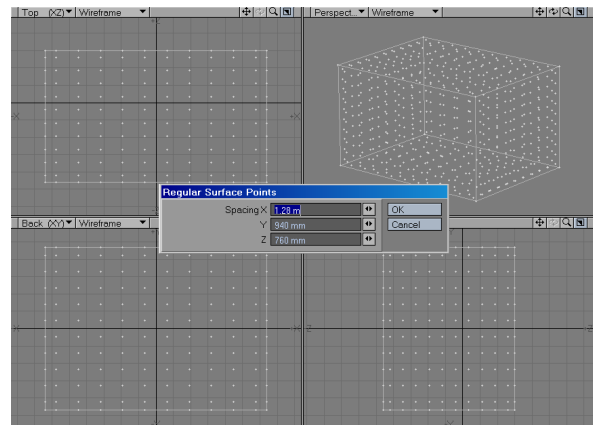
If you would like to use a **Color Map** with the **Spray Points** tool, simply click the **Add Vertex Color** checkbox and type a name for it into the **Map** field. If you have already created one or more **Vertex Maps**, you can use the drop down arrow to select one from the list.

Modeler will not display the **Vertex Map** in any of its viewport displays, however you can see the effect of the **Color Map** by running **Vertex Paint** from Modeler's **Map** menu.

## Stipple

The **Stipple** tool (**Create > Points**) will cover the surface of your object with an even array of points. When used in conjunction with the **Kill Polys** tool (**K**), (**Construct/Reduce/Remove/Remove Polygons**), it is an effective way to generate uniform and predictable arrangements of points.

To use the **Stipple** tool, place your geometry into the foreground layer. When you click the **Stipple** button, the **Regular Surface Points** Panel appears with fields for X, Y, and Z spacing for the points which the **Stipple** tool will create.

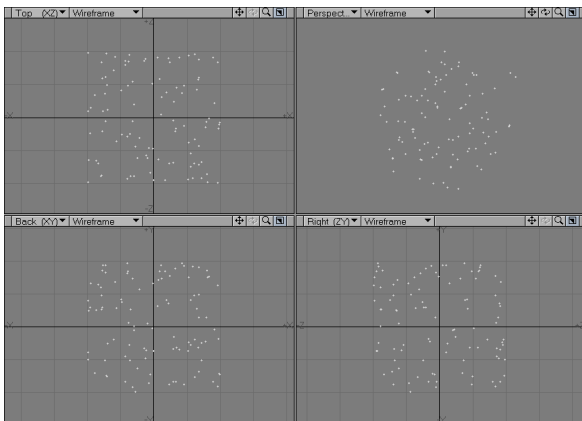


The initial spacing values reflect the dimensions of your object and are a good place to start experimenting. (You can undo the results of the **Stipple** tool by hitting **Ctrl Z** twice.) If you lower the **Spacing** settings, you will generate more points and position them closer together. Increasing the **Spacing** settings will generate fewer points and position them further apart. As always, the smaller the spacing settings, the longer the processing time.

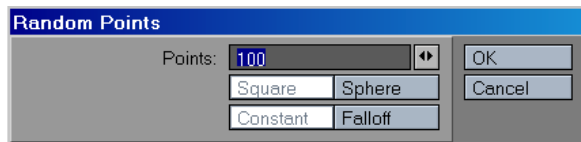


## Random Points

Use the Random Points command (**Create > Points: Random Points**) to create a defined number of points distributed randomly.



You can choose between a **Square** or **Sphere** shape. **Constant** generally confines the points to the selected shape, while **Falloff** tapers off the point distribution along the perimeter.

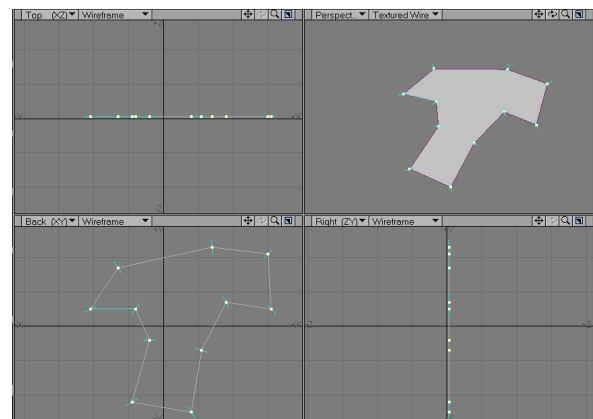


## Polygons

Polygons are multisided geometric shapes which are arranged so as to make a continuous surface for a 3D object. LightWave provides several tools for creating polygons, some of which use points you have already created, and some of which create the points as well as the resulting polygons.

### Pen Tool

The **Pen** tool (**Create > Polygons: Pen**) gives a quick way to create polygons on the fly, and creates the points as part of the process.



#### To create a polygon with the Pen tool:

Choose **Create > Polygons: Pen** and click in a viewport with your LMB. A polygon is created using points you define by clicking. If you drag with the mouse button, you can refine the point's position before you create it, which happens when you release the mouse button. You do not need to click in the same viewport for each point. Once you have the polygon defined as you wish, you can finalise it by clicking the Pen tool off.

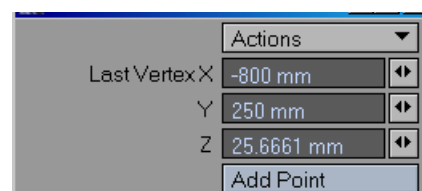
If you wish to create multiple polygons, then once you have completed a polygon, move the mouse cursor to where you wish to place the first point of a new polygon, click the **RMB**, then continue to build the new polygon using the **LMB**.



**NOTE:** While the pen tool is active, you can click on and drag any point using the **LMB** to edit its position.



**HINT:** If you create a new point in your polygon by clicking too closely to an existing point, Modeler can think you are editing the existing point and not create the point. If this is the case, simply click at a location that is not close to an existing point, then drag the new point to its desired location.





The **Add Point** button allows you to interactively add a point to your newly created polygon.



NOTE: The point will be positioned at the same location as the last created point, so may not be obvious until you change its position either with the numeric panel, or by clicking on it and dragging with the **LMB**.



NOTE: Once you create a polygon, you'll want to view it in the **Polygon Selection** mode to see if its surface is facing the correct direction based on the normal, as discussed elsewhere. If you are using the Back view for example, creating points clockwise will make a polygon with its *face* or normal facing you. Counter-clockwise creation of the points will make a polygon that faces away.

## Make Polygon

(default keyboard shortcut **P**)

The Make Polygon tool (**Create > Polygons: Make Polygon**) allows you to turn selected points into a polygon. Excluding the one and two-point specialty polygons, polygons should contain at least three points. You will likely find that your finished objects are constructed of a variety of polygons with different numbers of points.



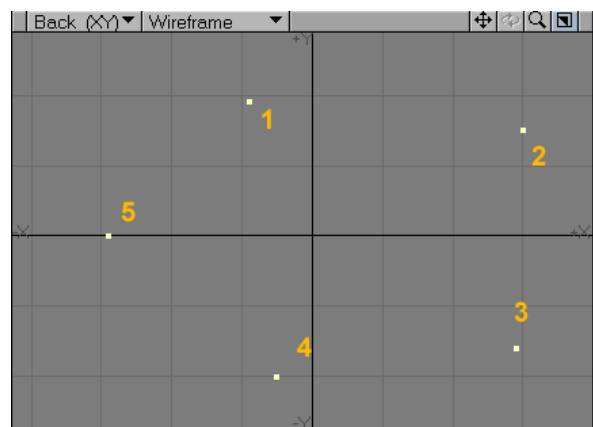
NOTE: Don't get discouraged if making polygons seems very tedious. Not that it isn't, but most of the time you'll create your objects by starting with primitives. Usually, creating polygons from points is the exception rather than the rule.



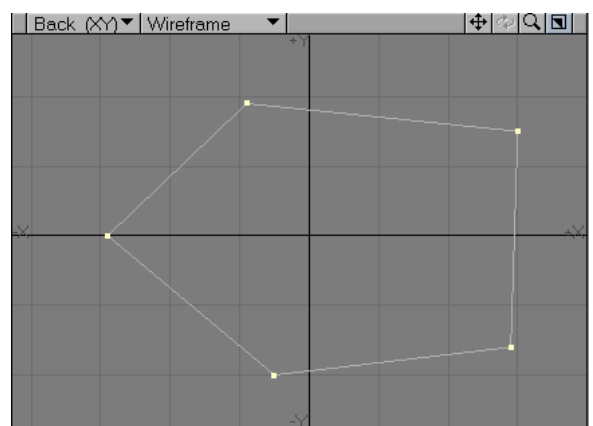
NOTE: You might hear artists talking about building objects point by point. This is the method they are referring to.

### To create a polygon from points:

**Step 1:** In the **Point Selection** mode, select your points in a clockwise order.



**Step 2:** Then, click **Create > Polygons: Make Polygon**.



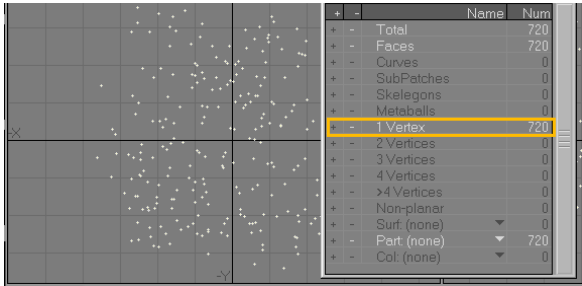
NOTE: Once you create a polygon, you'll want to view it in the **Polygon Selection** mode to see if its surface is facing the correct direction based on the normal. If you are using the back view for example, selecting points clockwise and making a polygon will create it with the *face* or normal of the polygon facing you. Counter-clockwise selection of the points will create a polygon that faces away.



## Points to Polys

### Single-Point Polygons

Sometimes you will want to turn single points into polygons. This is necessary for objects like stars in a starfield, since you cannot assign surface attributes to a point, and points don't appear in renders. Choose **Create > Polygons: Points to Polys** to turn selected points into single-point polygons (also known as Particles).



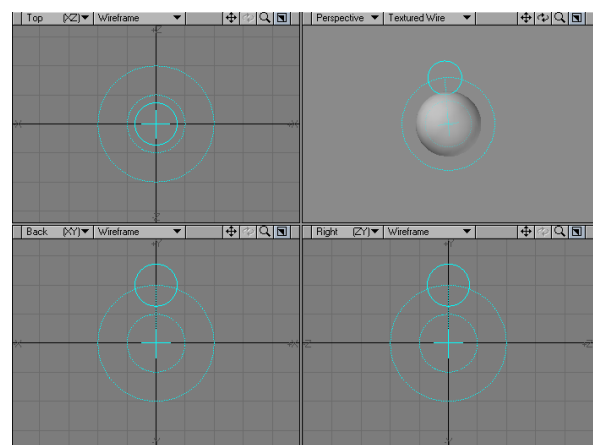
## Metaballs

**Metaballs** is one of three **Meta-primitives** in LightWave. **Meta-primitives** are similar to **HyperVoxels** except that they are polygon-based, and their surface and interaction is more apparent and viewable in Modeler. In fact, **Meta-primitives** are a good place to start before you set up with **HyperVoxels**, because of their real-time feedback. All Meta-primitives will interact with each other if they are in the same layer, even though they are different types..

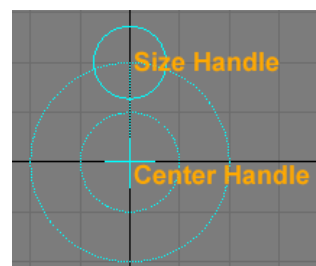
### To draw a Metaball object:

**Step 1:** Activate the **Draw Metaballs** tool. (**Create > Polygons: Metaballs**)

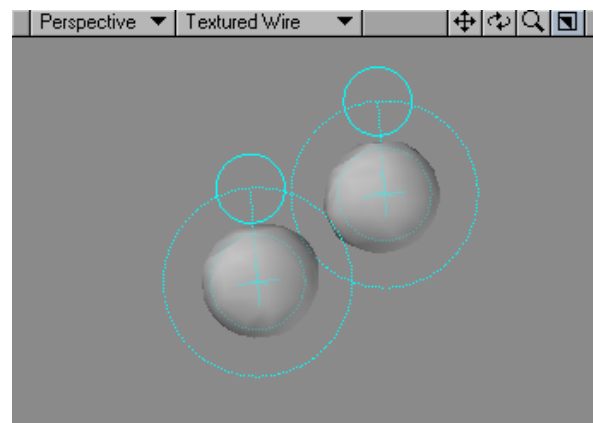
**Step 2:** Click in a Modeler viewport. Use any **Rendering Style** other than Wireframe and you will see the metaball



**Step 3:** You can move the metaball by dragging its center handle. You can change its size by dragging the sizing handle that looks like an orbiting circle.

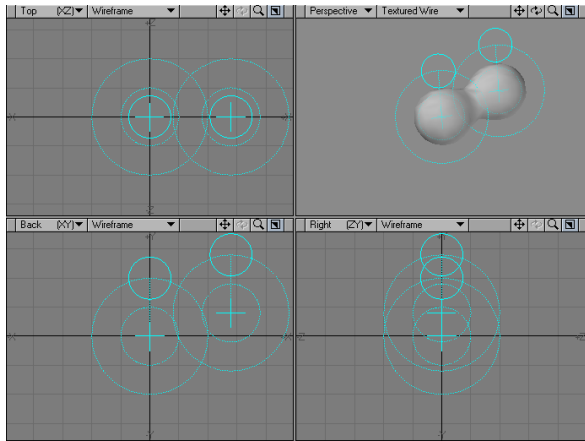


If you click anywhere else, another Metaball will be created.





**Step 4:** If you move or increase a metaball so that its surface approaches the surface of another, they will adhere to each other, like the stuff in a lava lamp.



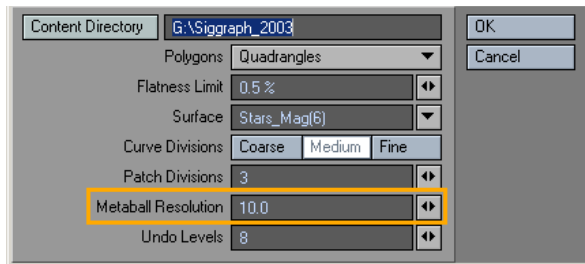
**Step 5:** Drop the **Draw Metaballs** tool (i.e., reselect it or press the space bar).

### Converting to Metaballs

You can also convert *points* into plain Metaballs using the **Make Metaballs** command (**Construct > Convert: Convert Metaballs**).

### Metaball Appearance

The amount of detail for *displaying* Meta-primitives is handled by the **Metaball Resolution** setting on Modeler's **General Options Panel** (**Modeler > Edit > General Options**). The value represents the number of subdivisions (pixels per metaball on the screen). To get a smoother surface, increase the value. There is no limit.



You can also toggle the Meta-primitive mesh on/off by choosing **Construct > Convert: Toggle Metamesh**. You may want to do this if you have a lot of Meta-primitives and your display refresh is too slow. This is not just a display change and is similar to toggling off SubPatches. If you save a Meta-primitive with the mesh off, you will not be able to see the surface in Layout.

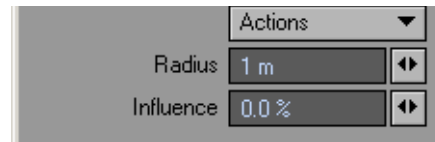
You can surface Meta-primitives as you would any object; however, you are allowed only one surface per layer.

### Animating Meta-primitives

You can animate Meta-primitives the same way you would animate any object. Of course, you won't take advantage of their benefits unless you animate point positions. For this, you can use features like morph targets, bones, **Displacement Maps**, and so on. Note that only Meta-primitives in the same layer will interact with each other.

### Editing Meta-primitives

Activate the **Edit Metaballs** tool (**Construct > Convert: Edit Metaballs**) to edit existing Meta-primitives. On the **Numeric Panel**, you can adjust the **Radius** and **Influence**.



### Radius and Influence Settings

The **Numeric Panel** has **Radius** and **Influence** settings.

**Radius** is simply the size of the Meta-primitive ball. Adjusting the **Influence** changes the interaction between balls. Setting a negative **Influence** will *invert* the primitive causing it to *eat away* neighbouring Meta-primitives instead of blending.

Deactivate the tool by dropping it, reselecting it, or hitting the space bar when you are done.

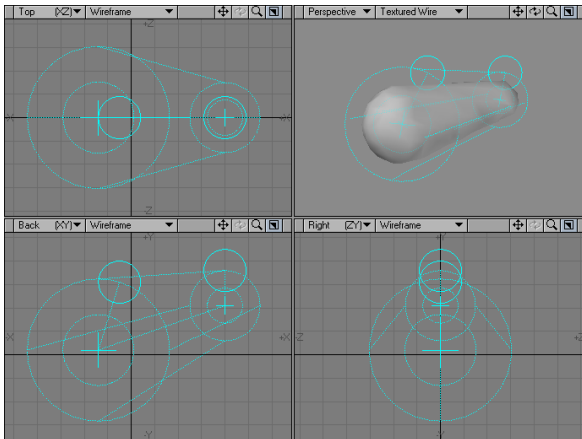


## Meta Edges

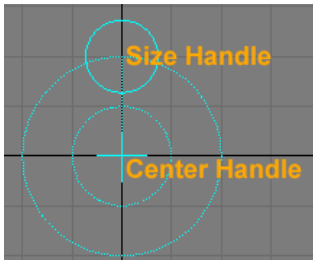
**Meta-Edges** is one of three **Meta-primitives** in LightWave. **To draw Meta-Edges:**

**Step 1:** Activate the **Draw Meta-Edges** tool. (**Create > Polygons: Metaedges**)

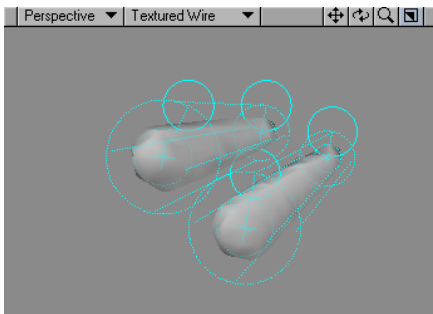
**Step 2:** Metaedges are essentially multi-point versions of plain Metaballs. Your initial clicking point establishes the first point. Hold the **LMB** down and drag out the second point. Other than there being two points, this tool works just like the **Draw Metaballs** tool.



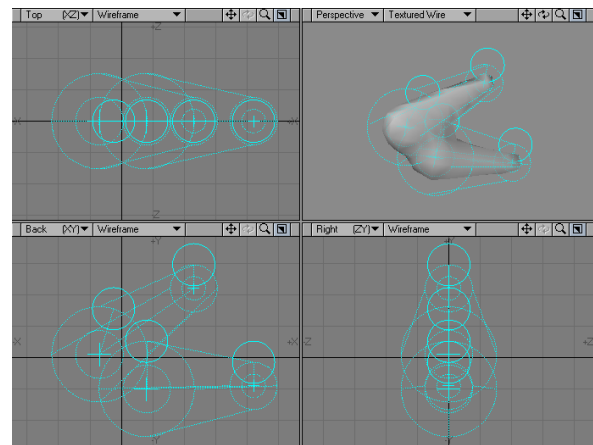
**Step 3:** You can move each end of the Meta Edge object by dragging its center handle. You can change its size by dragging the sizing handle that looks like an orbiting circle.



If you click anywhere else, another Meta-Edges object will be created.



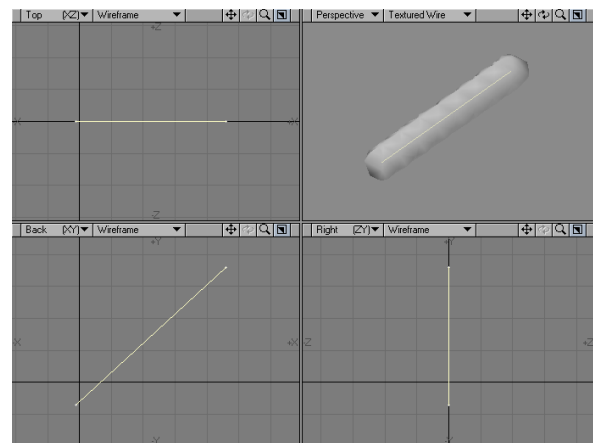
**Step 4:** If you move or increase meta-edges so that its surface approaches the surface of another, they will adhere to each other, like the stuff in a lava lamp.



**Step 5:** Drop the **Draw Meta-Edges** tool (i.e., reselect it or press the space bar).

### Converting to Meta-Edges

Meta-edges can be created from two-point polygons or curves. Simply select the polygon/curve and choose **Construct > Convert: Convert Metaedges**.



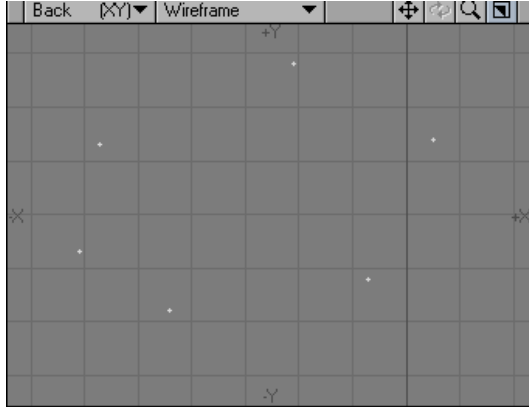
**NOTE:** The third type of **MetaPrimitive** is called a **MetaFace**. There is no tool to draw a MetaFace; it is created from existing geometry and is discussed later on page 262.



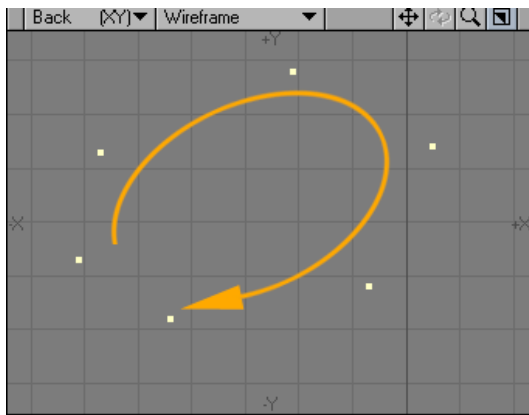
## Make Triangle Fan

You can use the Make Triangle Fan command, which is located in Create > Polygons: More, to create a subdivided polygon from selected points. The steps to create a triangle fan polygon are very similar to creating a polygon with the Make Polygon tool.

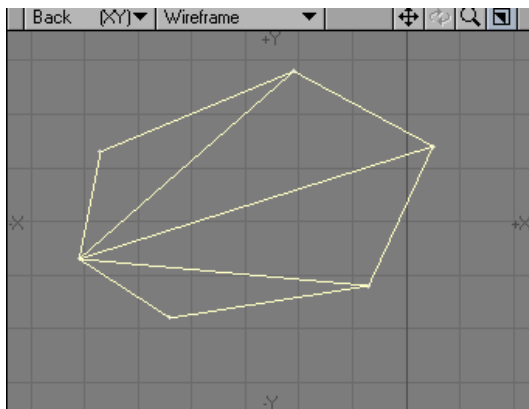
### To create a Triangle Fan from points:



**Step 1:** Create the points that you would like to use on the polygon using the **Point** tool.



**Step 2:** In **Point Selection** mode, select the points that you just created in order.



**Step 3:** Select the **Make Triangle Fan** tool. (Selecting in a clockwise fashion works best.) The triangles will share the first point selected.

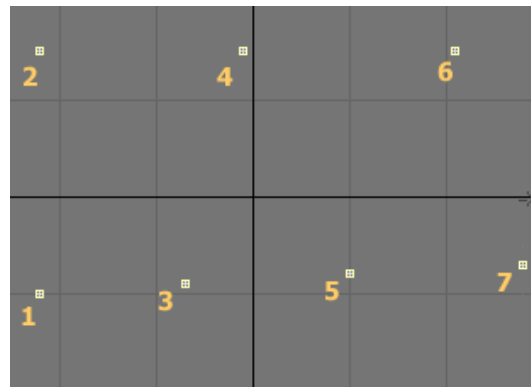
## Make Triangle Strip

The **Make Triangle Strip** tool, which is located in **Create > Polygons: More**, is similar to the **Make Triangle Fan** tool, but creates a strip of triangle polygons from a set of points. This can be useful for game engines that prefer to have geometry built in strips of triangles.

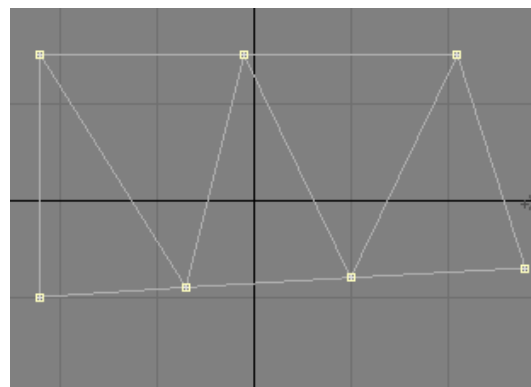
The steps to create a Triangle Strip Polygon are very similar to creating a polygon with the **Make Polygon** tool.

### To create a Triangle Strip Polygon:

**Step 1:** Use the **Points** tool to create the points that you would like to use on the polygon.



**Step 2:** Select the points in the correct order.



**Step 3:** Select Make Triangle Strip.





## Curves

In addition to making polygons from points, you can also create curves. **Curves** are a powerful Modeling tool and help you create smooth edges and flowing organic objects or object details. They are also used for some Modeling tools as a directional path.

You can create curves using a minimal number of points, yet achieve a smoothness that would take many times the number of points if you attempted to mimic the curve with a polygon. Moreover, with fewer points, it is easy to subtly or dramatically change the curve's shape. However, by themselves, curves will never render in a finished image. They are basically a freeform Modeling tool used to create polygonal objects.

There are basically two types of curves: open and closed. An open curve has a beginning and an end — essentially a curving line. A closed curve has no beginning or end; it is a closed loop. A circle is a closed curve.

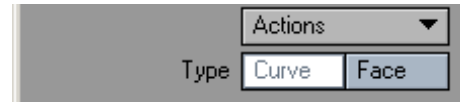
## Sketch Tool

The **Sketch** tool (**Create > Curves: Sketch**) is purely a two-dimensional Modeling tool. With it, you can draw polygons and curves in any viewport.

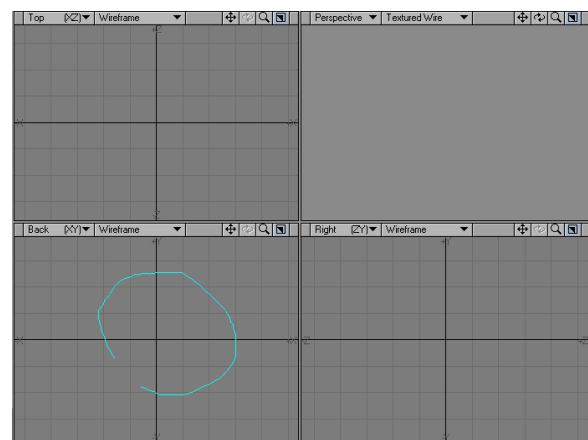
### To sketch a polygon:

**Step 1:** Choose **Create > Curves: Sketch**.

**Step 2:** Open the **Numeric Panel** and select **Face** as the **Type**.

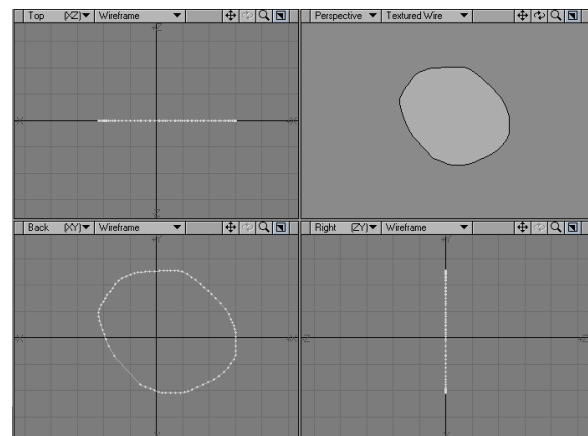


**Step 3:** Drag out the desired shape. (It is not necessary to touch the beginning of the line with the end. The polygon is closed automatically.)



When you release your mouse button, a polygon is created using the shape you sketched out.

The polygon is always centered in the depth dimension of the viewport.



### Sketching Curves

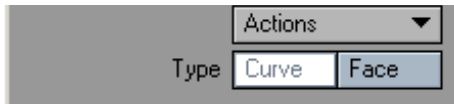
You can use the **Curve** option on the **Numeric Panel** of the **Sketch** tool to create a spline curve instead of a polygon. The tool operates exactly as previously described, except an open spline curve is created instead of a polygon.

### To sketch a curve:

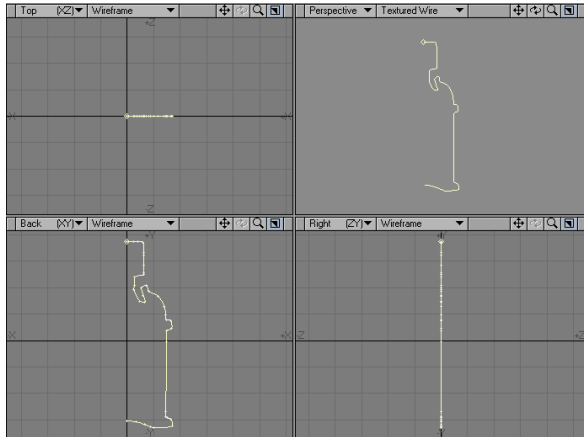
**Step 1:** Click **Create > Curves: Sketch**.



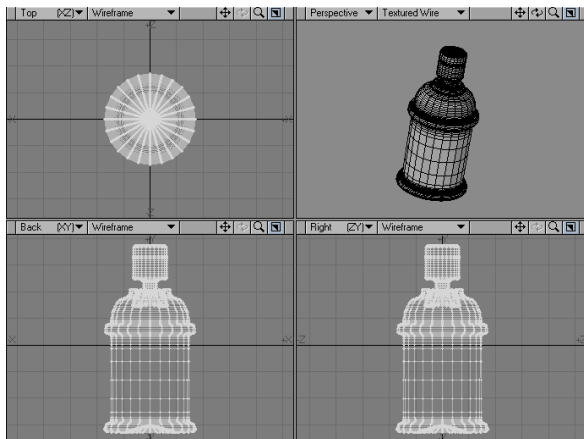
**Step 2:** Open the **Numeric Panel** and select **Curve** as the **Type**.



**Step 3:** Drag out the desired shape. When you release your mouse button, a curve is created using the shape you sketched out. The curve is always written on a plane positioned at 0 of the viewport's perpendicular axis.

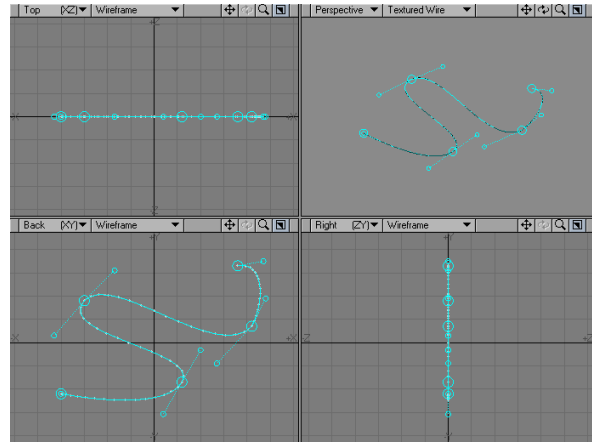


Creating open curves can be used in many Modeling operations. The image below is an example of taking the curve created above and using the **Lathe** tool to create a solid object. Instant Spray Can!



## Bezier Tool

The **Bezier** tool (**Create > Curves: Bezier**) allows you to create standard spline curves using tangent controls. If you've used any vector program you will feel right at home using the **Bezier** tool.



**To use the Bezier tool:**

**Step 1:** Choose **Create > Curves: Bezier**.

**Step 2:** Place your mouse pointer where you would like to begin your curve.

**Step 3:** Click and hold your **LMB** — the beginning and ending of your first curve is created. Drag your mouse to where you want the curve to end and release the mouse button.

**Step 4:** Successive clicks will create additional curves.

**Step 5:** Drag the tangent handles (circles) to adjust the curve shape. Holding the **Ctrl** key while dragging will allow you to move both sides of the tangent simultaneously.

**Step 6:** Click and drag with the **LMB** in the large circles that represent the curve control points to change the control point's position.

**Step 7:** Drop the tool by hitting the **space bar**.

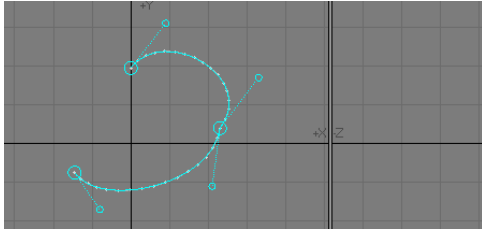
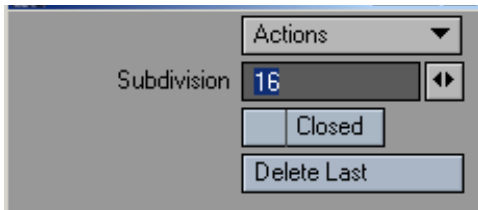


**NOTE:** Using the **Bezier** tool to generate a complex curve actually generates a normal Modeler spline curve between each control point.

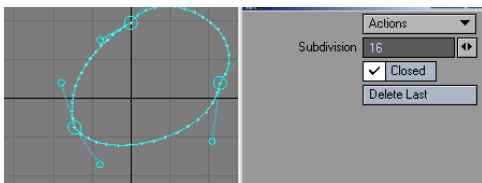
As such, you cannot re-edit the curves using the **Bezier** tool but, you can edit the control points created on the finished splines. To make the curves a single entity, select all the curves (you must be in **Polygon** selection mode to do this) and use the **Detail> Merge Polygons** tool (**Shift Z**) to join all the curves together. (This is essential if the curve is to be used later on with Modelers **Rail Bevel** or **Rail Clone** tool).



On the **Numeric Panel**, you can set the subdivision level (for smoother or less smooth curves), create a closed curve, and delete the last tangent point.



Open Curve

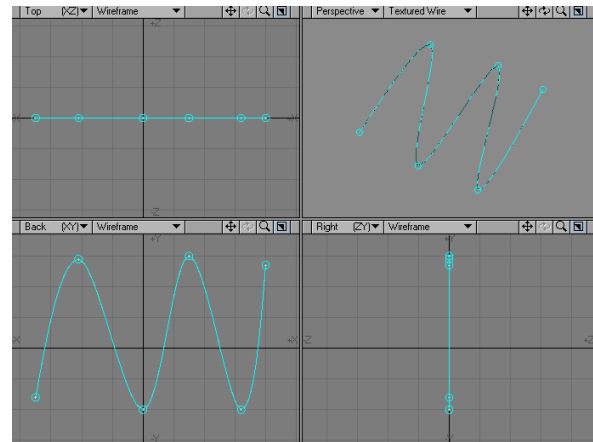


Closed Curve

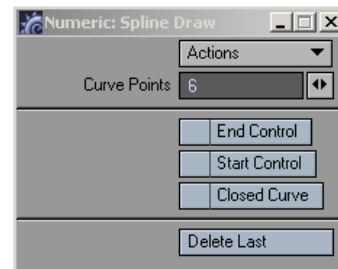
## Spline Draw Tool

The **Spline Draw** tool (**Create > Curves: Spline Draw**) is simply a tool for making spline curves.

To use the tool, just click and move points. Clicking off the curve adds new points to the end.



Its **Numeric Panel** offers options for making control points, discussed later, and deleting the last point from the curve. Changing the **Curve Points** value will resample the curve into more segments. Note that resampling the curve can change its shape, particularly if the number is incremented gradually, as is the case with the mini-slider. If you want a more refined curve that accurately matches the one you have drawn, enter the new number directly into the field to avoid the intermediate curves.

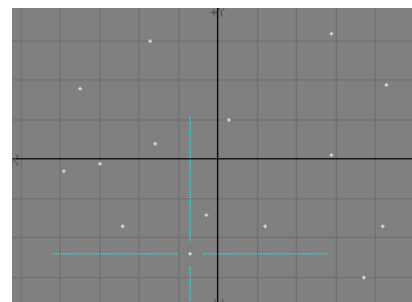


### Creating from Points

You can also create curves from points in a similar manner to creating polygons.

#### To create a curve:

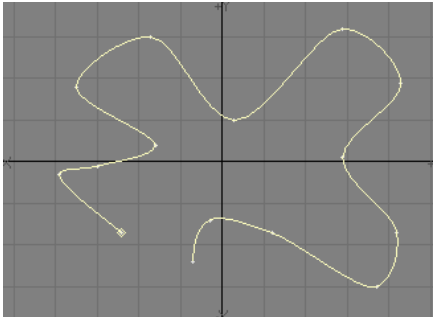
**Step 1:** Create a few points. Clicking with your **RMB** using the **Points** tool (**Create > Points: Points**) will work.



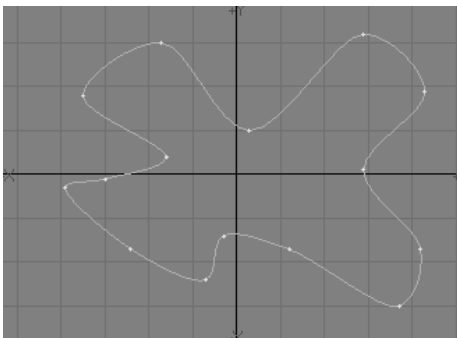


**Step 2:** Your points must be selected in the order you want the curve to flow. However, if you immediately go to — or were using — the **Point Selection** mode, the points should already be in the selected state. If not, select them in the desired order.

**Step 3:** To create an open curve from the points, click **Create > Curves: Make Curve > Make Open Curve** or press **Ctrl P**.



To create a closed curve from the points, click **Create > Curves: Make Curve > Make Closed Curve**.



You can modify the curve's shape by simply moving the individual points. Notice how the curve flows smoothly through the interior points. It is easiest to see this as you drag points on either side of a point. If you need additional control, you can add additional points to an existing curve.

#### **To add points to a curve:**

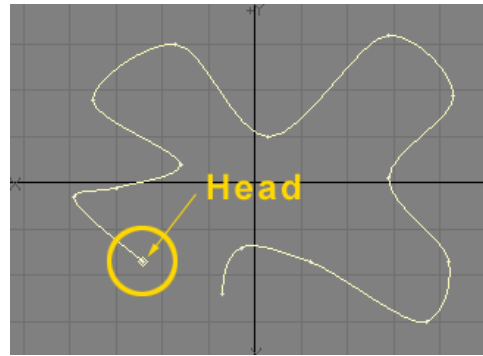
Select the spline or splines, activate the **Add Points** tool (**Multiply > Subdivide: Add Points**) and click anywhere along the curve.

#### **To delete points from a curve:**

Select the points and use the Cut command (**Ctrl X**) or the delete key.

#### **Curve Direction**

Curves also have a head and tail. This is important for certain Modeling operations. When the curve is selected in the **Polygon Selection** mode, the head is indicated by a small diamond; it is the first point you selected when creating the curve.



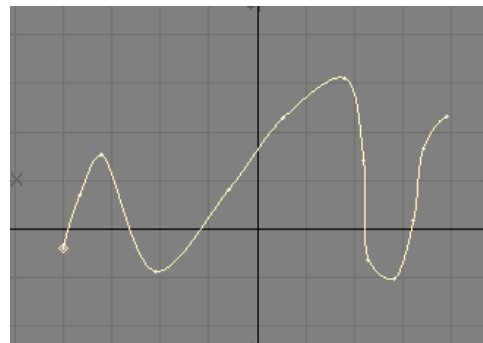
You can flip the head and tail around by using the **Flip** command (**Detail > Polygons: Flip**).

#### **Using Control Points**

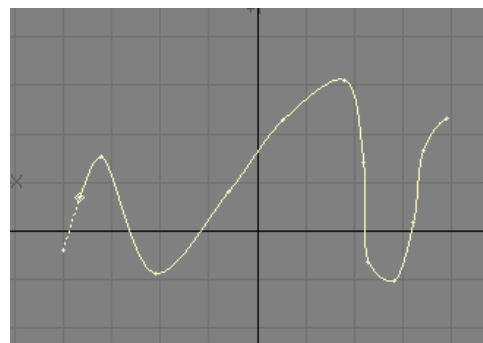
When using a curve as a Modeling tool, sometimes you need to adjust the curve to get it to "that perfect angle." Because of the way that points control an open curve's shape, it can be difficult to get the desired result at the ends of a curve. Fortunately, Modeler lets you have extra "control" points at each end, which affect the curve's shape, but don't actually become part of the curve's Modeling characteristics.

#### **To activate control points on a curve:**

**Step 1:** If you have more than one curve, select your desired curve first using the **Polygon Selection** mode.

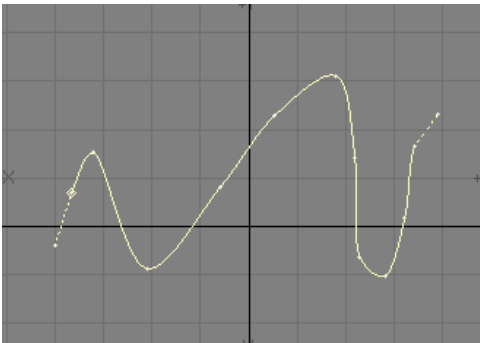


**Step 2:** Choose **Detail > Curves: Control Points > Begin Control Point** to detach the first point from the curve. It is your starting control point. Dragging this point will change the shape of the curve at the new starting point.





Choose **Detail > Curves: Control Points > End Control Point** to detach the last point from the curve. It is your ending control point. Dragging it will change the shape of the curve at the new ending point.



The **Control Points** commands will also toggle the control point state off, if selected again.



NOTE: It is perfectly legal to have only a beginning or ending control point on a curve.



HINT: If you plan to use control points on a curve, you should add an extra point at the beginning and end of a curve for that purpose.



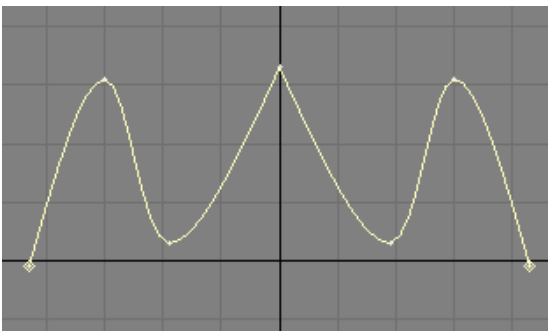
HINT: If you require a sharper corner anywhere on a curve, simply add an extra point close to the corner using **Construct > Subdivide : Add Points**.

## Smoothing Two Overlapping Curves

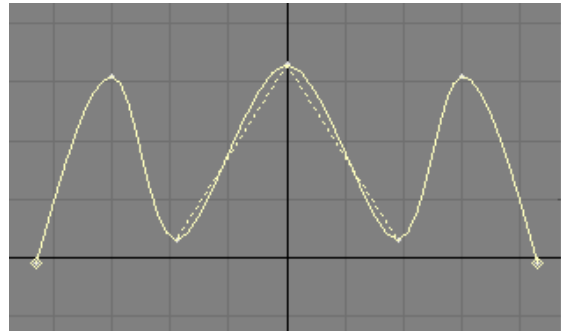
When two curves share the same end point, because of point merging or some other reason, the curve may not flow smoothly through that point. This can be particularly troubling when designing a complex spline cage. You could kill the curve (**K**), reselect the points and create a new curve, but there is an easier way.

### To smooth curves sharing a common end point:

**Step 1:** Select the two curves (**Polygon Selection** mode).



**Step 2:** Click **Detail > Curves: Smooth**. This smooths the joint as though the two curves were originally plotted as one continuous curve—even though they remain individual curves.



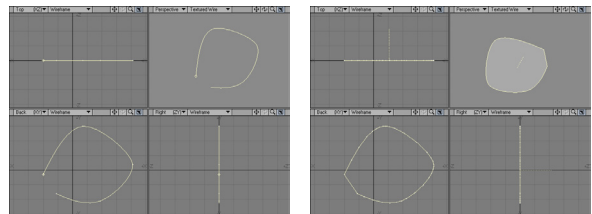
NOTE: Make sure that the point they share is in fact one point. (If not, merge or weld the points into one.)

## Turning Curves into Polygons

Since curves will never be seen by themselves, sometimes you need to turn them into polygons.

### To convert a curve into a polygon:

Select the curve(s) and choose **Construct > Convert: Freeze**. Each independent curve will become a distinct polygon. An open curve will be closed (between its end points) before it is converted to a polygon.

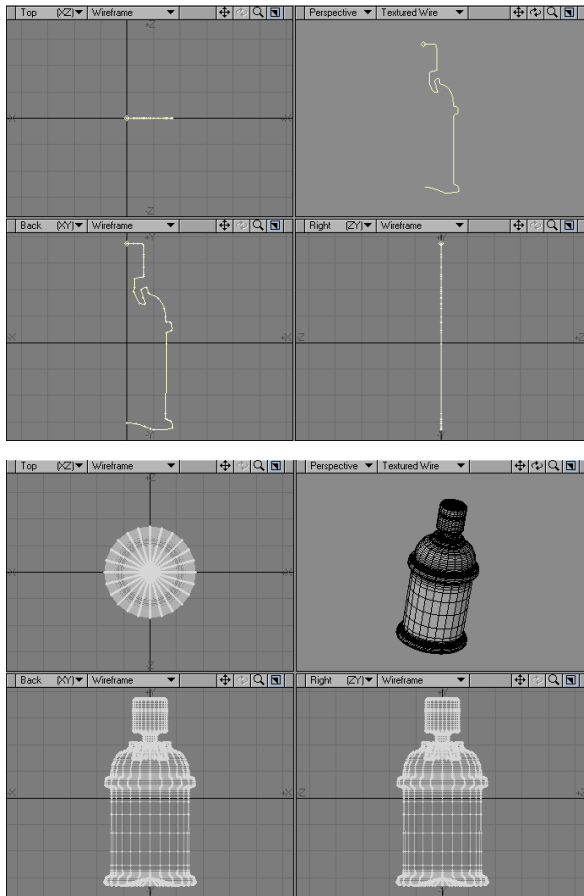


The number of points that Modeler will use to approximate the curve depends on the setting for **Curve Division** on the **General Options Panel** (**Modeler > Edit > General Options**).



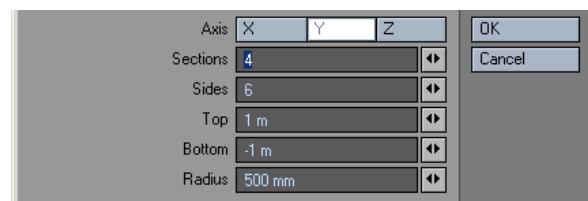
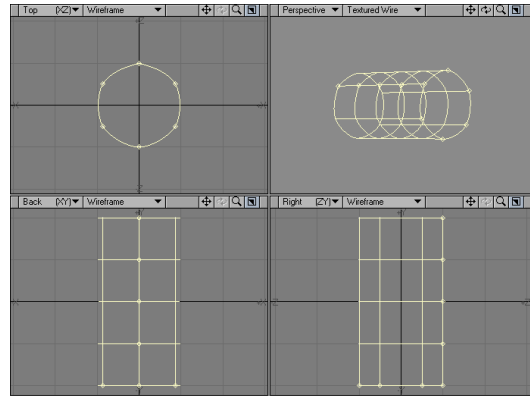
## Using Modeling Tools on Curves

Most of the standard Modeling tools will work on a curve as well. However, a curve is sometimes converted into a polygon before the tool is applied. This happens with the **Extrude** and **Lathe** tools (**Multiply** menu), for example.



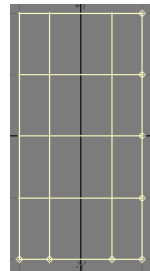
## Spline Cage

**Create > Curves: Spline Cage** creates a cylindrical spline cage (connected curves).

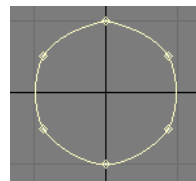


**Axis** — The primitive is aligned with this axis.

**Sections** — Sets how many vertical segments should be used.



**Sides** — Determines how many segments should be used around the perimeter.



**Bottom** — Bottom is the starting point of the primitive along the selected Axis.

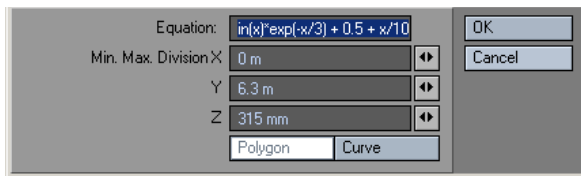
**Top** — Top is the ending point of the primitive along the selected Axis.

**Radius** — Refers to the radius of the primitive along the axes.

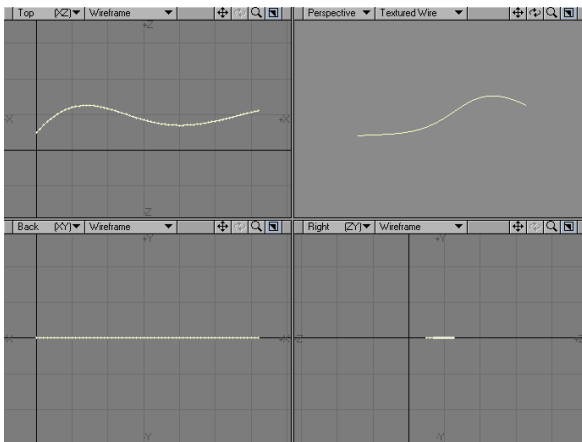


## Plot 1D Tool

The **Plot1D** tool, which is located in **Create > Curves: Plot1D**, generates either a curve consisting of two-point polygons or a spline curve. The curve is generated along the X axis, and its height is in the Z axis. Get your maths head on for this tool.



The height is determined by the value of the expression in the **Equation** field at each x. In the fields **Min**, **Max**, and **Division**, enter the x value for the start of the curve, the x value for the end of the curve, and the number of divisions of the curve.









---

## Chapter 4: Modify Tab

---



## Modify Tab

The Modify Tab contains a variety of tools to make changes to existing geometry. These can be applied to individual selected points, edges, or polygons, portions of an object, or to an entire object. The available operations range from simple relocation of points or polygons to resizing the geometry in one or more dimensions, to complex deformations of the entire mesh.

### Falloff Mode

Several of the tools on the **Modify** tab have a **Falloff** pop-up menu on the lower half of the numeric panels. By changing the **Falloff** setting, you can effectively make the function of the **Move** tool have the same effects as using the **Shear**, **Magnet**, **DragNet**, and **Drag** tools, by just changing its Falloff setting. You might think of these tools as just shortcuts to a defined group of **Modify** settings. This type of selection via falloffs is also referred to as a *Soft Selection*.

We will be explaining the falloff tools here at the start of the chapter.

### Falloff Types

None — **None** means no falloff. The **Move** tool is set to this by default.

Linear — **Linear** makes the effect fall off along a linear axis, and operates like the tools **Shear**, **Twist**, **Taper1**, and **Taper2**.

Point — **Point** falloff works like the **Drag** tool.

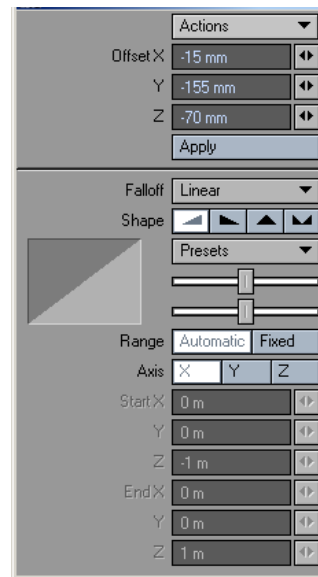
Point Radial — **Point Radial** operates much like the **DragNet** tool.

Polygon — **Polygon** means falloff is immediate around a polygon, so you can move only a single polygon. It behaves like the **drag** tool but works on single polygons instead of points.

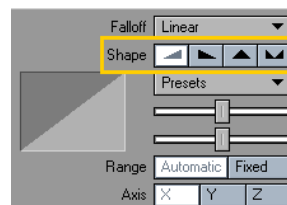
Radial — **Radial** means the effect will fall off in a radial pattern (in a cylindrical or spherical shape), so it is the default for **Magnet**, **Vortex**, **Pole1**, and **Pole2**.

Weight Map — **Weight Map** uses the selected **Weight Map** for the falloff. *Unweighted* points will not move at all, while heavily weighted points will move the most. Negatively weighted points will move inversely. This allows you to use an irregularly shaped falloff. This mode is particularly useful for creating facial endomorphs.

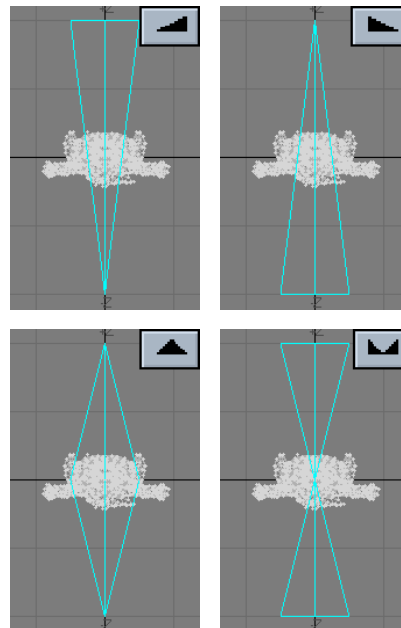
### Linear Settings:



### Shape



You can quickly select between various basic falloff directions using the **Shape** buttons. The **Shape** setting determines the **Area of Influence**. This is shown graphically in both the numeric panel and viewport windows. You can also use the **Left** and **Right** cursor keys to change the selected **Shape**.



### Tension Sliders



You can adjust how the effect falls off using the two sliders, which act like tension spline controls for the beginning (upper slider) and ending (lower slider).

## Presets

You can quickly set up some common curves by selecting from the **Preset** pop-up menu. The graph on the numeric panel, as well as the outer edges of the axis tree (when using a fixed range) gives you a visual picture of the falloff. You can also use the **Up** and **Down cursor keys** to cycle through the **Preset** selection.

## Range

The Range option determines where the Falloff's center is located. By default, a linear falloff tool's effect is automatically applied to the object (or selected items) 100 percent at one end and zero at the other, along the axis perpendicular to the editing viewport. This is indicated by the **Range** option at the bottom portion of the numeric panel, which has **Automatic** selected.

To use the tool in the **Automatic** mode via the numeric panel (by clicking the **Apply** button), select the perpendicular **Axis**. (If the axis has not been set either manually or by a previous mouse-based modify operation, the **Axis** setting is used to compute the falloff when you click **Apply**.)

Automatic — The center is set by clicking in the viewport.

Fixed — The center is set by entering in values in the numeric window.

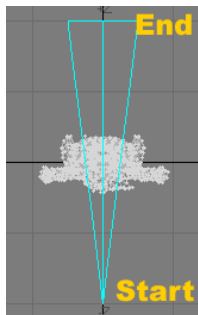


**HINT:** Using the **Automatic Range** is the simplest method. Thus, if possible, model your object along one of the three axes.

Instead of using an **Automatic Range**, you can define a specific **Fixed Range** using an *axis tree*. This type of **Range** can be placed at any position and at any angle in 3D space. The effect tree looks like a pair of crossed wedges. The tool's effect is applied along this tree with zero at its tip and 100 percent at its base (fat end) — portions of the object beyond the base are still affected 100 percent. The edges of the tree reflect the fall-off curve shape.

## To create a fixed range:

**Step 1:** Select the desired linear falloff tool and drag out the axis tree with your **RMB**. (The **Range** option will be set to **Fixed** on the numeric panel.)



**Step 2:** Reposition the ends by dragging them with your **RMB**.



**NOTE:** The XYZ positions of the starting and ending points are reflected in the numeric panel. You can edit these values, if necessary.

## To reset to Automatic Range:

To reset to **Automatic Range**, click your mouse pointer on an inactive interface area. Note this will also switch you back to the **Automatic Range** mode.

## Moving an Object on an Arbitrary Axis

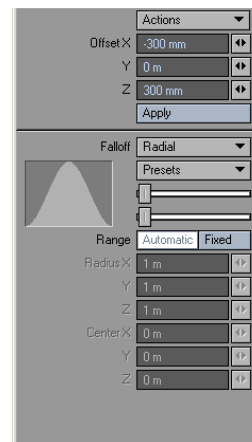
Let's say you had an object that was not aligned nicely along the X, Y, or Z axis. If you wanted, you could define a fixed range and align the effect tree with the object. However, this can be a tedious process. A better alternative would be to use a Perspective viewport.

Remember, editing generally occurs along a viewport's arbitrary horizontal and vertical axes. When using a Perspective viewport, this can be something other than directly along the X, Y, or Z axis. Thus, you can align your object so that it is perpendicular in the viewport and just use the **Automatic Range**.

## Radial Settings:

This influence area falls off in a radial pattern (a sphere or cylinder). It is set independently of any selected polygons or points, but works in conjunction with them. That is, if points/polygons are selected, only those within the influence area are affected.

By default, the tool's effect is automatically applied to the entire object (or selected items). This is indicated by the **Range** option, at the bottom portion of the numeric panel, which will have **Automatic** selected.



## Setting the Falloff Shape

How the effect falls off is determined by the two sliders, which act like tension spline controls for the beginning/end (upper slider) and center (lower slider). You can quickly set up some common curves by selecting from the **Preset** pop-up menu. The graph gives you a visual picture of the falloff.

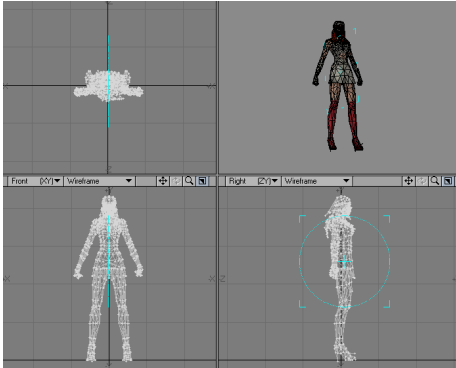
## Defining a Specific Range

Instead of using an **Automatic Range**, you can define a specific **Fixed Range**, which can be positioned and sized anywhere in 3D space.

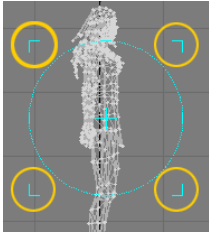


### To set the influence area:

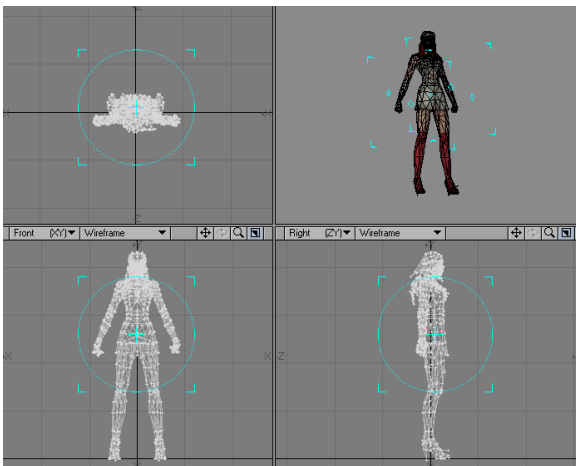
**Step 1:** Select the desired **Radial** falloff tool and drag out the influence area tree with your **RMB**. (The **Range** option will be set to **Fixed** on the numeric panel.) At this point, you have defined a cylindrical area that extends infinitely along an axis perpendicular to the viewport. Use the **Ctrl** key to constrain to a circle.



**Step 2:** Resize the initial area as needed by dragging the outline edge handles with your **RMB** or reposition by dragging the center handle.



**Step 3:** To limit the depth of the area, drag out the outline in a different viewport. The amount of influence is strongest at the center and diminishes towards the edges. Polygons with all points outside of the area will not be affected.



**NOTE:** Creating the radial falloff influence area is similar to creating a Box primitive, particularly in a perspective viewport.

### To reset the influence area:

To reset the influence area, click your mouse pointer on an inactive interface area. Note this will also switch you back to the **Automatic Range** mode.

### The Numeric Panel

The **Radius X**, **Y**, and **Z** values reflect the size and shape of the influence area. If you have defined a cylindrical (i.e., two-dimensional) area, one axis will be 0. The **Center X**, **Y**, and **Z** values define the center of the area of influence.



## Translate

The tools on the Translate section allow you to relocate all or part of the geometry, in some cases with varying level of effect across the geometry.

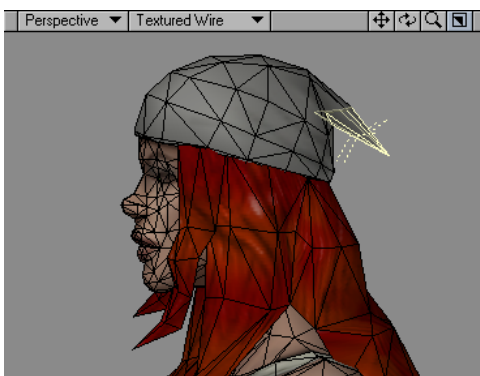
### Move Tool

(default keyboard shortcut **T**)

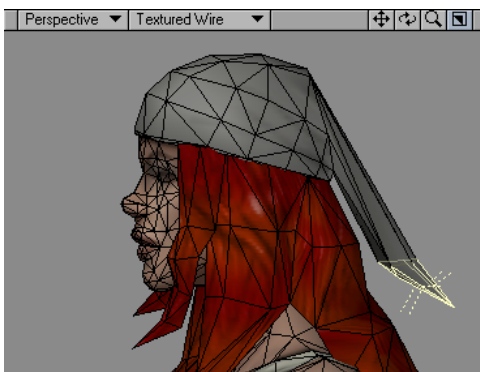
The **Move** tool Move Tool (**Modify > Move**) is the basic tool used to reposition geometry. This tool is used often so it's worth learning the shortcut key.

#### To move geometry:

**Step 1:** Select your points or polygons.



**Step 2:** Choose **Modify > Move (T)**, click and drag your mouse in any view. The selected geometry will follow your movement.



NOTE: Holding the **Ctrl** key before you click and drag will constrain the movement along one of the axes. It is also not necessary to place the cursor directly on the selected item. You can move from anywhere.



NOTE: The falloff modes for this tool are described in detail starting on page 136.

### Drag Tool

(default keyboard shortcut **Ctrl T**)

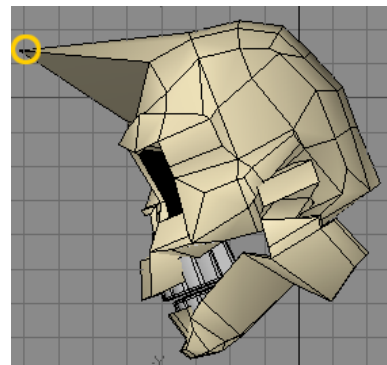
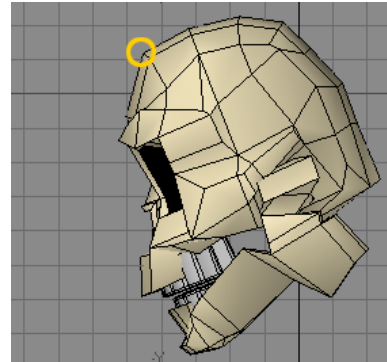
To move points around without having to select them, select Move, then select other points, use the Drag tool.

#### To drag points:

**Step 1:** To limit the edit to certain points, select them first. The **Drag** tool will not affect unselected points unless none are selected. Thus, if you have points in-line with each other in a viewport and don't want to drag all of them, select the target points first. (Note that all of the selected points do not move as you drag, only the ones under your mouse pointer.)

**Step 2:** Choose **Modify > Drag** to access the **Drag** tool.

**Step 3:** Position your mouse pointer directly over a point, click and drag.



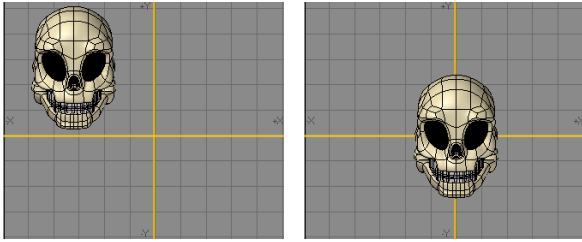
NOTE: The falloff modes for this tool are described in detail starting on page 136.



## Center Data

(default keyboard shortcut **F2**)

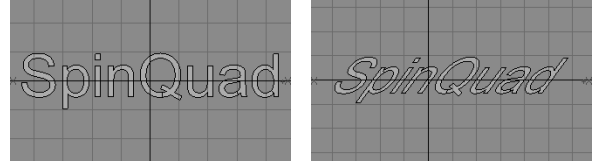
The Center Data command (**Modify: Center**) centers the contents of the foreground layer(s) on the Origin (i.e., 0, 0, 0).



## Shear Tool

(default keyboard shortcut **J**)

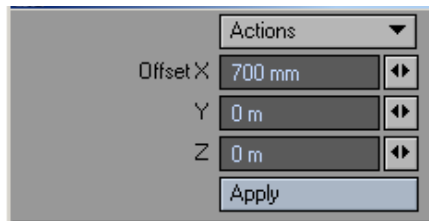
Shearing is making an object slant to one side. A good example is italicized text. Think of **Shear** as **Move** with axial falloff. That is, the strength of the moving influence is not equal across the object, but gradually falls off along a selected axis.



### To use the Shear tool:

Select **Modify > Shear**, click and drag your **LMB** in a viewport in the direction you want the slant.

The information display tells you how much **Offset** you applied along two of the three axes, depending on the view you edit in. Hold the **Ctrl** key down while dragging to constrain the shear to your initial dragging axis.



The **Offset X**, **Y**, and **Z** values on the numeric panel will reflect the amount of movement along those axes for the tool's last operation.



NOTE: The falloff modes for this tool are described in detail starting on page 136.

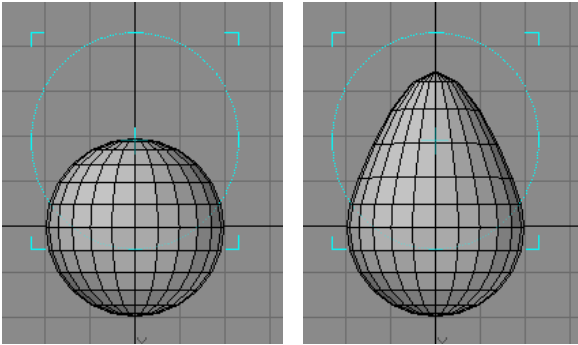




## Magnet Tool

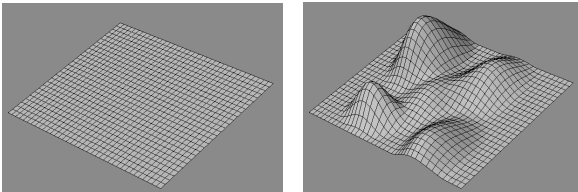
(default keyboard shortcut :)

The **Magnet** tool (**Modify > Magnet**) is a nifty tool for pushing in or pulling out sections of objects. The effect is smoothly applied, so it tends to create soft-edged bulges or dips in a surface. You might use the **Magnet** tool to create an egg shape from a sphere. (Simply enclose the upper hemisphere in the influence range, center the area near the top of the sphere, and drag upward with the LMB.) You can actually use the tool immediately as sort of a freeform deformation or you can create an area of influence.



### To use the Magnet tool:

Select **Modify > Magnet** and drag your **LMB** in a viewport. The initial pointer location defines the center of the move, although this may not be obvious if it is located outside of a defined range.



The information display will indicate the **Offset** amount for the affected axes, that is, the distance you have moved your mouse.



The **Offset X**, **Y**, and **Z** values on the numeric panel will reflect the amount of movement along those axes for the tool's last operation. You can edit the values and then click the **Apply** button to apply them, which you can do multiple times.



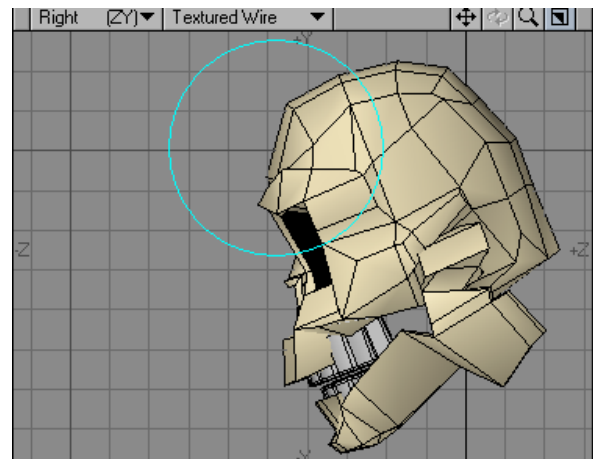
NOTE: The falloff modes for this tool are described in detail starting on page 136.

## DragNet Tool

(default keyboard shortcut ;)

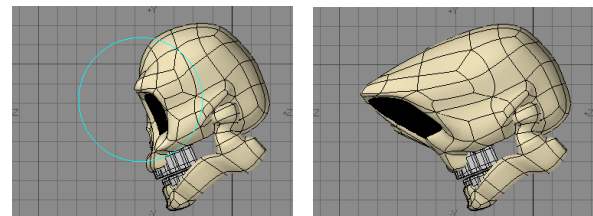
The **DragNet** tool (**Modify > DragNet**) combines the **Drag** and **Magnet** tools. While you might use it somewhat interchangeably with **Magnet**, they operate quite differently. **Magnet's** radial falloff influence area works independently from the actual use of the tool. With **DragNet**, the spherical range is always centered around the initial point of dragging. Moreover, you must actually drag a point, although other points in the influence area will also move.

The influence area is defined by the **Radius** around the initial pointer position. You can graphically set this by dragging out a circle with your **RMB**.



### To use the DragNet tool:

Select **Modify > DragNet**, click and drag a point. Other points in the influence area will also move, though not as much.



The information display tells you the **Offset** value, that is, how far you dragged the mouse. If you set the influence area interactively with the **RMB**, the information display shows the **Radius** value.



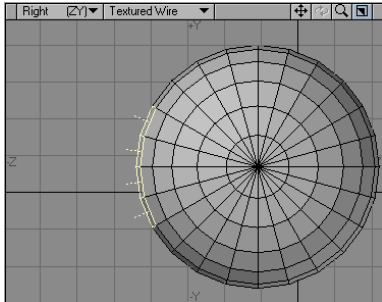
NOTE: The falloff modes for this tool are described in detail starting on page 136.



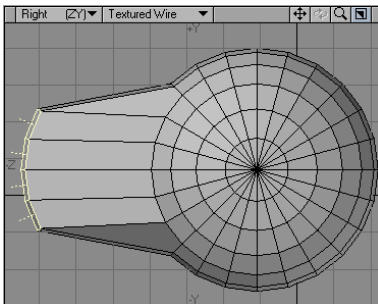
## Move Plus

The **Move Plus** tool (**Modify > Move Plus**) is a detailed version of the **Move** tool used to reposition geometry.

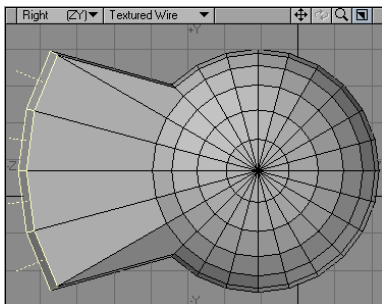
This tool is almost identical to LightWave's interactive **Move** tool. However, it provides two additional features. When dragging your selection with the right mouse button, the selection will be translated along the local normals. When dragging with the right mouse button with shift held down, the selection will be translated along the averaged normal.



Before Move Plus



Move Plus LMB



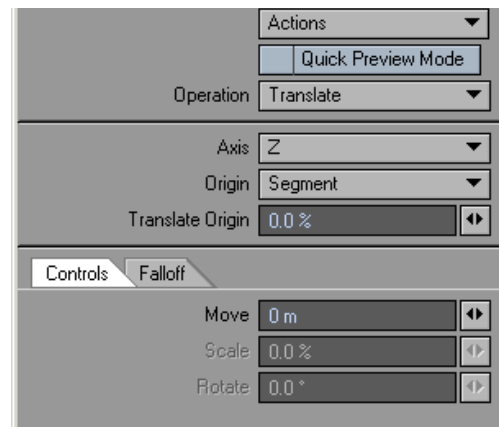
Move Plus RMB



**NOTE:** **Move Plus** doesn't replace the **Move** tool as it doesn't have the **Falloff** settings that are found in the **Move** tool.

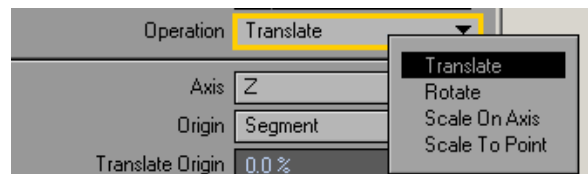
## Translate Plus Tool

**Translate Plus** is found under the **Modify Tab** and functions as a detailed **Move**, **Rotate**, and **Scale** tool. You can use a point's normal, a segment, or a segment's normal to define how to move, rotate, scale a selection. You can even just use global X, Y, or Z. Each of these operations can have a falloff.

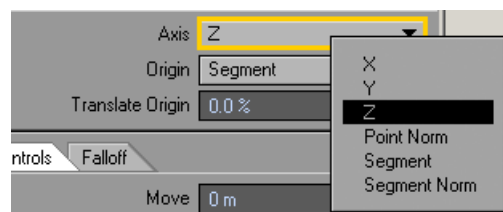


**Quick Preview Mode** — The **Quick Preview Mode** will show a temporary blue outline of the translation and the action won't take place until you have applied (RMB) the action. This is particularly handy when working with very heavy meshes.

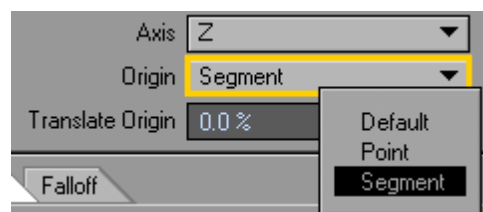
**Operation** — This setting defines what will be performed on the object. Options are: Translate, Rotate, Scale on Axis, Scale to Point.



**Axis** — What axis will be used to determine how selection is "Operated" on. **X**, **Y**, and **Z** are just the default axes. The other options are **Point Normal**, **Segment**, and **Segment Normal**.



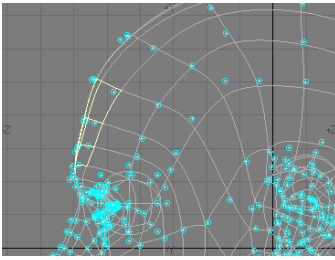
**Origin** — Refers to where the **Axis** originates. By default it is placed at center of whatever the **Axis** is. There are times when it needs to be somewhere else. Generally, there won't be a need to change this unless the operation is **Scale on Axis**. The options are **Default**, **Point**, or **Segment**.



When **Point** and **Segment** are selected, the object will display



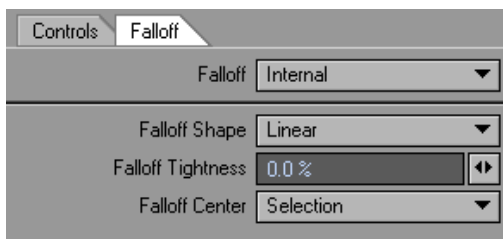
blue circles over each point. Select one point for **Point** mode and select two points for **Segment** mode.



**Translate Origin** — Enables moving of the **Origin**. For example, if **Point Normal** was the axis chosen, it will move the **Origin** along the normal.

**Controls Tab** — Whichever **Operation** is selected will be editable and the others will be greyed out. Generally, there is no need to use the controls because it is easier to click on the screen itself to move selections.

**Falloff Tab** — The options for falloff are **Internal** and **None**. **Internal** means that the falloff is only within the selection. **Falloff** settings can be edited by using the options at the bottom of the panel.



**Falloff Shape** — Defaults to **Linear**, but there are also **Bell**, **Curve Down**, and **Curve Up**. The shape can be modified further by using **Falloff Tightness**.

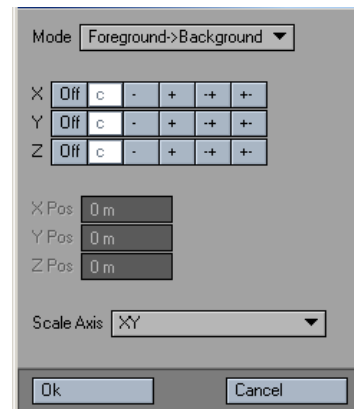
**Falloff Tightness** — Determines whether points in falloff range will be closer to selection (or origin) or closer to their original position.

**Falloff Center** — Same as the **Origin** settings except instead of **Default** it is a **Selection**.

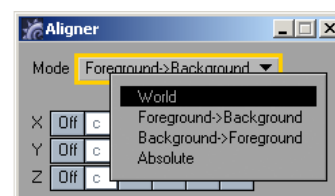
Even though holding **Ctrl** or using the **MMB** will constrain movement, sometimes that is hard to do in perspective. Also, if moving selections with LightWave's **Move** tool is slow (due to large poly count), this tool is handy because it moves in real time.

## Aligner Tool

**Aligner** will align geometry not only with axes, but also other geometry. The **Mode** setting determines what type of alignment you want.



### Modes



**World** — The **World** mode will align geometry in the foreground to the X, Y and Z axes. The **c** setting will center on the axis. The **minus (-)** or **plus (+)** settings will place the geometry on the negative or positive side of the axis. **Off** disables that axis. So if you had a "floor" at Y = 0 and you wanted the object centered, but resting on this floor, set the Z and X axes to c and the Y axis to +.

**Foreground->Background** — In the **F.G.->B.G.** mode, the foreground is aligned to the background. This mode uses a bounding box around all of the geometry in the reference layer (the background). You can center or position on the negative or positive side of the reference. The **-+** setting will align the right edge of the geometry with the right edge of the reference layer. The **+-** setting will align the left edge of the geometry with the right edge of the reference layer. You may also scale the geometry to the reference layer. Select the axis or axes to scale using the **Scale Axis** pop-up menu.

**Background->Foreground** — The **B.G.->F.G.** mode works just like **F.G.->B.G.**, but uses the foreground as the reference for the background.

**Absolute** — The **Absolute** mode works like the **World** mode except you can numerically define where the center is. In other words, you can make it something other than 0, 0, 0.

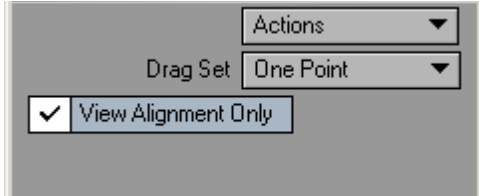
**Scale Axis** — Gives the option to scale the geometry to the reference layer. Select the axis or axes in this pop-up menu.



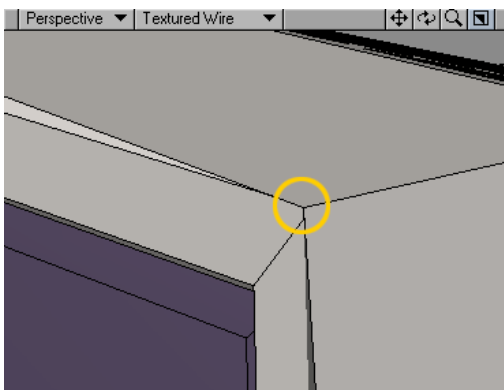
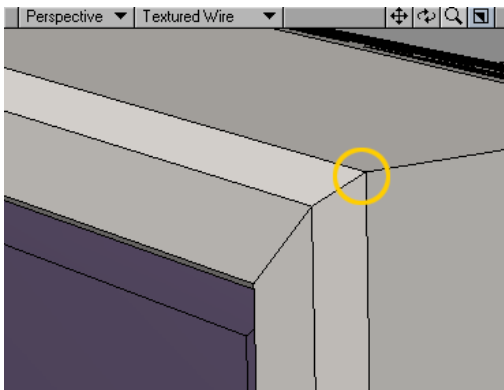
## Snap Drag Tool

(default keyboard shortcut **Shift** **G**)

The **Snap Drag** tool (**Modify > Snap Drag Tool**) allows points to be dragged and snapped to align with other points. When you drag a point with the **Snap Drag** tool, the point is snapped to the grid or to one of the non-moving points, if any.



If you set **Drag Set** to **One Point** on the numeric panel, only the single point will move. If you choose **Connected Points**, points that are connected to the point being dragged will also move. If you choose **All Points**, all points will move. As with other tools, you can use selection to limit which points are affected. **Snap** makes it easy to manipulate *unconnected* parts without having to make explicit selections.



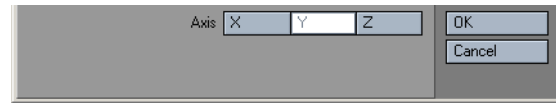
With **View Alignment**, points will be *snapped* to approximately align with the other points along an axis perpendicular to the viewport, but not fully in three dimensions.



NOTE: Although this tool works on individual points, selecting polygons and working in the **Polygon Edit** mode will limit its effect to points that share a vertex with the selected polygons.

## Center 1D

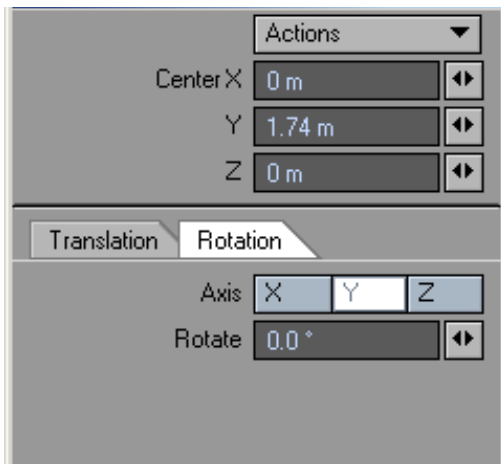
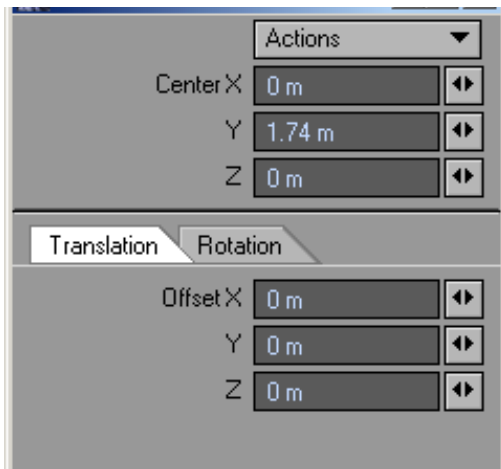
The **Center1D** command (**Modify > Center1D**) is similar to **Center**, except a requester will appear that lets you center along a single axis. The center of the object is placed at the 0 position for the selected **Axis**.



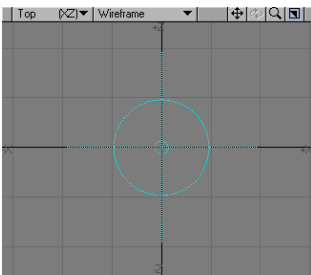


## Rove Tool

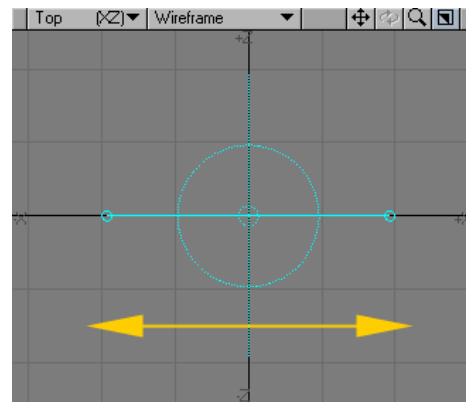
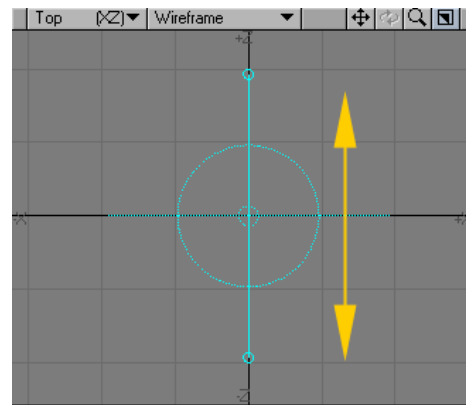
The **Rove** tool (**Modify > Rove**) allows you to move and rotate with a single tool!



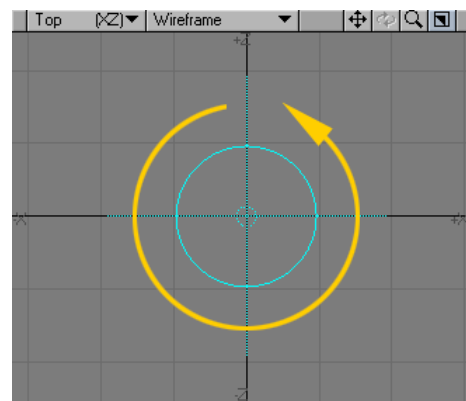
**Position Rove Tool** — Click and drag off the tool's widget to place it where you want the action center to be.



**Translation** — To move an object, simply click and drag. Dragging on an axis constrains movement to that axis.



**Rotation** — Rotate the object about the center of the tool by clicking and dragging the rotation ring. The center of rotation can be repositioned by dragging the center of the tool.

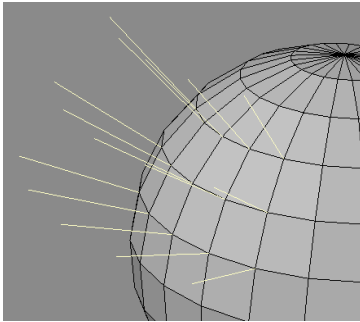


**NOTE:** The **Rove** tool only works in one Layer.

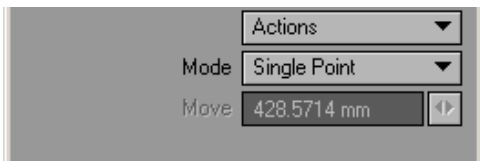


## Point Normal Move

**Point Normal Move** is located under the **Modify Tab** and allows you to move points along their normals.



The yellow lines in the image above represent the normals for some of the points on the ball object. Using **Point Normal Move** would move each point along its normals.



### Mode:

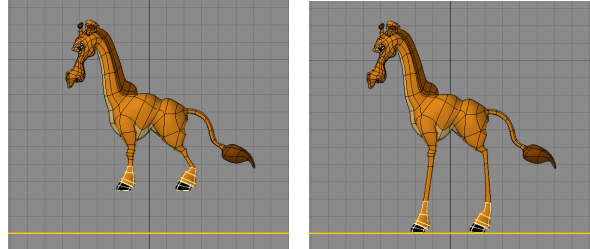
**Single Point** — allows you to run it without selecting points first and then you can move any points you want without selecting them.

**Selected Points** — select the points you want to move, select **Point Normal Move** tool and then click and drag anywhere on the screen to move them in and out.

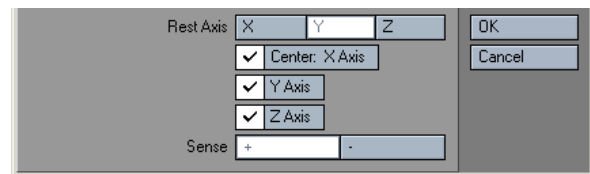
## Rest On Ground

(default keyboard shortcut **F3**)

The **Rest On Ground** command (**Modify > Rest On Ground**) will move selected polygons so that they rest on a defined ground plane, that is, where X, Y, or Z equals 0.



The **Rest Axis** is the axis perpendicular to the plane you want the object to rest on. You may also center the object on any of the axes.



The **Sense** option rests the object on the positive (+) side of the plane or the negative (-) side. The default setting would be ideal for, say, centring a vehicle around the Origin and placing it so it sits flat on the Y axis plane.

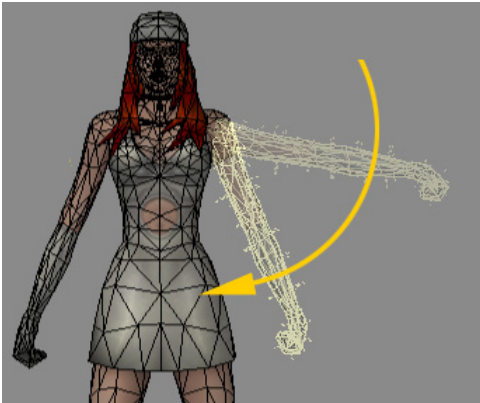


## Rotate

### Rotate Tool

(default keyboard shortcut **Y**)

Select **Modify > Rotate**, click and drag your **LMB** left for counter-clockwise rotation or right for clockwise rotation. The center of the effect is determined by the **Action Center** setting. Hold the **Ctrl** key down as you drag to rotate in 15-degree increments. The information display (lower-left corner of the interface) will tell you the angle of rotation, that is, the number of degrees you have rotated.



Rotating an entire object in Modeler is very similar to as doing it in Layout, except that you rotate around the perpendicular axis of the viewport. Because of that, you will often have to use various viewports to rotate an object.

#### The Rotate Numeric Panel

The **Angle** defines the degree of rotation. The **Axis** setting defines the axis of rotation. When you click in a viewport, the axis is an imaginary line running perpendicular to the view you are editing in (i.e., coming straight out towards you). For the **Back** view, it would be the Z axis; in the **Top** view, it would be the Y axis; and in the **Right** view, it would be the X axis. If arbitrary axes were used, no **Axis** will be selected.

The **Center XYZ** values define the center of rotation coordinates.



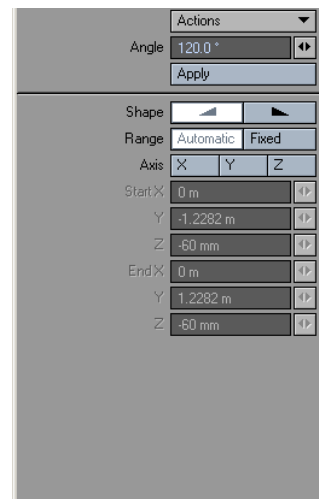
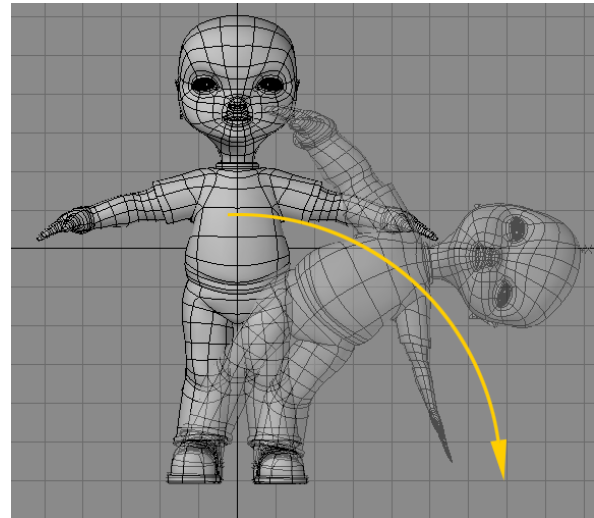
**HINT:** In any viewport, pressing the **R** key (**Rotate 90 Right**) will rotate an object clockwise 90 degrees around a perpendicular axis located at the current mouse pointer location. You can also map the **Rotate 90 Left** command to a hot key as well.



**NOTE:** The falloff modes for this tool are described in detail starting on page 136.

## Bend Tool

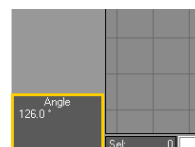
The **Bend** tool (**Modify > Bend**) is a combination of **Rotate** and **Shear**. It will move one side of an object and rotate it at the same time, causing the entire object to bend.



#### To use the Bend tool:

Select the tool and drag your **LMB** in the direction you want the bend to occur. The initial pointer location defines the center of the rotation. The trick is to use the **Bend** tool in the viewport that is perpendicular to the axis of the bend. Let's say you have a tube whose main axis is along Y. Use the **Bend** tool from the top or bottom view. Also, experiment with the placement of the cursor relative to the object. Activating **Bend** when the cursor is centered over the object has a different effect than activating **Bend** with the cursor located away from the object.

The information display tells you the **Angle** of bending, that is, the number of degrees you bent the object. Hold the **Ctrl** key while dragging to constrain movement to increments of 15 degrees—this makes it much easier to bend along a single axis.



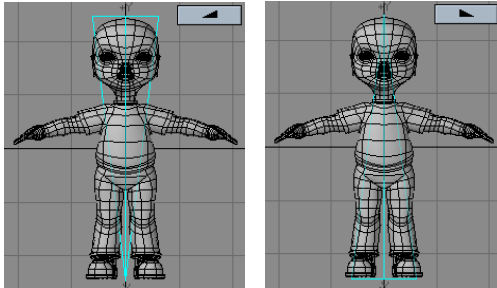




**Angle** — The **Angle** value on the numeric panel will reflect the degrees of rotation.

**Shape** — You can quickly select between two basic falloff directions using the **Shape** buttons.

The **Shape** setting determines the **Area of Influence**. This is shown graphically in both the numeric panel and viewport windows. You can also use the Left and Right cursor keys to change the selected **Shape**.



### Range

The **Range** option determines where the Falloff's center is located. By default, a linear falloff tool's effect is automatically applied to the object (or selected items), 100 percent at one end and zero at the other, along the axis perpendicular to the editing viewport. This is indicated by the **Range** option at the bottom portion of the numeric panel, which has **Automatic** selected.

To use the tool in the **Automatic Mode** via the numeric panel (by clicking the **Apply** button), select the perpendicular **Axis**. (If the axis has not been set either manually or by a previous mouse-based modify operation, the **Axis** setting is used to compute the falloff when you click **Apply**.)

**Automatic** — The center is set by clicking in the viewport.

**Fixed** — The center is set by entering in values in the numeric window.

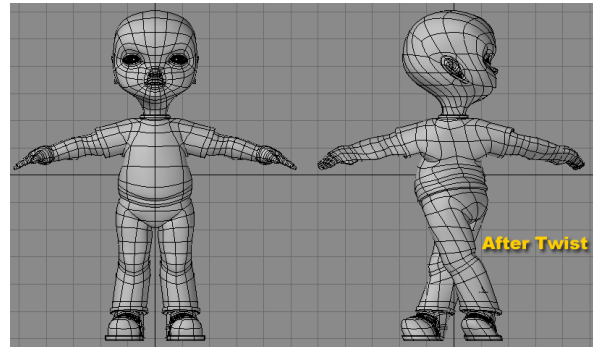


**HINT:** Using the **Automatic Range** is the simplest method. Thus, if possible, model your object along one of the three axes.

Instead of using an **Automatic Range**, you can define a specific **Fixed Range** using an *axis tree*. This type of **Range** can be placed at any position and at any angle in 3D space. The effect tree looks like a pair of crossed wedges. The tool's effect is applied along this tree with zero at its tip and 100 percent at its base (fat end) — portions of the object beyond the base are still affected 100 percent.

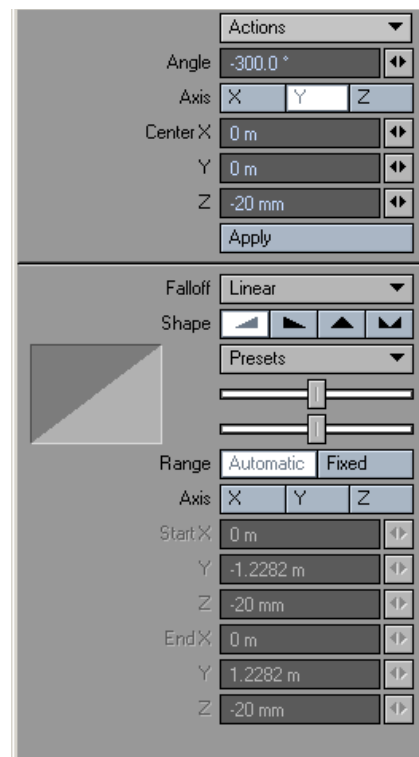
## Twist Tool

The **Twist** tool (**Modify > Twist**) is similar to **Shear**, although the results are quite different. With **Shear**, the sides of the object are moved one way or the other. With **Twist**, one end of the object is rotated while the other remains stationary. Think of Twist as Rotate with axial falloff. That is, the strength of the rotational influence is not equal across the object, but gradually falls off along a selected axis. The twist occurs around the X, Y or Z axis, so you want to set the tool in a viewport perpendicular to the axis you want the twist to occur around.



### To use the Twist tool:

Select the tool and drag your **LMB** left for counterclockwise twisting or right for clockwise. The center of the effect is determined by the **Action Center** setting.





The information display tells you the **Angle** of twisting, that is, how many degrees you twisted. Hold the **Ctrl** key down while dragging to twist in increments of 15 degrees.

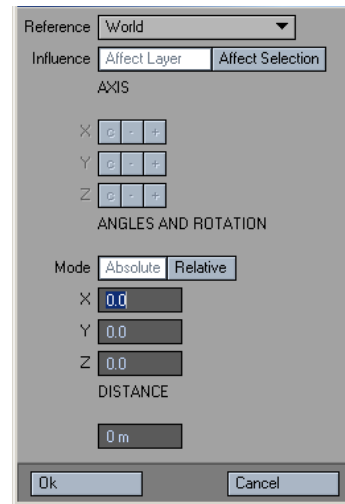
The **Angle** value on the numeric panel reflects the degrees of rotation. The **Axis** buttons relate to the rotational axis. If you use an arbitrary axis (i.e., twist in a perspective viewport), no **Axis** buttons are selected. The **Center X, Y, and Z** values indicate the center point of rotation, through which the axis runs.



NOTE: The falloff modes for this tool are described in detail starting on page 136.

## Dangle

Dangle will rotate a layer or selection relative to a rotation center. The **Reference** is what determines the axes used for rotation. The **Selection** is a bounding box around the selected geometry, **World** is the Origin, and **B.G. Layer** is a bounding box around the background layer geometry.



If **Influence** is set to **Affect Layer**, the rotation is applied to all of the geometry in the layer. With **Affect Selection**, only the selected geometry is rotated. Note that the rotation center can still be based on the **Selection**, even if you are using **Affect Layer**.

When the **c** setting is used for **Axis**, the rotation center is at the center of the **Reference**. Use the - or + options to move the rotation center to the negative or positive side of the bounding area, along the axis — these obviously have no effect if **World** is used.

When the bounding area is computed, an angle for the top “polygon” is measured based on its normal. If you use **Absolute** mode, it will change the rotation angle to be the exact amount entered. **Relative** will add the rotation to the geometry.

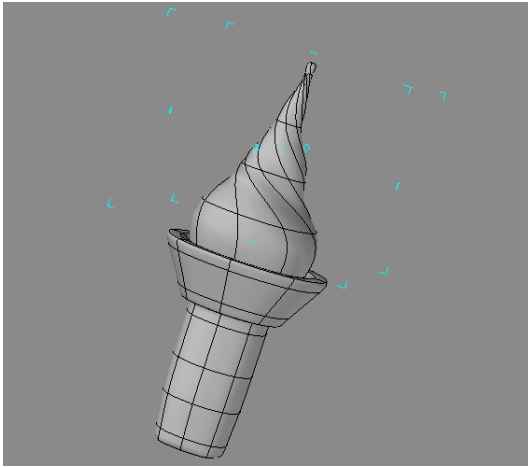
Enter the desired rotational values in the XYZ fields. These relate to the axis around which the rotation is performed.

To use the **Distance** function, select two points prior to running Dangle. The distance between those two points will be shown in the **Distance** field. Enter a new value and set **Influence** to **Affect Selection**. Click **Ok**. The two points will now be apart by the entered distance. Essentially, the points are scaled, using the rotation center defined by Dangle, until the points are at the defined distance. If you use **Affect Layer**, the entire layer will be scaled up proportionately. (If only one point is selected, the Origin is used as the other point.)



## Vortex Tool

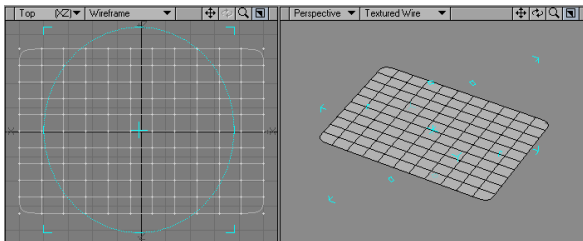
The **Vortex** tool (**Modify >Vortex**) will smoothly rotate a selected area of an object.



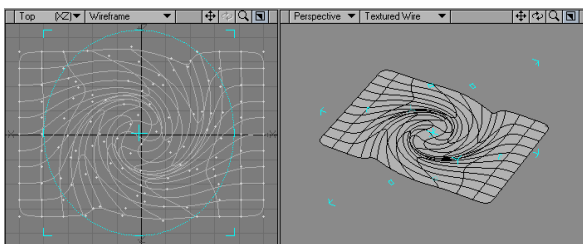
**HINT:** You might use **Vortex** to create a swirl on an ice cream cone, a whirlpool in an ocean, or a cosmic whorl of stars.

### To use the Vortex tool:

Select the tool and drag your **LMB** in a viewport. Dragging right will rotate the influence area clockwise, while dragging left will rotate counterclockwise. The center of the effect is determined by the **Action Center** setting, discussed previously.



Select Range



Vortex Applied

The information display will tell you the amount of **Angle** of rotation.



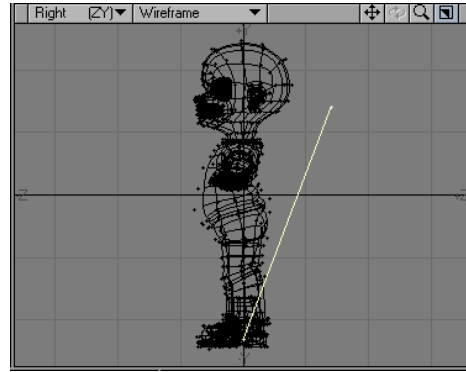
**NOTE:** The falloff modes for this tool are described in detail starting on page 136

## Rotate Any Axis

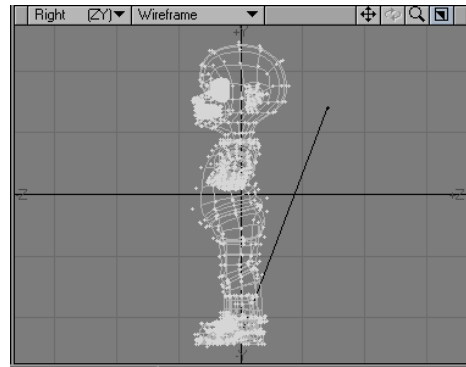
The **Rotate Any Axis** command (**Modify >Rotate Any Axis**) will rotate selected polygons around an arbitrary axis defined by a two-point polygon in the background layer.

### To Use Rotate Any Axis

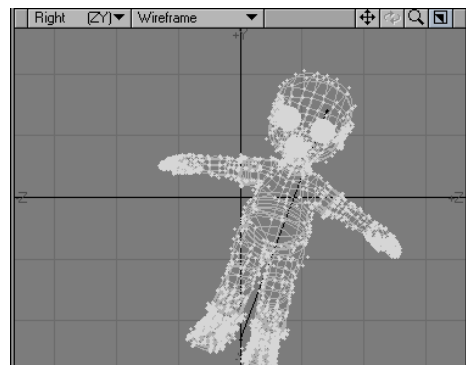
**Step 1:** Create a 2-point polygon in a background layer that you would like to use to define the arbitrary axis you would like to rotate around.



**Step 2:** Switch layers so that your object is in the Foreground and the 2-point polygon is in the Background.



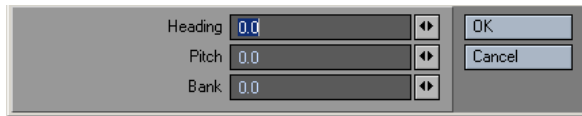
**Step 3:** Select **Rotate Any Axis** from the Toolbar and enter a value.





## Rotate HPB

The **Rotate HPB** command (**Modify > Rotate HPB**) lets you rotate the contents of a layer using heading, pitch, and bank values, as in Layout.

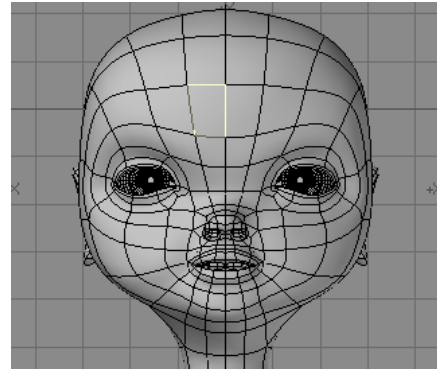


## Rotate About Normal

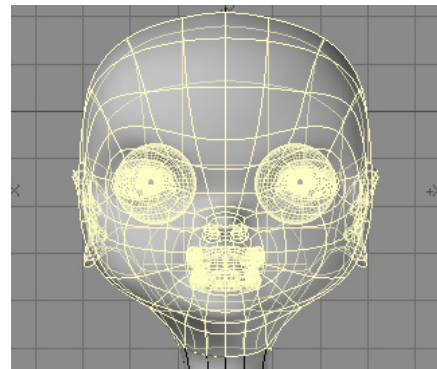
The **Rotate About Normal** command (**Modify > Rotate About Normal**) will rotate selected polygons using the surface normal of the first selected polygon as the perpendicular axis of rotation.

### Steps to Use Rotate About Normal:

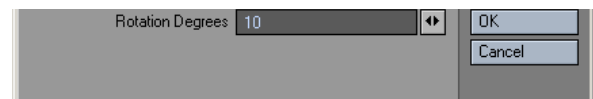
**Step 1:** Select the “Source” Polygon.



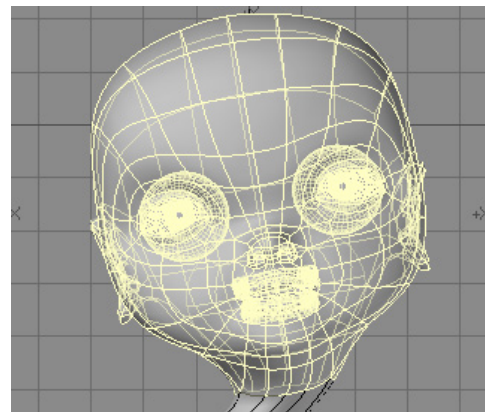
**Step 2:** Select the Polygons that you would like to rotate.



**Step 3:** Select **Rotate About Normal** and enter a value.



The Polygons selected in step 1 and step 2 will rotate around the polygon that you select in step 1.



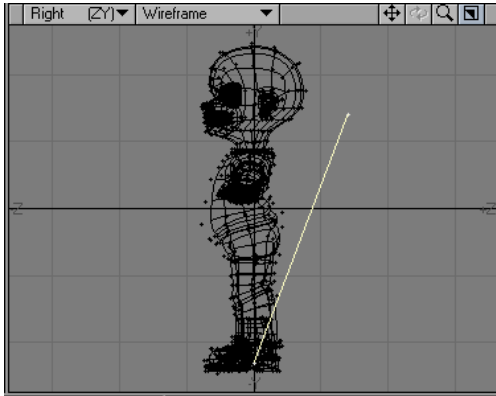


## Rotate Arbitrary Axis

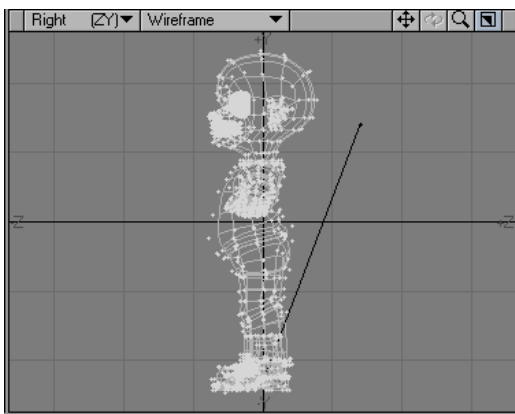
The **Rotate Arbitrary Axis** command (**Modify > Rotate Arbitrary Axis**) will rotate selected polygons around an arbitrary axis defined by a two-point polygon in the background layer.

**To Use Rotate Arbitrary Axis follow these steps:**

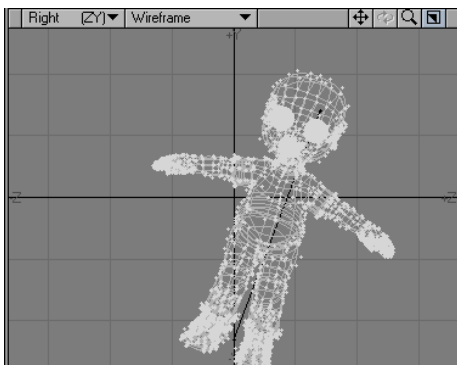
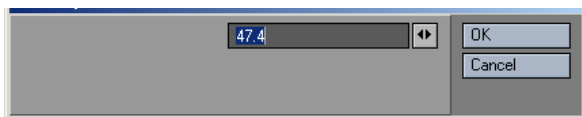
**Step 1:** Create a 2-point polygon in a background layer that you would like to use to define the arbitrary axis you would like to rotate around.



**Step 2:** Switch layers so that your object is in the Foreground and the 2-point polygon is in the Background.



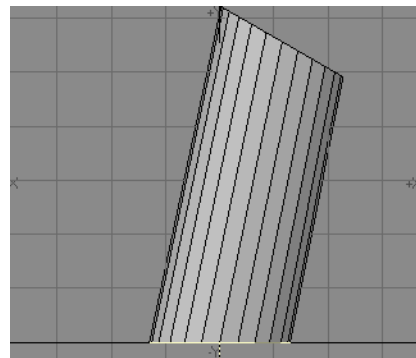
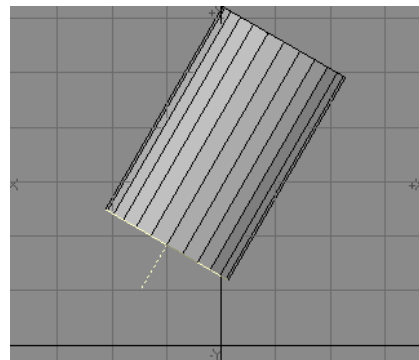
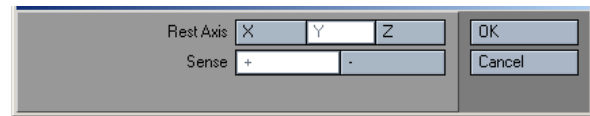
**Step 3:** Select **Rotate Arbitrary Axis** from the Toolbar and enter a value.



## Rotate to Ground

(default keyboard shortcut **F4**)

The **Rotate To Ground** command (**Modify > Rotate-To-Ground**) will rotate and move selected polygons to a defined ground plane based on the first polygon selected. The **Rest Axis** is the axis perpendicular to the plane you want the object to rest on. The **Sense** option rests the object on the positive (+) side of the plane or the negative (-) side.

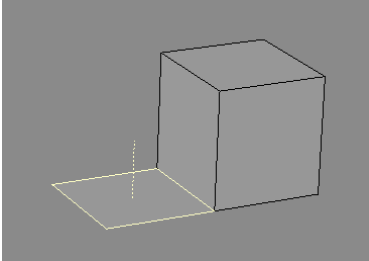




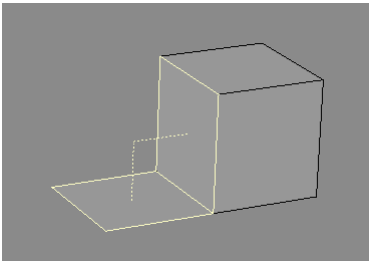
## Rotate to Object

The **Rotate To Object** command (**Modify > Rotate To Object**) will rotate in relation to a polygon. The first selected polygon defines the target orientation. The second and any other selected polygons will rotate and align to the first selected polygon.

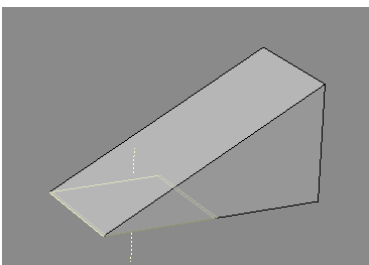
**Step 1:** Select Destination Polygon.



**Step 2:** Select Polygons to rotate.



**Step 3:** Select **Rotate To Object**.



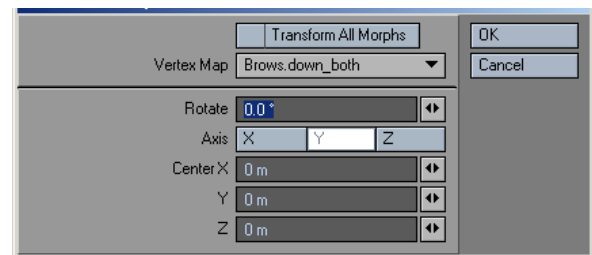
**Step 4:** will rotate to Step 1's selection.

## Rotate Morph Function

Rotating the base of an **Endomorph** with the normal **Rotate** tool often does not add the proper amount of rotation to its morph targets. You may achieve better results with the **Rotate Morph** function (**Modify > Rotate Morph**).



NOTE: You can select an Endomorph to rotate from the **Vertex Map** drop-down list, or selecting the **Transform All Morphs** option will apply the rotation to all Endomorphs in the object (which is often the preferred option in many cases where the base has been altered on an object containing many morphs)



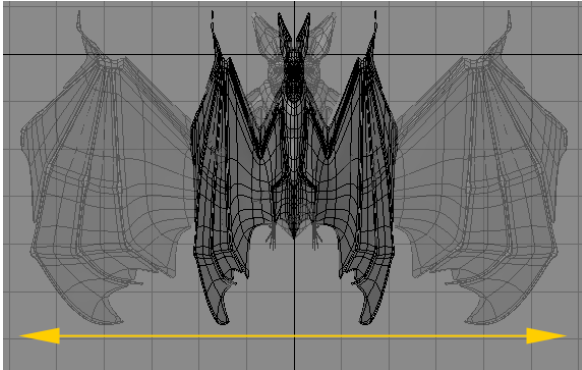


## Transform

### Stretch Tool

(default keyboard shortcut **H**)

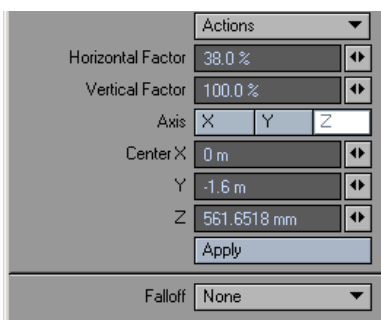
The **Stretch** tool (**Modify > Stretch**) is similar to the **Size** tool (discussed later), except that you can scale the object independently along each axis.



#### To stretch an object:

Select the **Stretch** tool (**Modify > Stretch**) and, using the **LMB**, drag your mouse right or left to increase or decrease the size along the horizontal axis. Moving up or down will increase or decrease the size along the vertical axis. The center of the effect is determined by the **Action Center** setting, discussed previously. The information display will tell you the scaling factor you have applied. Hold the **Ctrl** key while dragging to constrain movement along the initial dragging axis.

#### The Stretch Numeric Panel



The **Axis** setting defines the axis of scale. When you click in a viewport, the axis is an imaginary line running perpendicular to the view you are editing in. The **Horizontal Factor** and the **Vertical Factor** represent the amount of scaling on those axes for the viewport. The **Center XYZ** values define the coordinates for the center of scaling.

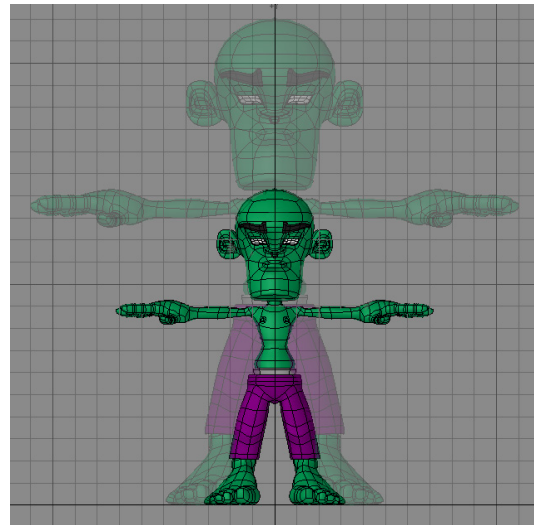


NOTE: The falloff modes for this tool are described in detail starting on page 136.

### Size Tool

(default keyboard shortcut **Shift H**)

The **Size** tool (**Modify > Stretch: Size**) allows you to scale an object uniformly along each axis, retaining its proportions.

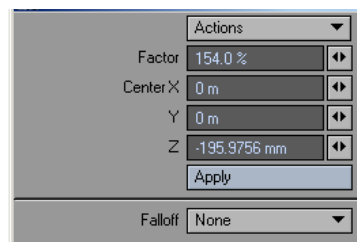


#### To size an object:

Select the **Size** tool (**Modify > Size**) and, while holding down the **LMB**, drag your mouse right to increase or left decrease the size of the object. The center of the effect is determined by the **Action Center** setting, discussed previously. The information display will indicate the scaling factor you have applied.

#### The Size Numeric Panel

The **Size Numeric Panel** has only two basic settings: the **Center XYZ** values which define the coordinates for the center of sizing; and the size **Factor**, which defines amount of sizing, as a percent.



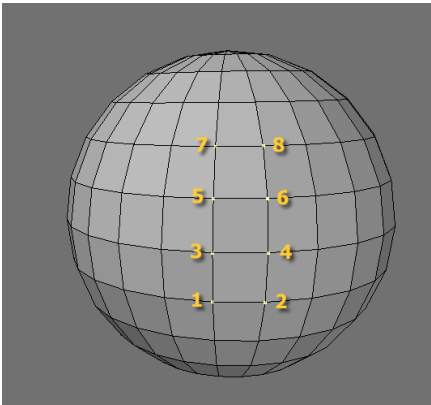
NOTE: The falloff modes for this tool are described in detail starting on page 136.



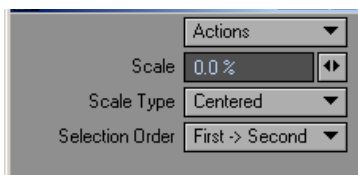


## Segment Scale

Segment Scale (**Modify>Transform>Segment Scale**) allows you to select a group of points (in order) and scale them as one segment. This is a real time interactive tool that allows direct manipulation in any Modeler viewport.



Points on this sphere have been selected in an order that allows **Segment Scale** to move the points along an edge.



### Scale

Enter the percentage you want the segment scaled or use the **Slider** button next to the **Scale** text box to increase or decrease the value.



NOTE: The **LMB** can be used in any Modeler viewport to manipulate the points directly.

### Scale Type

Allows you to choose which points to move.

Centered — Will move both sides equally.

Side 1 Only — Will move side 1 only.

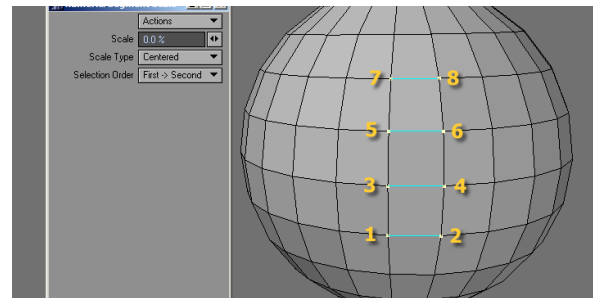
Side 2 Only — Will move side 2 only.

### Selection Order

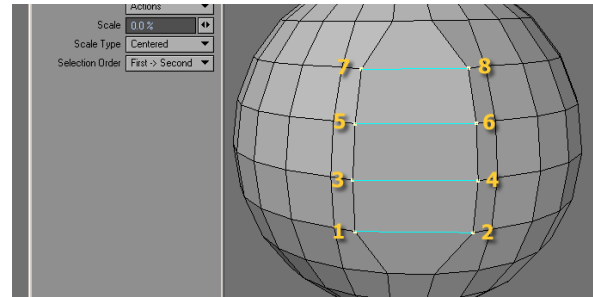
**Segment Scale** uses the point order of the selected points.

First -> Last — Loop Order.

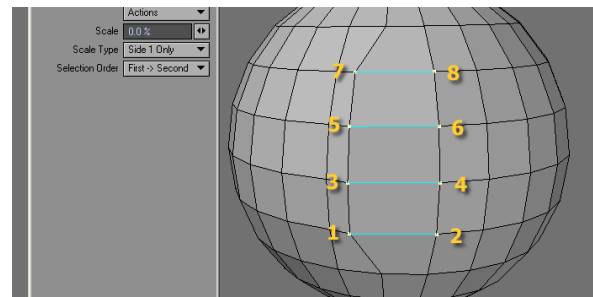
First -> Second — Pairs Examples:



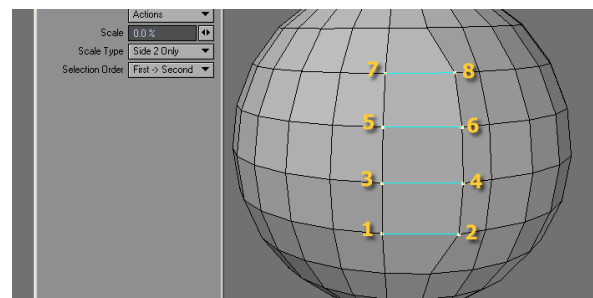
Scale Type = Centerd, Selection Order = First >Second. Before edit.



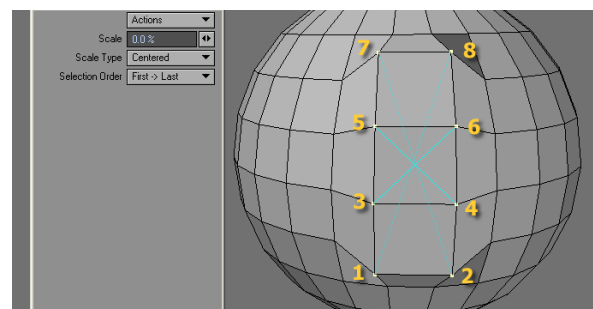
Scale Type = Centered, Selection Order = First->Second. After edit.



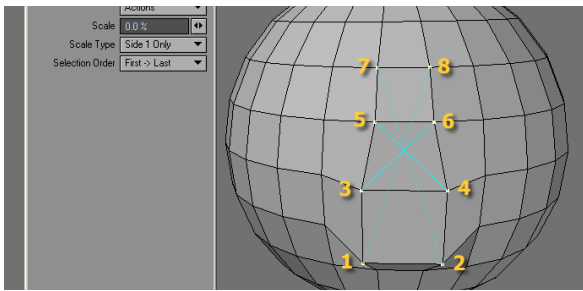
Scale Type = Side 1 Only, Selection Order = First->Second. After edit.



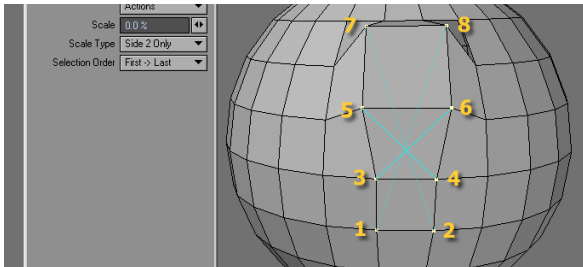
Scale Type = Side 2 Only, Selection Order = First->Second. After edit.



Scale Type = Centered, Selection Order = First->Last. After edit.



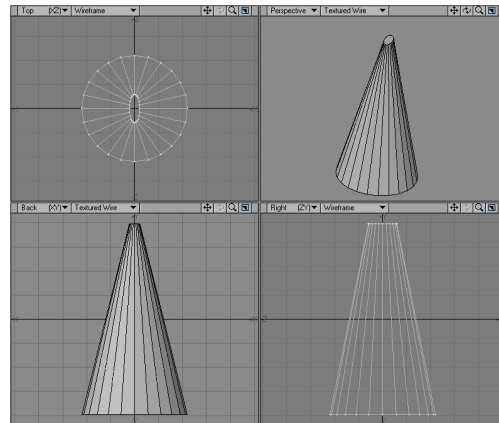
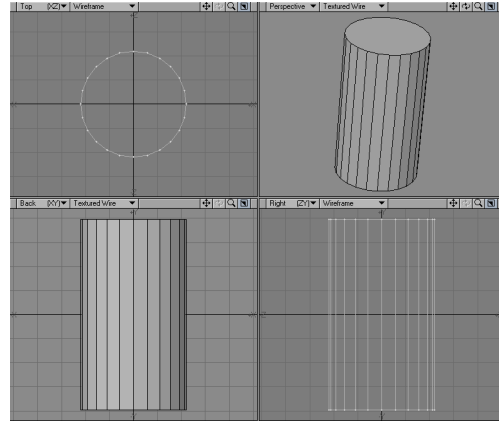
Scale Type = Side 1 Only, Selection Order = First->Last. After edit.



Scale Type = Side 2 Only, Selection Order = First->Last. After edit.

## Taper Tool

The **Taper** tool (**Modify> Taper**) is very similar to the **Taper Constrain** tool. The difference is that it allows you to scale the affected end independently on the two axes. Basically, **Taper** is to **Taper Constrain**, what **Scale** is to **Size**. Think of **Taper** as **Stretch** with axial falloff. That is, the strength of the moving influence is not equal across the object, but gradually falls off along a selected axis.

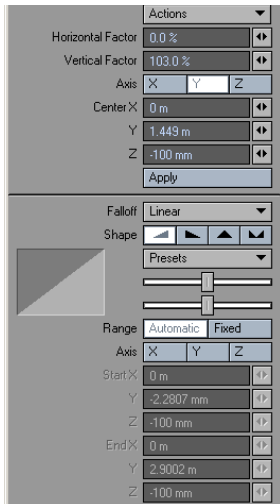




### To use the Taper tool:

Select **Modify > Taper** and drag your **LMB** left/right to apply the scaling horizontally. Drag your **LMB** down to apply it vertically. The center of the effect is determined by the **Action Center** setting, discussed previously.

The information display indicates the **Scale** factor you applied along the two affected axes. Hold the **Ctrl** key down while dragging to constrain the tapering to your initial dragging axis.



The **Horizontal Factor** and **Vertical Factor** values on the numeric panel reflect the amount of scaling in those directions relative to the viewport used. The **Axis** buttons relate to the axis along which the effect occurs. If you use an arbitrary axis (i.e., taper in a perspective viewport), no **Axis** buttons are selected. The **Center X**, **Y**, and **Z** values indicate the center point of tapering, through which the axis runs.



NOTE: The falloff modes for this tool are described in detail starting on page 136

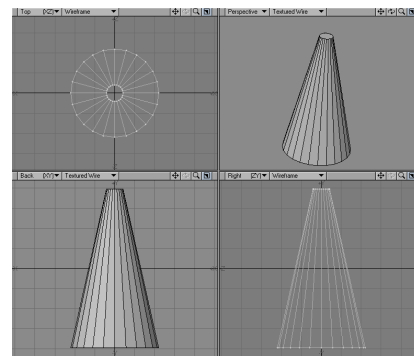
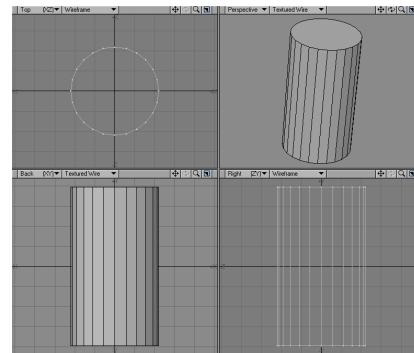
## Taper Constrain

The **Taper Constrain** tool (**Modify > Taper Constrain**) will resize an object at one end. You could use this tool to turn a box into a pyramid. Think of **Taper Constrain** as **Size** with axial falloff. That is, the strength of the sizing influence is not equal across the object, but gradually falls off along a selected axis.

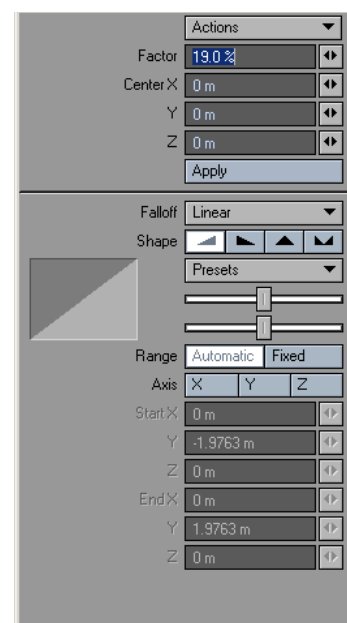
### To use the Taper Constrain tool:

Select **Modify > Taper Constrain** and drag your **LMB** left/right to apply the scaling. The center of the effect is determined by the **Action Center** setting, discussed previously.

The information display indicates the **Scale** factor you applied.



The **Factor** value on the numeric panel reflects the amount of scaling. The **Center X**, **Y**, and **Z** values indicate the center point of scaling.

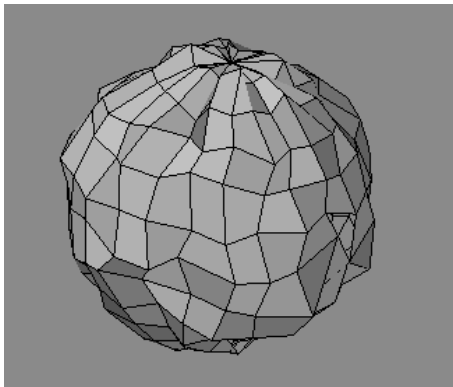
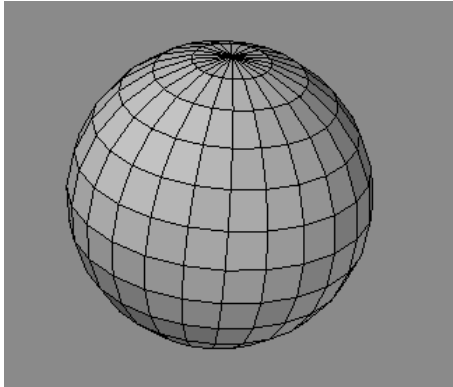




## Jitter Command

(default keyboard shortcut **Shift J**)

The **Smooth** command's ugly sister is **Jitter (Modify > Jitter)**. This command will add a roughness to the polygons of an object by randomly moving points within a certain radius of their current position. Jitter has several **Type** settings, each applying the effect in a different manner.



**Uniform** — **Uniform** performs a uniform jitter on every point within a rectangular region. A point may be shifted by as much as the value specified for the **Radius** parameter.

**Gaussian** — **Gaussian** results in a slightly less ragged jitter than the **Uniform** option, performing a normal distribution around the starting point's location in an ellipsoid of the given **Radius**. A point may be shifted by as much as the value specified for the **Radius** parameter.

**Normal** — **Normal** uses the local surface normal and randomly moves points plus or minus within the specified **Range** along that surface normal.

**Radial** — **Radial** randomly distributes points inward or outward along a radial vector from a single point defined by the **Center** coordinates.

**Range** is the value used to set the range within which a point may move from its current location.

**Scaling** — **Scaling** is a variation of the **Radial** option. Points are jittered outward from a center point defined by the **Center** coordinates. **Factor** is the value used to set the range, as a percentage, within which a point may move from its current location.

A jitter value of 0 applies no jittering, meaning no change will occur. Values above or below 0 will affect the object.

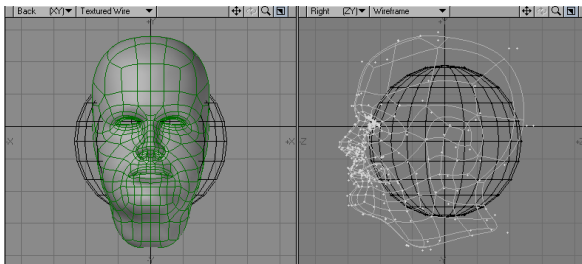


## Background Conform (BG Conform)

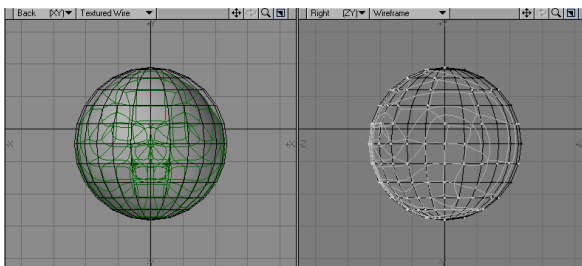
The **Background Conform** function is available through the Legacy plugin Spherize.p. To install it and get the use of BGConform, Spherize and Squarize, use the Add Plugin command. To add it to your **Modify** tab, use the Edit Menu function (**Alt F10**). This function will take an object in the Foreground layer and fit it to an object in the Background layer.

### Steps for using Background Conform:

**Step 1:** Place the object that you want manipulated in the Foreground and the “effector” object in the Background.



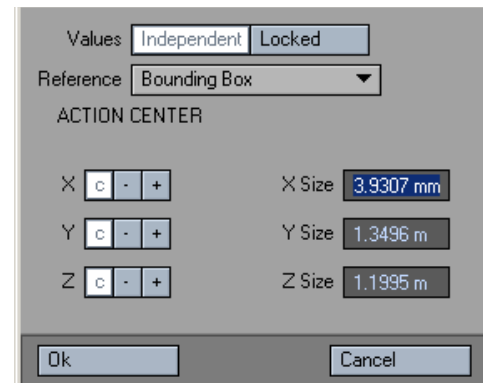
**Step 2:** Click on the **Background Conform** button and presto! Instant conform.



NOTE: This is a handy tool for creating **Morph Targets**!

## Absolute Size

**Absolute Size (Modify > Transform: More)** allows you to scale geometry with great flexibility and precision. With the **Values** setting set to **Independent**, the object can be scaled independently on each axis. If **Locked** is used, the aspect of the geometry is maintained, meaning you only need to change one value and the others will change accordingly, to keep the proper relative proportions of the geometry.



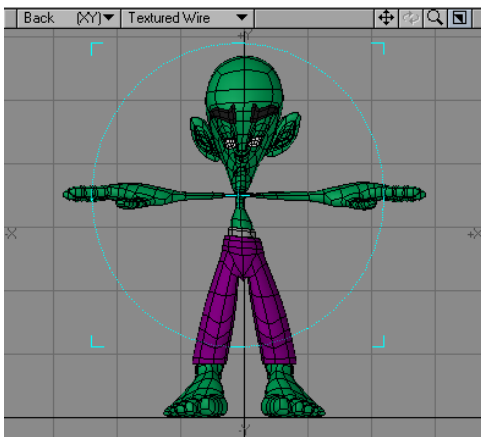
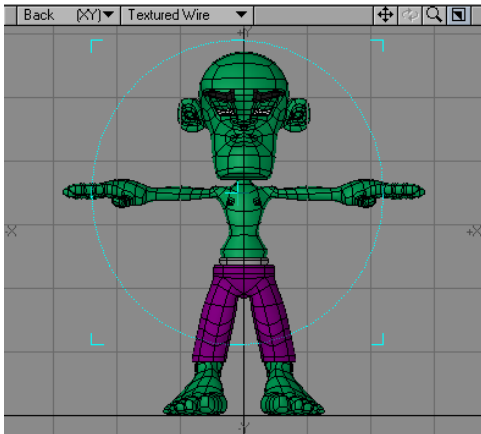
### Reference

The **Reference** setting determines a reference for the selected **Action Center**. **Bounding Box** uses a bounding box around all of the geometry. **World** uses the Origin. **B.G. Layer** uses a bounding box around geometry in the background. **Absolute** lets you numerically set the center position. The **c** setting will scale using the center of the reference. The minus (-) or plus (+) setting will scale using the negative or positive side of the reference.



## Pole Evenly Tool

The **Pole Evenly** tool (**Modify > Pole Evenly**) scales an area of an object along the vertical and horizontal axes.



The **Factor** value on the numeric panel reflects the amount of scaling. The **Center X**, **Y**, and **Z** values indicate the center point of scaling. You can edit the values and then click the **Apply** button to apply them, which you can do multiple times.

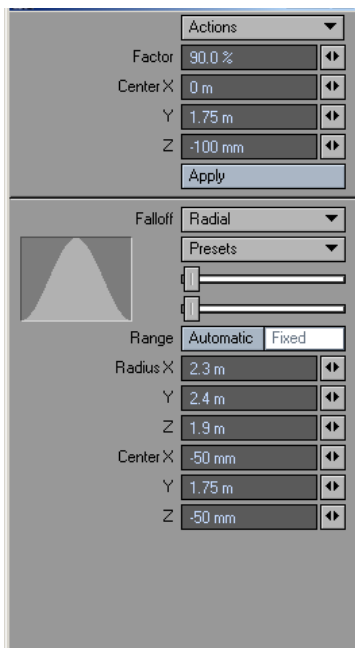


NOTE: The falloff modes for this tool are described in detail starting on page 136.

### To use the Pole Evenly tool:

Select **Modify > Pole Evenly** and drag your **LMB** in a viewport. The center of the effect is determined by the **Action Center** setting, discussed previously.

The information display indicates the **Scale** factor you applied.

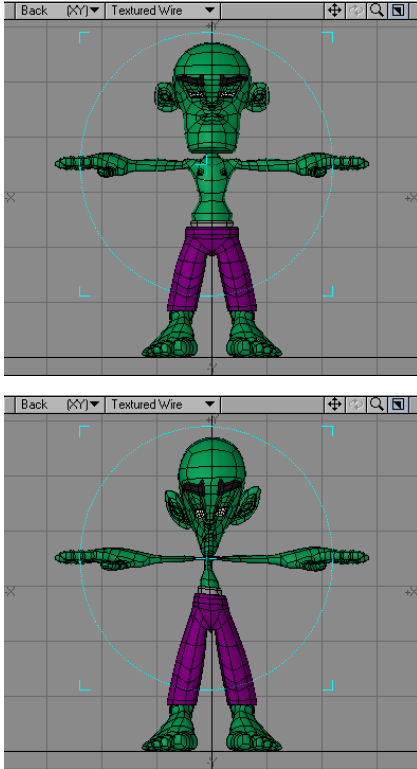




## Pole Tool

The **Pole** tool (**Modify** > Stretch: **Pole**) is identical to **Pole Evenly**, except that you can scale an area independently on the two axes—dragging your LMB left/right to affect horizontally and up/down to affect vertically.

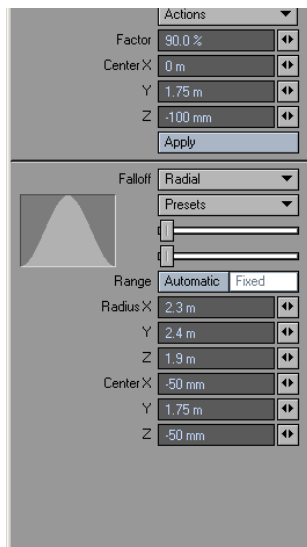
The information display indicates the **Scale** factor you applied along the two relevant axes.



### To use the Pole tool:

Select **Modify** > **Pole** and drag your **LMB** in a viewport. The center of the effect is determined by the **Action Center** setting, discussed previously.

The information display indicates the **Scale** factor you applied



The **Horizontal Factor** and **Vertical Factor** values on the numeric panel reflect the amount of scaling in those directions in relation to the viewport used. The **Axis** buttons relate to the axis along which the effect occurs. If you use an arbitrary axis (i.e., use in a perspective viewport), no **Axis** buttons are selected. The **Center X**, **Y**, and **Z** values indicate the center point of scaling, through which the axis runs. You can edit the values and then click the **Apply** button to apply them, which you can do multiple times.



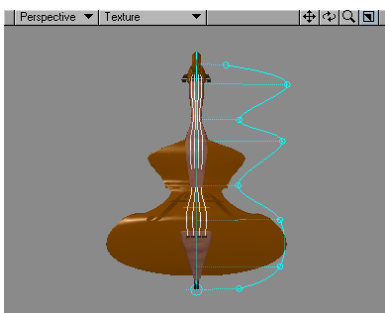
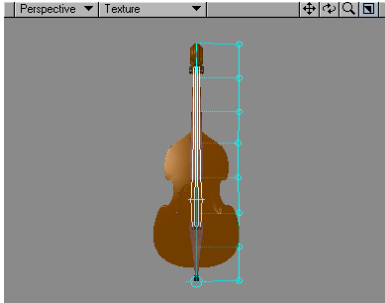
NOTE: The falloff modes for this tool are described in detail starting on page 136.





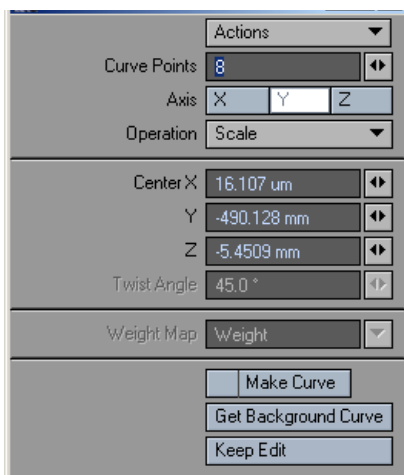
## Spline Guide Tool

Generally, the **Spline Guide tool (Modify > Spline Guide)** performs interactive deforming operations based on a *guide* curve. Clicking in the window creates the initial guide curve and center axis. Initially, the guide curve will be aligned along the longest distance of the selected geometry, extending from its minimum to maximum point. The center axis is centered within the geometry.

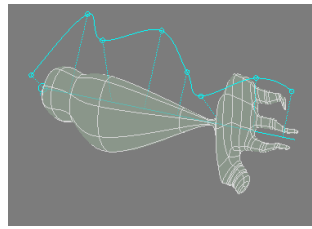


Initially, the guide curve points have no effect on the geometry—like bones in their rest position. However, moving the points will vary the strength of the effect. Moving the center axis will change the influence of all the points on the curve.

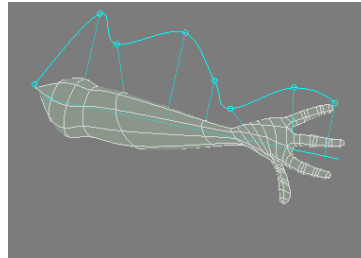
What effect is applied is determined by the **Operation** setting on the numeric panel.



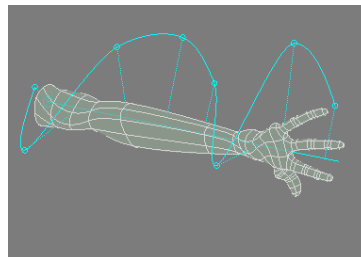
The **Scale** operation scales the geometry in the plane perpendicular to the axis around the center axis. Moving the guide points closer to the center shrinks the geometry, moving them farther away as it expands it.



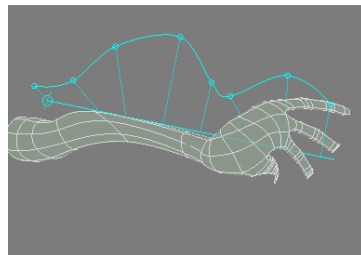
The **Stretch** operation scales the model in the plane perpendicular to the axis around the center like the **Scale** operation, but uses different amounts of scaling in the plane (call it X-Y) depending on the X and Y distance of the guide points from their initial positions. Thus, the geometry can be expanded in one direction and reduced or left unchanged in the other.



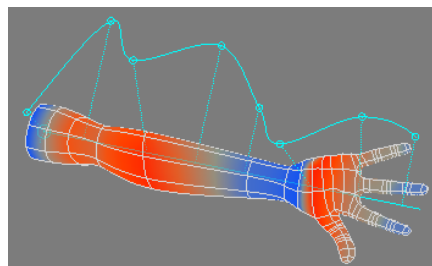
The **Twist** operation rotates the geometry around the center axis. The **Twist Angle** parameter acts as an angle scaling factor.



The **Bend** operation bends the model to conform to the curve.

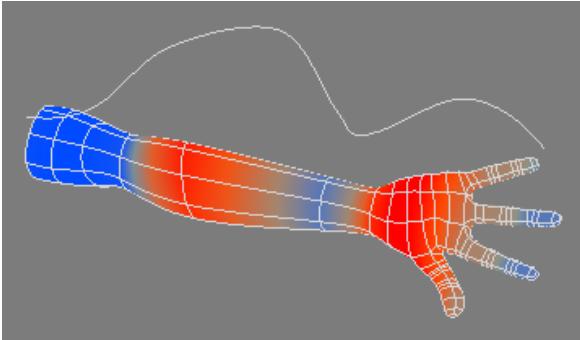


The **Weight Map** operation works a little differently than the other operations in that it doesn't deform the geometry, but affects the selected **Weight Map** instead. Essentially, it sets the values based on the reshaped curve. Move them closer to the center axis for negative weights, farther for positive.





If you activate **Make Curve**, a curve polygon will be created from the guide curve when you deselect the tool. Use this to save the guide curve shape. This curve can be placed in a background layer and *loaded* into the guide curve by clicking **Get Background Curve**.



Click **Keep Edit** to accept the current changes without resetting the curve. This is useful if you want a certain curve to be applied in subsequent operations.

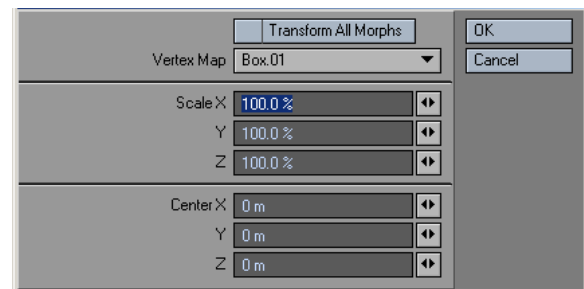
Changing the **Curve Points** value will *resample* the curve into more segments. Note that resampling the curve can change its shape, particularly if the number is incremented gradually, as is the case with the mini-slider. If you want a more refined curve that accurately matches the one you have drawn, enter the new number directly into the field to avoid the intermediate curves.

## Scale Morph Tool

Scaling the base of an **Endomorph** with a normal sizing tool may not add the proper amount of scaling to its morph targets. You may get better results with the **Scale Morph** function (**Modify > Scale Morph**).



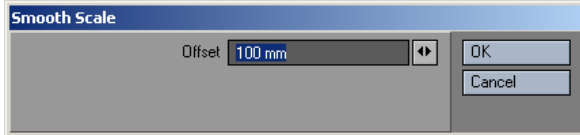
NOTE: You can select an Endomorph to scale from the **Vertex Map** drop-down list, or selecting the **Transform All Morphs** option will apply the scale to all Endomorphs in the object (which is often the preferred option in many cases where the base has been altered on an object containing many morphs)



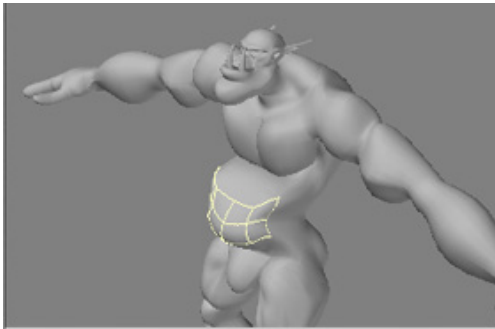
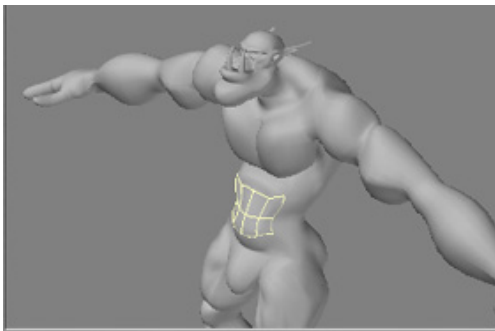


## Smooth Scaling

A variant of the **Stretch** tool is the **Smooth Scale** command (**Modify > Smooth Scale**). This command will scale selected polygons, but attempt to smooth things out as it scales. No new geometry is added, so the smoothing is done using existing points and polygons. The scaling is done based on the normals in the selected geometry.



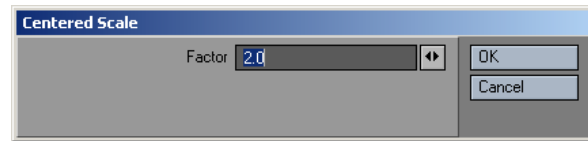
The **Offset** value, editable on the numeric panel, is the exact amount that the object will increase in size overall; however, due to the smoothing effect, not all parts will increase this much. Negative numbers are also allowed, which reverse the effect.



**HINT:** **Smooth Scale** is great for use on SubPatch objects. Use it on the stomach of a figure to give it a beer belly.

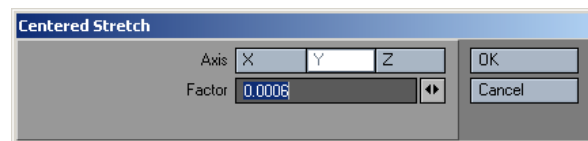
## Center Scale

**Center scale** is located under the **Modify Tab** and will scale a selection around its center point. **Factor** is the value (percentage) of the scale operation.



## Center Stretch

**Center Stretch** is located in the **Modify Tab** and will stretch a selection on the X, Y, or Z axis around the selections center. **Factor** is the value (percentage) of the stretch operation.





## Smooth

(default keyboard shortcut **Shift M**)

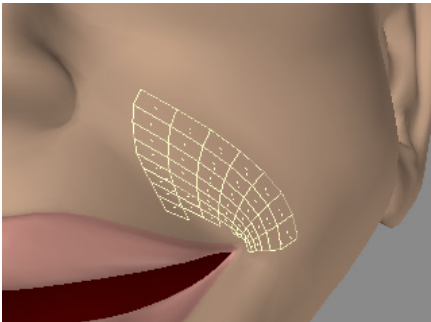
Applying the **Smooth** operation to a group of selected points or polygons will take the average position of the points and smooth out the area.

### Steps for applying the Smooth operation

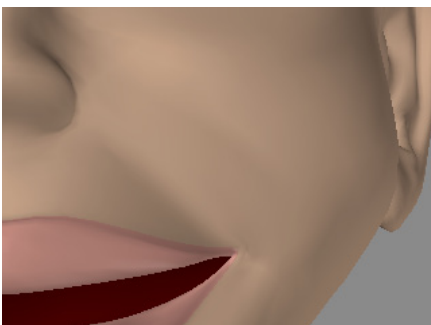
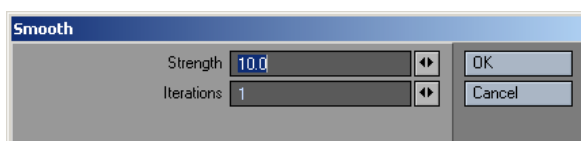
**Step 1:** Locate an area where you would like to smooth out the geometry.



**Step 2:** Select either the points or the polygons that make up the area.



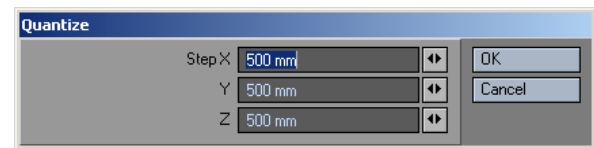
**Step 3:** Enter in the values for **Strength** and **Iterations**. **Strength** will set the amount of smoothing you would like to use. The **Iterations** value will determine how many times to apply smooth.



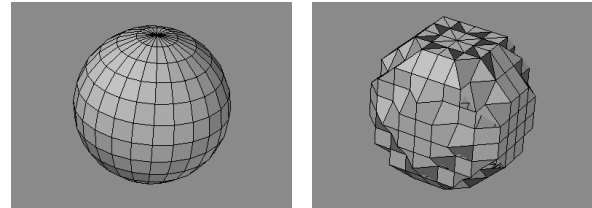
NOTE: Working with smaller values of **Strength** and larger values of **Iterations** is suggested.

## Quantize Tool

The **Quantize** command (**Modify > Quantize**) will snap points to specific coordinate spacing intervals (like a grid).



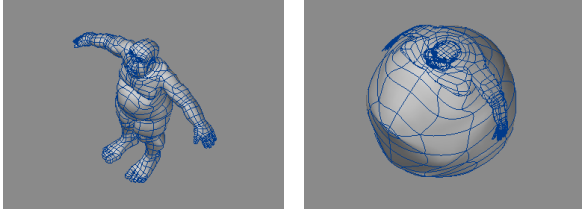
This is very useful for snapping a non-planar polygon into a specific plane or flattening one side of an organic or curved object.





## Spherize

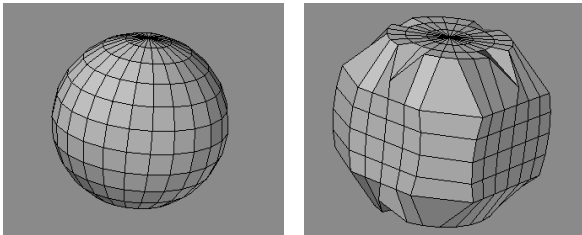
Like BGConform, the **Spherize** tool can be added to your **Modify tab** by using the Add Plugin command and getting the Spherize.p plugin from the Leacy plugins folder. It can be added to the Modify tab using the **Edit Menus** command **Alt F10**. **Spherize** will take the selected geometry and deform it into a sphere with its center being the origin. (0,0,0).



NOTE: There isn't a numeric panel for this tool so the outcome of the operation is dependent on the size and placement of your object.

## Squarize

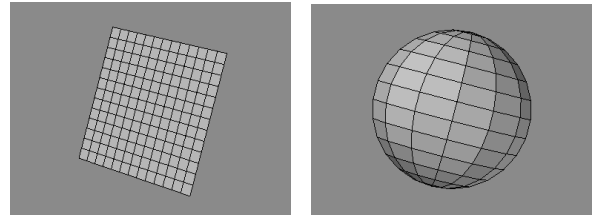
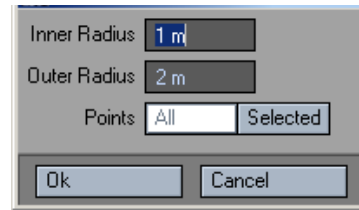
Like BGConform, the **Squarize** tool can be added to your **Modify tab** by using the Add Plugin command and getting the Spherize.p plugin from the Leacy plugins folder. It can be added to the Modify tab using the **Edit Menus** command **Alt F10**. **Squarize** will take the selected geometry and force it to the object's bounding box.



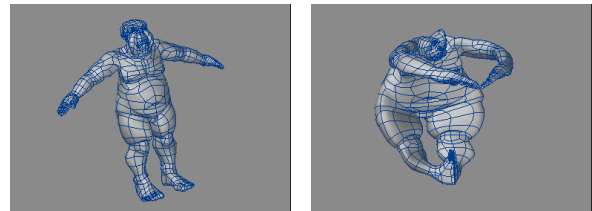
NOTE: There isn't a numeric panel for this tool so the outcome of the operation is dependent on the size and placement of your object.

## Wrap Sphere

The **Wrap Sphere** tool is located under the **Modify Tab**. **Wrap Sphere** will take the selected geometry (or all geometry) and deform it into a sphere. This is a great tool to use to create a Globe that unwraps into a flat map.



Results will vary depending on which objects Wrap Sphere is applied to.

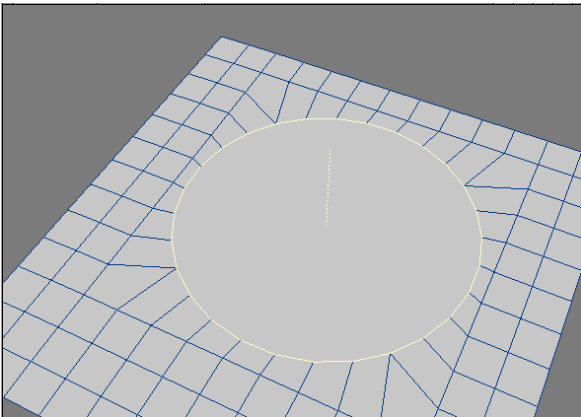
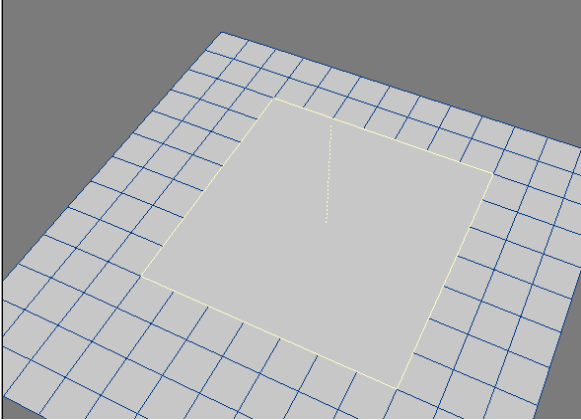




## Untangle

When a polygon is selected, this tool will make it circular. The more sides the polygon has, the closer to a perfect circle it will become.

\









---

## Chapter 5: Multiply Tab

---



## Multiply Tab

These tools allow you to add to geometry using a variety of tools. Existing polygons can be used to create new detail in a model, including processes whereby one polygon can be turned into a complete 3D object, and objects can be multiplied in a variety of ways.

## Extend

The Extend tools offer ways to use one or more existing polygons to create additional polygons and add depth and detail to objects or to create entire objects.

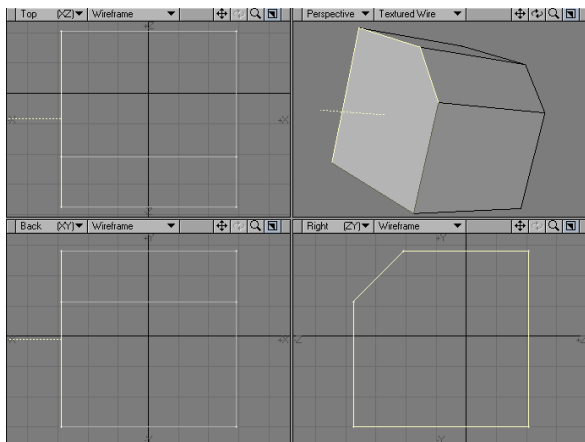
## Bevel Tool

(default keyboard shortcut **B**)

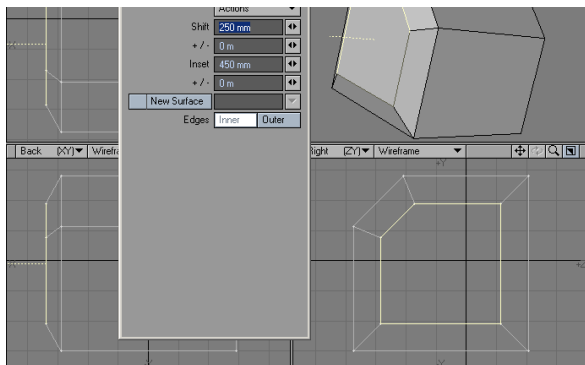
The **Bevel** tool (**Multiply > Extend: Bevel**) is likely one of the most commonly used Modeling functions. Basically, it takes a polygon and extrudes it away from the original along the surface normal. It is similar to using the **Extrude** tool, but there will be no polygon at the position of the original polygon. When applied to multiple polygons, each polygon receives its own bevel — even on a double-sided polygon.

### Steps for Beveling a Polygon

**Step 1:** Select the polygon(s) you would like to **Bevel**.



**Step 2:** Left-click and drag to create a bevel.



Dragging your pointer after activating the tool lets you graphically place the bevel. Deactivating **Bevel** or selecting another tool makes the bevel. Click on an open area of the interface to cancel

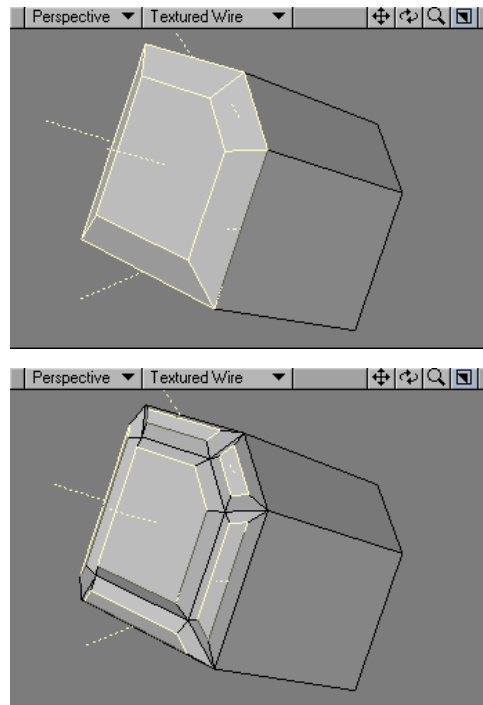
the bevel before it is made or use **Undo**.

Dragging up/down lets you interactively change the **Shift** amount, the distance that the selected polygon will move along its surface normal. On the numeric panel, positive values make the bevel move in the same direction as the normal. Negative values make it move in the opposite direction.

Dragging left/right lets you interactively change the **Inset** amount, the distance that the edges of the polygon will move in the same plane as the polygon. On the numeric panel, positive values move inward and negative values move outward.

Drag at an angle to simultaneously change **Shift** and **Inset**. Hold the **Ctrl** key to constrain to **Shift** or **Inset**, depending on which direction you drag initially.

If you select polygon(s), the top polygon will remain selected so you can perform bevel operations one after another. Click your **RMB** to accept the current bevel and begin a new bevel operation. You can also press the **B** key twice to make and start a new bevel. You can also bevel multiple polygons simultaneously!



### Other Numeric Options

The +/- fields on the numeric panel let you set randomisation for the **Inset** and **Shift** values. The actual value applied will be varied by a random number between plus or minus the +/- value. For example, if **Inset** was 20m and its +/- was 5m, then the actual **Inset** would be between 15 and 25 meters. Generally, you want to use this feature only when beveling many polygons simultaneously.





**HINT:** To avoid negative random values, make sure the +/-value is always less than or equal to the related **Inset** or **Shift** value.

**Inner** is the default **Edges** setting and bevels the polygon inward and forward along the surface normal. **Outer** will bevel outward and backward; however, the original polygon does not move and the new bevelled edges extend away from the surface normal. This option reverses the mouse movement effects. In most cases, you use **Inner**.

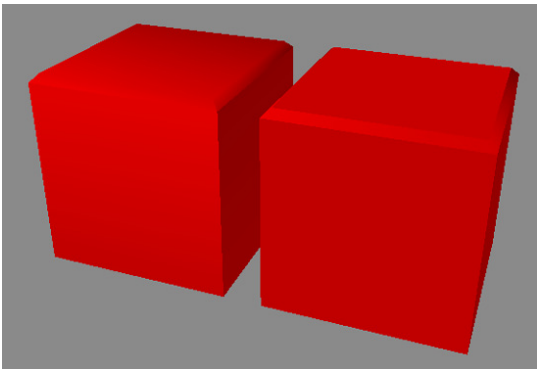
By default, the surface name of the source polygon is used for the new geometry. If you select **New Surface**, you can enter a surface name (even an existing one) in the input field. You can change this setting anytime before *accepting* the bevel.



**NOTE:** The selected polygon retains its surface name; only the new edge polygons receive the defined name.

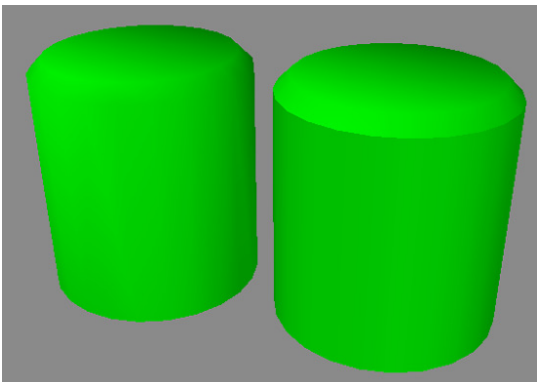
## Bevelling Tips

After performing a bevel, the selected polygon remains selected. To accentuate the bevelled edge when the surface uses the **Smoothing Surface** option, perform a quick cut and paste to separate the selected (top) polygon from the added sides. This action precludes LightWave from smoothing over the bevelled edge. You may also want to cut and paste the original polygon (before performing the bevel). Make sure you do not merge points after doing this!



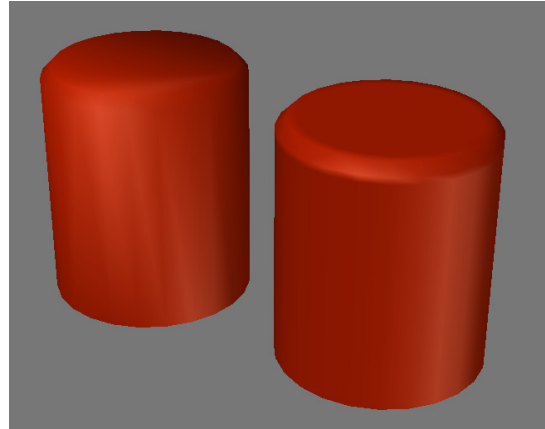
Left: Polygons Not Separated, Right: Polygons Separated

If you use the same surface, which uses **Smoothing**, and the geometry goes from straight to curved, you may want to separate the polygons (with a cut-and-paste) at the transition point to get a clean edge.



Left: Polygons Not Separated, Right: Polygons Separated

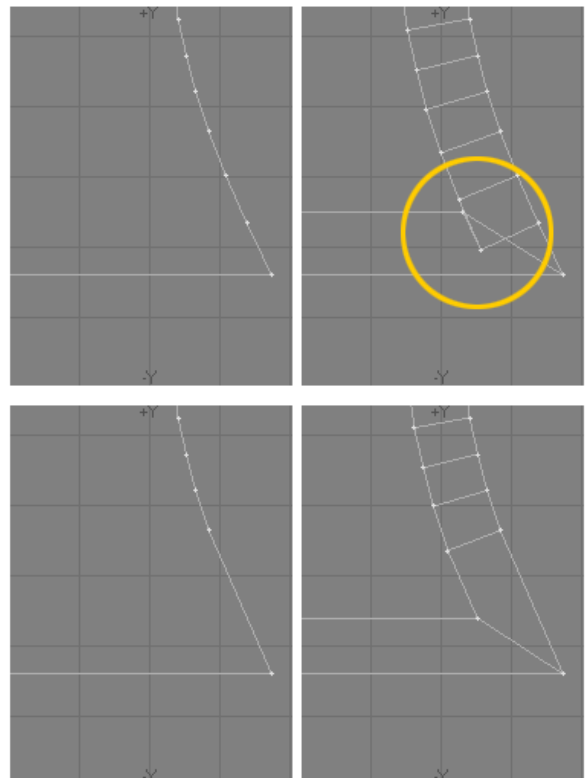
If you wish for your bevel to become a smooth rounded edge that appears to blend into the object instead of the sharp defined edge you get with separating the polygons, add an extra bevel before, and after your bevel (a subtle shift and subtle inset). This will allow LightWave to correctly shade the bevel to appear the smoothly connect with the rest of the geometry.

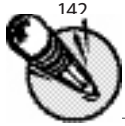


If you want to create an enclosed solid object when bevelling a 2D polygon, copy it to another layer first and choose **Detail > Polygons: Flip**. Go back to the original layer and perform the bevel normally. Then, paste the flipped polygon back into the original layer and choose **Detail > Polygons: Merge Polys** to merge points.

Cut and paste a polygon (like one side of a box) away from its host and bevel it a very small amount. Then, cut and paste the top polygon as well. This makes a nice clean edge that catches specular glints.

When you bevel polygons with sharp edges, you can often run into the problem of points being too close at the corners — very common when you bevel text characters. One solution is to delete unnecessary points near the corners prior to bevelling. Don't be scared to delete too many points; the change is usually so slight that no one will notice, particularly if the text is animated.



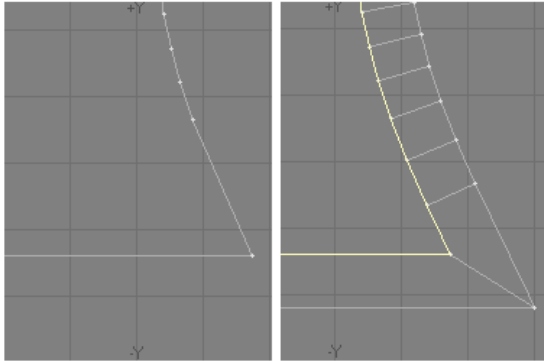


147



NOTE: Sometimes you also need to manually drag points on the new polygon away from the corner.

Another corner-fixing trick is to bevel out instead of in. This has the benefit of not requiring you to work on any points, but it affects the spacing between characters, if you are working with text. For this, use the **Outer** setting for **Edges** in the **Bevel** numeric panel.



## Edge Bevel

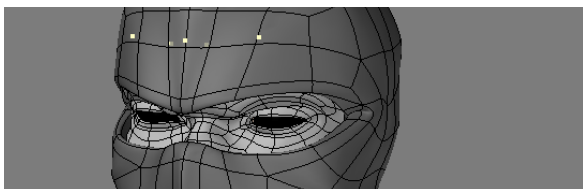
(default keyboard shortcut **Ctrl B**)

**Edge Bevel** is a tool for rounding or splitting edges. It can be used in either point, edge or poly mode. In point or edge mode, the edges, or the edges that are defined by selected points, will be split (beveled). In poly mode, only the outer edge of the selection will be split.

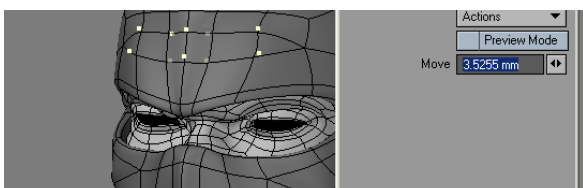
Select **Edge Bevel** (**Multiply > Extend: Edge Bevel**) to Bevel selected polygons or edges (by point selection). The Bevel can be adjusted by scrubbing in the interface or by using the numeric window. Further adjustments can be made until the tool is dropped (**space bar**).

### Steps for bevelling an edge

**Step 1:** Select a group of points that make up the edge you would like to bevel.



**Step 2:** Choose **Multiply > Edge Bevel** and adjust the **Move** setting.



NOTE: If only one point is selected, nothing will happen when the tool is run (it's an **Edge Bevel** tool, not a **Point Bevel** tool)

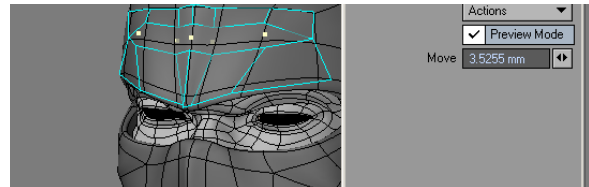


NOTE: If all or no polygons are selected (in poly mode), the tool will do nothing because it bevels only the outer edge of the poly selection. If they are all selected, there is no outer edge.



HINT: If you want to round the edges on a cube that is higher resolution, select only the edges of the cube, not any of the points between edges.

Use the **Preview Mode** on slower machines to speed up interactivity. To use the tool in **Preview Mode**, make selection first and then choose **Edge Bevel** from the **Multiply Tab**. To adjust size of bevel, click and drag on screen. When it is at the desired position, right click and it will set it. The size of the bevel can still be adjusted more if needed or if finished, just drop the tool.



NOTE: **Edge Bevel** will give unpredictable results if the points selected have missing adjacent polygons.

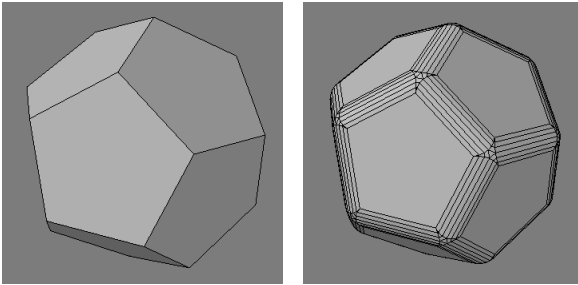


HINT: Edges that are not in the outer edge category can also be split. Selecting any two or more points in a row will define an edge to be split. Therefore, on outer edges, the hard corners will become soft, on inner edges, the edges will be split but stay on their current plane.



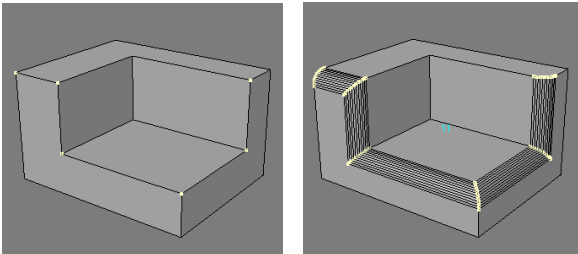
## Rounder Tool

The **Rounder** tool (**Multiply > Extend: Rounder**) will bevel, smooth edges and generally round off geometry in a variety of ways. It rounds based on point selection, bevels edges, smooths corner edges, has interactive controls, presets and more.



### Selecting Edges

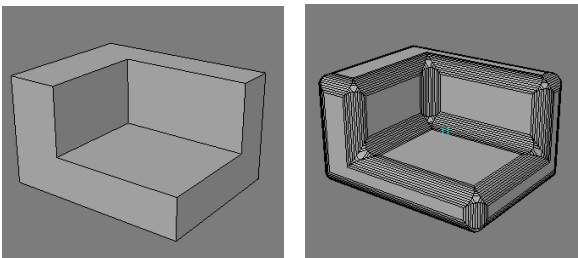
The edges you want to round are selected by selecting the edges or points at either end of the edge — illustrated below, or selecting the edge in Edge mode:



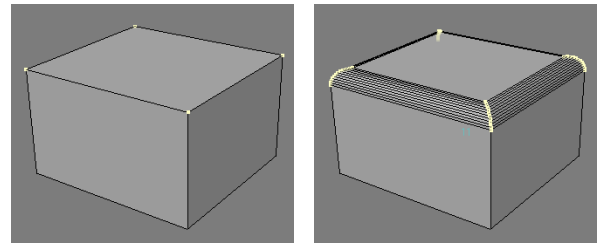
If no points are selected, as with other Modeler tools, **Rounder** will assume *all* the points have been selected (and hence round all edges/points).



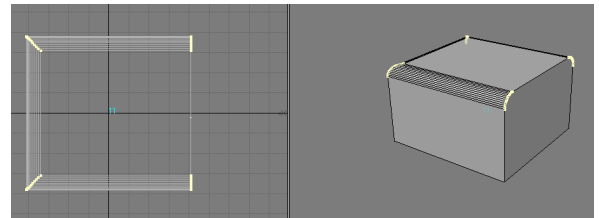
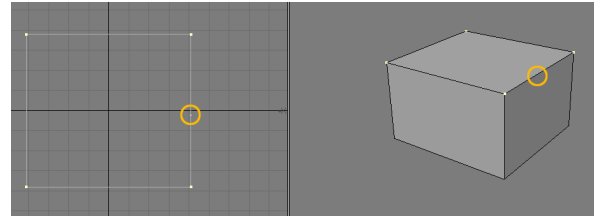
Hint: If the selected object is small in scale and Rounder is activated, the default values may make the object explode in size. If this happens, open the Numeric Panel and reduce the Inset Scale.



NOTE: With the above method of edge selection it is NOT possible to select all but one edge of a polygon (eg. to round three edges of a square on one side of a cube). As shown on the left, if all four points are selected then the polygon is bevelled.



To only round three edges (of the four), add an extra point to the edge you don't want rounding.



### Selecting Polygons

If no polygons are selected, **Rounder** will scan all the polygons of the object. If you are only rounding a few edges of an object containing a large number of polygons you can speed up the action of **Rounder** by selecting the polygons containing only the points/edges you are rounding.



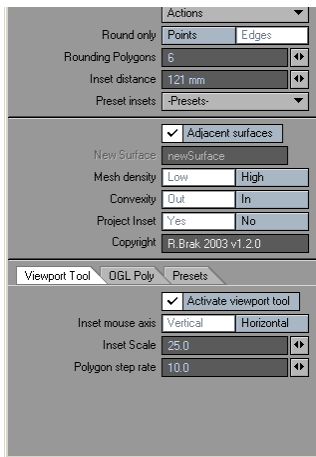
NOTE: When selecting polygons make sure you select all the polygons adjacent to the edges/points you are rounding, otherwise **Rounder** will generate an error.

**Clean Geometry.** An important part of not generating errors with **Rounder** is to ensure the geometry is clean:

1. Rounding edges must be shared by *exactly* two polygons.
2. Polygons containing rounded edges *must* be planar.
3. Make sure no isolated points or two-point polygons are selected



## The Rounder Numeric Window



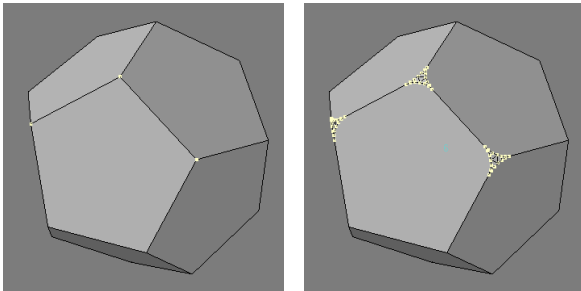
### Actions

**Activate Rounder** — you must either select **Activate** or click on a Modeler viewpoint before **Rounder** will actually round any edges.

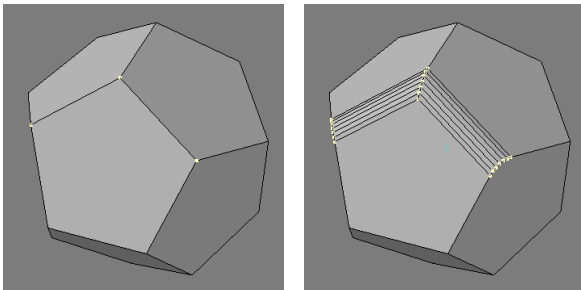
**Reset Rounder** — select this to set all the controls on the panel back to their previous (ie. the last time **Rounder** was used) values.

### Points/Edges

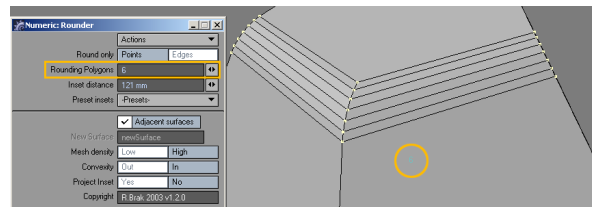
Select **Points** if you only want the points of your selected points to be rounded.



Select **Edges** if you only want the edges defined by your selected points or edges to be rounded.

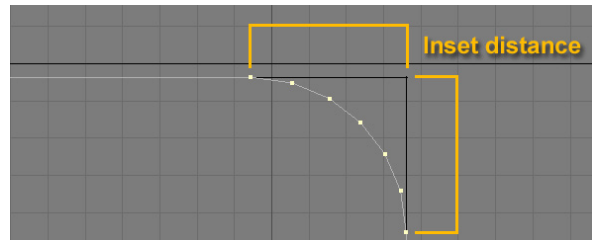


**Number of rounding polygons** — This option sets the number of bevels that will be applied to the edge as illustrated below:



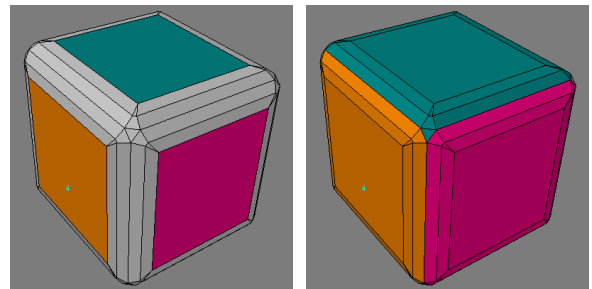
**NOTE:** The number of **Rounding Polygons** setting will appear in the viewport.

**Inset distance** as illustrated below:



**Inset Presets.** This is a popup list of preset inset distances (in the units chosen in Modeler's **Options** panel). You can choose which list of presets to use on the **Presets Tab**.

**Adjacent surfaces.** This controls what surface is allocated to the newly created rounding polygons. If this is selected then **Rounder** will assign the surface to that of the original polygon adjacent to the selected edge as illustrated below:



If it is unchecked, then a new surface is assigned to the new polygons. The name of the new surface is given by the "New Surface" control.



**NOTE:** The central polygon on the point corner is adjacent to all the original polygons and hence its surface allocation is intrinsically ambiguous — it is assigned the surface the rounding algorithm first comes across.

**New Surface.** All newly created rounding polygons are assigned





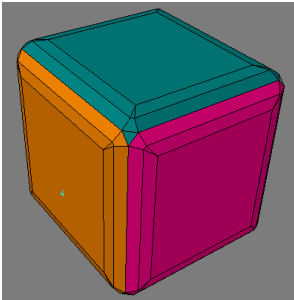
this surface.



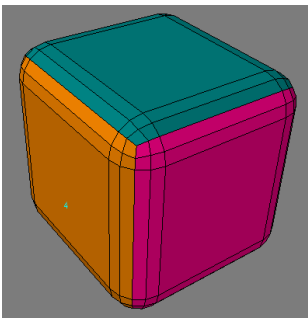
NOTE: If the **Adjacent Surfaces** checkbox is selected this option is not available (and hence ghosted).

**Mesh Density.** This determines the type of mesh on rounded points.

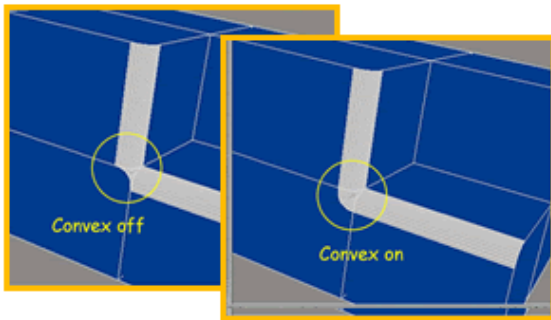
**Low** — this creates a mesh with a minimum number of polygons.



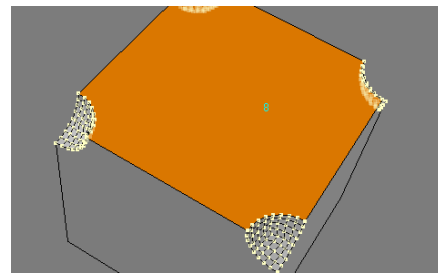
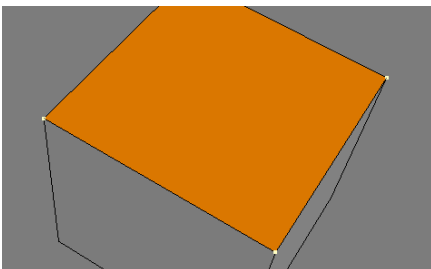
**High** — this creates a mesh on the corners with only quads.



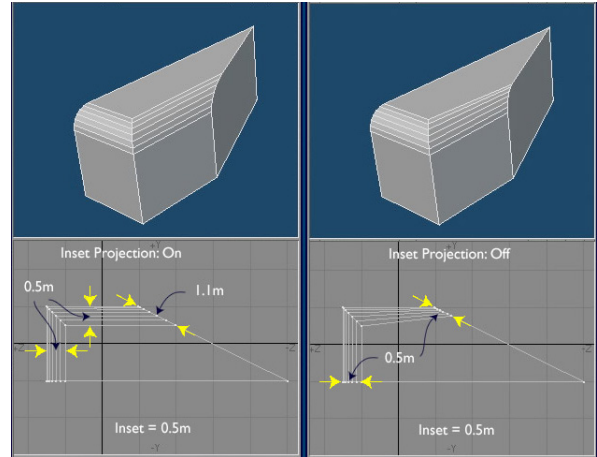
**Convexity** is illustrated below.



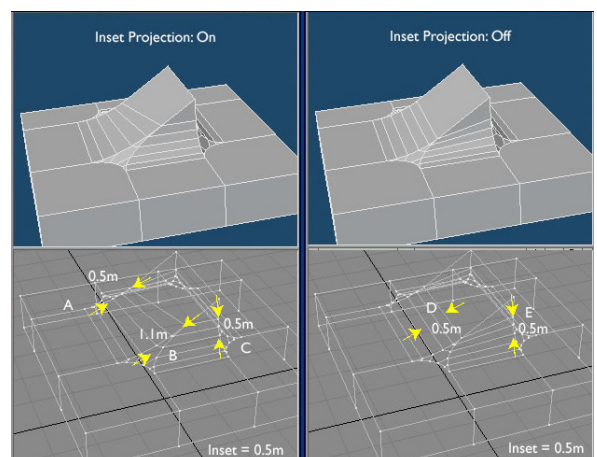
Rounding points will cause them to be “indented” as shown below.



**Project Inset** — This button controls how Rounder projects the inset at edges. This is illustrated below:

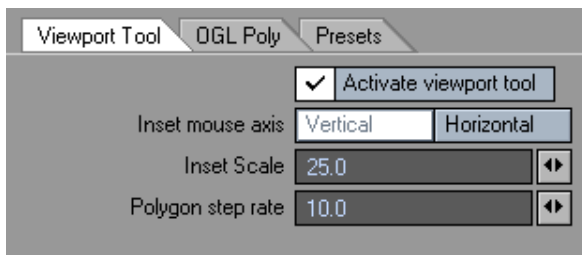


When the projection is on, the width of the rounded edge is equal to the inset distance (0.5m), but where the rounded edge intersects the side, the width is 1.1m — which it must be to keep the rounded width constant along the edges. If the inset projection is off, then where the rounded edge intersects the side the width is set equal to the inset distance — this results in a tapering rounded edge. Why have this option? Well, there are situations where the choice of the intersections widths is ambiguous and **Rounder** has to “guess” what you want. It may not “guess” correctly and in that case it is better to force **Rounder** to not use projections, as is illustrated for the “wedge” below:



**Viewport Tool Tab**





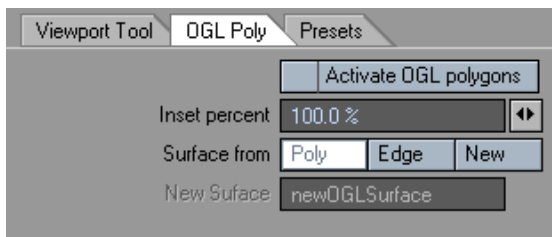
**Activate Viewport tool** — The **Viewport** tool allows you to drag the mouse in the viewport to interactively change either the inset distance or the number of rounding polygons by dragging either horizontally or vertically.

**Inset mouse axis** — This switch allows you to choose which axis controls the inset distance. If you want to drag horizontally to change the inset distance (and vertically for the number of rounding polygons), then choose **Horizontal**. Choose **Vertical** for the converse situation.

**Inset scale** — This controls the sensitivity of the mouse dragging when changing the inset distance.

**Polygon step rate** — This controls the sensitivity of the mouse dragging when changing the number of polygons.

## Open GL Polygon Tab



## OGL Polygons?

These are additional polygons added to remove OGL display defects. The OGL polygons are in the same plane as the “flat” part polygons adjacent to the rounding polygons. Roughly speaking it stops the OGL renderer smoothing the edge into the bulk of the object and keeps the smoothing to the rounder part of the object.

**Activate OGL polygons** — Switch on or off the OGL polygons.

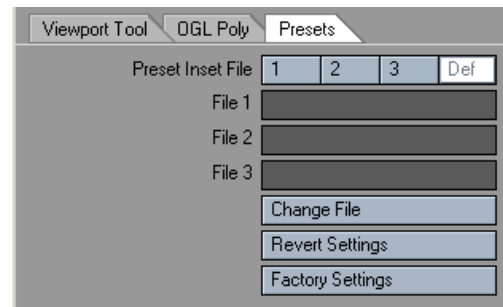
**Inset percent** — The size of the polygons is taken as a percentage of the inset distance.

**Surface from** — The surface name assigned to the OGL polygons can be:

1. **Poly** — the same as the original polygon (before rounding)
2. **Edge** — the same as that of the new rounding polygons (as determined by the main panel)
3. **New** — the name can be a new name as determined by the next control.

**New Surface** — The name of the new surface as chosen by the previous control.

## Presets Tab



All the settings on the whole panel are now saved in a file, called **RounderConfigs.txt**, and reloaded when Modeler is launched. You can “Revert” to the settings in that file or you can revert to the **Factory** (ie in-built) settings.

You can assign up to three different inset preset files to the three “File” boxes. This allows quick switching between different sets of inset presets - or you can use the default (in-built) set.

### More detail for the inset preset files:

1. You need to create inset preset files. You can create the file using a text editor with *one* number per line (with *no* units) . Create as many files as you like (eg one for mm steps, one for m steps) etc.
2. Click on one of the **Preset Inset File** numbers (eg “File 1”), then click on the **Change File** button.
3. Then navigate to one of your inset preset files and select it.
4. All the inset presets are now loaded in and should appear in the inset preset pop-up.
5. Repeat (if desired) for the other two cases.
6. Now, each time you click on a **Preset Inset File** number, all the preset insets in the associated file are loaded.

**Preset Inset File.** Select one of the four possible inset files to use with the **Inset Presets** pop-up menu on the main panel. **File 1**, **File 2** and **File 3** are the three presets files, while **Def** is the default built in set of inset presets.

**File 1** — The name of the **File 1** inset preset file is displayed here.

**File 2** — The name of the **File 2** inset preset file is displayed here.

**File 3** — The name of the **File 3** inset preset file is displayed here.

**Change file.** This button enables you to navigate to a text file containing a list of inset presets. First select which of the three files, **File 1**, **File 2** or **File 3** in the **Preset Inset File** row on controls, then click the **Change File** button and navigate to your preset inset file.

**Revert Settings** — Click on this button to change all the settings in the whole panel to those saved in the “RounderConfigs.txt” file.



**NOTE:** All the settings of the entire panel are saved to the “RounderConfigs.txt” file whenever you quit Modeler.

**Factory Settings** — Click this button to reset all the settings on the panel to the default built-in settings.



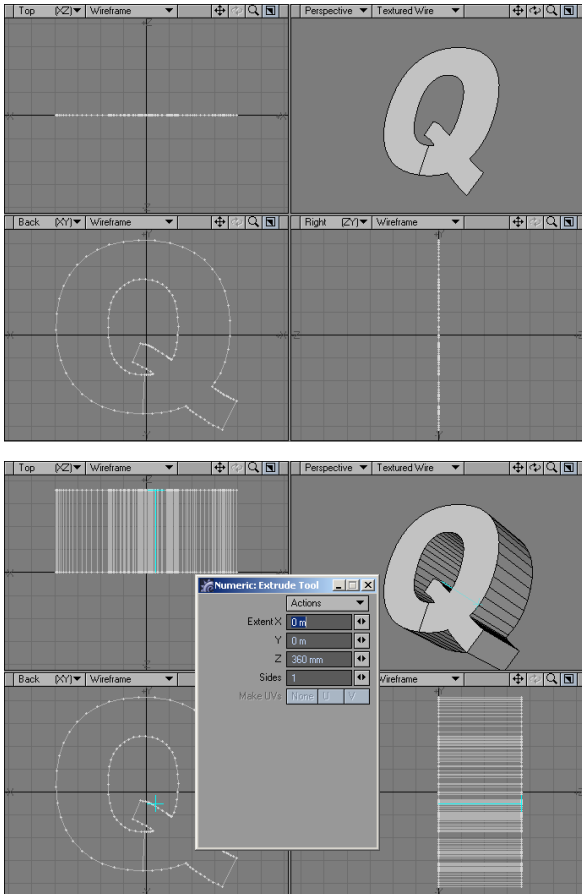
## Extrude

(default keyboard shortcut **Shift E**)

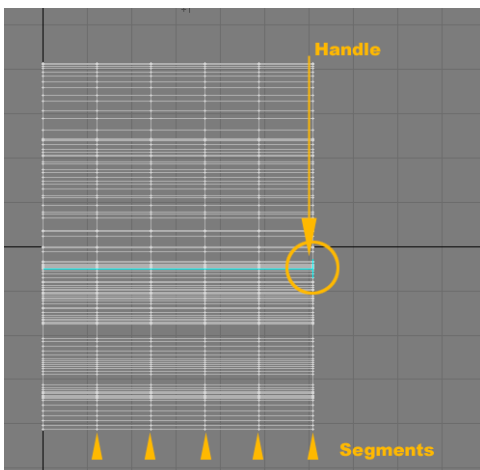
Use the **Extrude** tool (**Multiply > Extend: Extrude**) to give depth to a two-dimensional shape.

You can extrude any two-dimensional polygon or curve.

The distance and angle from the initial dragging point sets the extrusion length and direction. As you continue to drag your mouse, you see the extrusion length and angle change accordingly.



The initial dragging point is not significant, only the distance and direction matter. In fact, when you release the mouse button, the axis will reposition itself starting at the object. You can further manipulate the axis by dragging its end in any viewport.



When you are satisfied, simply click **Multiply > Extend: Extrude** again (or select another tool). To reset, click in a non-active part of the interface or click **Undo**.

The numeric panel for **Extrude** defines the relative XYZ coordinates for the end of the axis in the **Extent** fields. You can change the number of **Sides** created on the panel, which can be useful for multi-segmented items like snakes, or objects you wish to bend or twist.



### Make UVs Option

The **Make UVs** option at the bottom of the numeric panel assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.



**NOTE:** The **Extrude** tool treats the original polygon(s) as a *cap* on the end of the extruded shape. Therefore, to create an enclosed object, you should extrude backwards (ie. behind the original polygon(s)) for all the polygons to be facing outwards. Extruding forward in the direction of the polygons normal will create inverted polygons and your extrusion will appear to be inside-out. To correct this, simply select the affected polygons and then select **Detail > Polygons: Flip**.

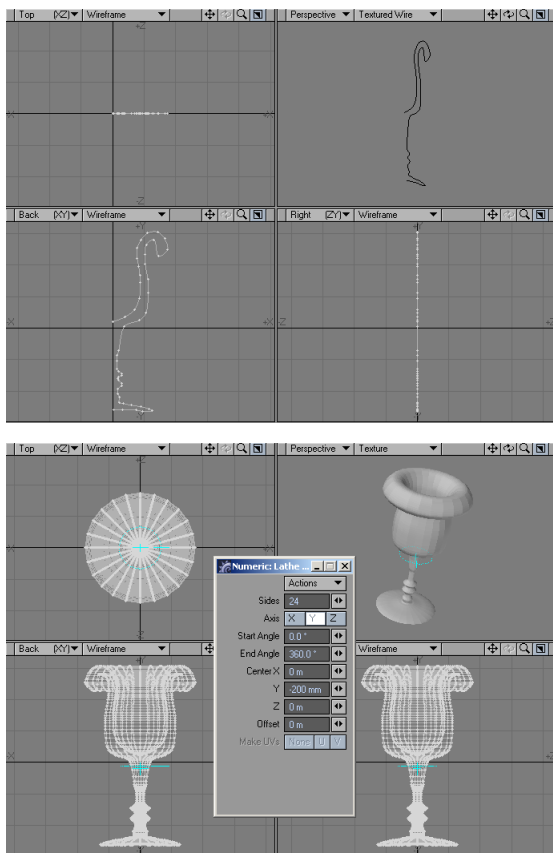


## Lathe Tool

(default keyboard shortcut **Shift** **L**)

The **Lathe** tool (**Multiply > Extend: Lathe**) will *sweep* an object around an axis. Essentially, **Lathe** creates a cylindrical object with a profile of the original form. For example, if you lathe a circular-shaped polygon, you create a doughnut. You can sweep any two-dimensional object or curve. The classic use of this tool is to create a vase or bottle.

Dragging your mouse in a viewport determines the initial axis. The lathe axis has a crosshair at its center (the initial dragging point), which is used to drag the entire axis around. To change the angle, drag either end of the axis. You can also drag out the axis, which gives you more control over the effect, but the length of the axis does not in itself influence the impact of the tool.

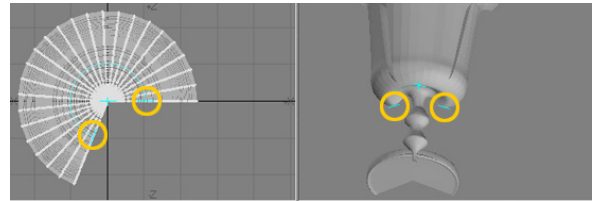


When you are satisfied, deselect the tool or select a different one to execute the edit. To cancel, simply click in an open area on the interface or click **Undo**.



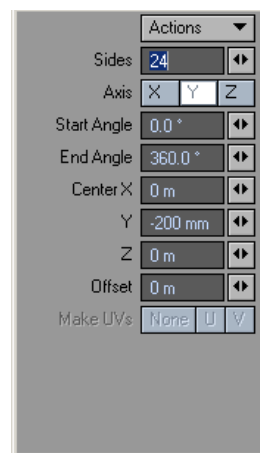
**NOTE:** With this tool, polygons may end up flipped the wrong way. To correct them, simply choose **Detail > Polygons: Flip**.

If you wish to change the number of sides in the lathed object (like the number of slices in a pie), you can use the Left and Right cursor keys. If you'd prefer to lathe less than a full 360 degrees, you can drag the handles as indicated in the following illustration. (Note: If the two handles are overlapped, you can still drag to separate them.)



### Numeric Options for Lathe

You can also open the **Lathe** tool's numeric requester to numerically change settings.



The **Start Angle** and **End Angle** fields let you enter the degree of lathing, **Sides** sets the number of sides, and **Offset** will shift the shape along the lathe axis. The **Center** values define the center of rotation coordinates.



**HINT:** To create a coiled spring: create a disc with, say, a one meter diameter. Select **Lathe** and place the axis about one meter to one side of the disc.

Then, open the numeric requester and set the **End Angle** to 720 degrees (*two full rotations*), **Sides** to 48, and **Offset** to 2m.



**HINT:** If you plan on using a high level of subpatching on your lathed object, eight sides will usually suffice for most round subpatched objects.

### Make UVs Option

The **Make UVs** option at the bottom of the numeric panel assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.



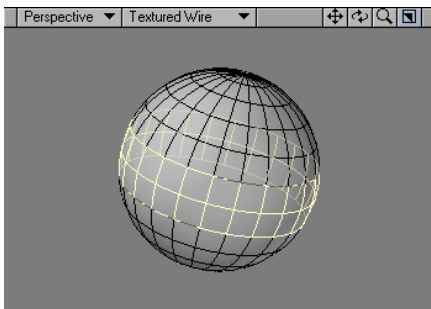
## Smooth Shift Tool

(default keyboard shortcut **Shift** F)

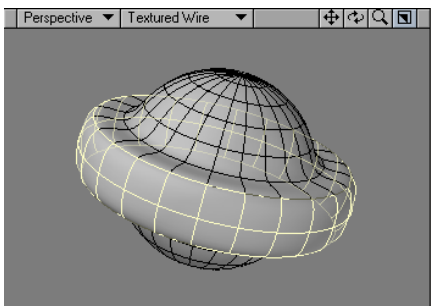
The **Smooth Shift** tool (**Multiply > Extend: Smooth Shift**) is a beguiling tool that may not seem useful at first; however, you will find a variety of uses for it, like creating thin grooves in an object. **Smooth Shift** moves selected polygons along their *smoothed* vertex normals. The smoothed vertex normal is computed using the average of the normals for the polygons that share that vertex. This tool is similar to **Bevel**; however, **Bevel** applies effects individually to each selected polygon, whereas **Smooth Shift** treats all selected polygons as a group.

### Steps to Smooth Shift a group of polygons

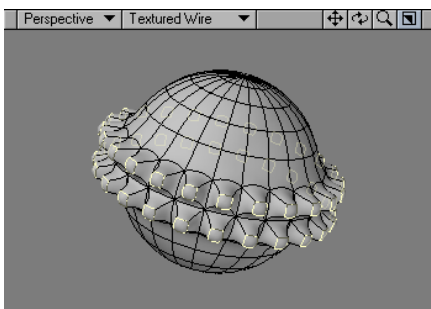
**Step 1:** Select the geometry you would like to extend.



**Step 2:** Choose **Multiply > Smooth Shift** and left click and drag.

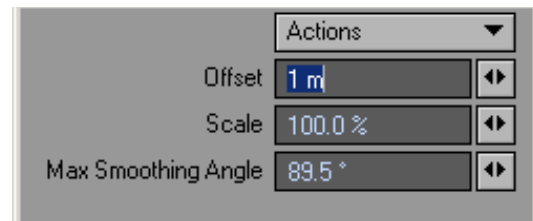


**NOTE:** Using **Bevel** would have resulted in this:



Polygons on a convex surface tend to expand since the normals at their vertices point slightly apart, while polygons on a concave surface tend to shrink. Polygons that are smoothed together stay together since they share normals at their shared vertices. Where polygons pull apart, either because their smoothed vertex

normals diverge and exceed the **Max Smoothing Angle** or because some are selected and some are not, the gaps are filled with bands of new polygons.



The **Offset** is the distance of shifting along the vertex normal. Positive values make the shift occur in the same direction as the normal. Negative values make it shift in the opposite direction.

The smoothing method in **Smooth Shift** requires a cutoff angle. The **Max Smoothing Angle** on the numeric panel determines whether polygons will be smoothed over. Any two polygons with adjoining edges at angles greater than this value will be smoothed across.

Dragging your mouse horizontally will change the **Offset** amount on the numeric panel.

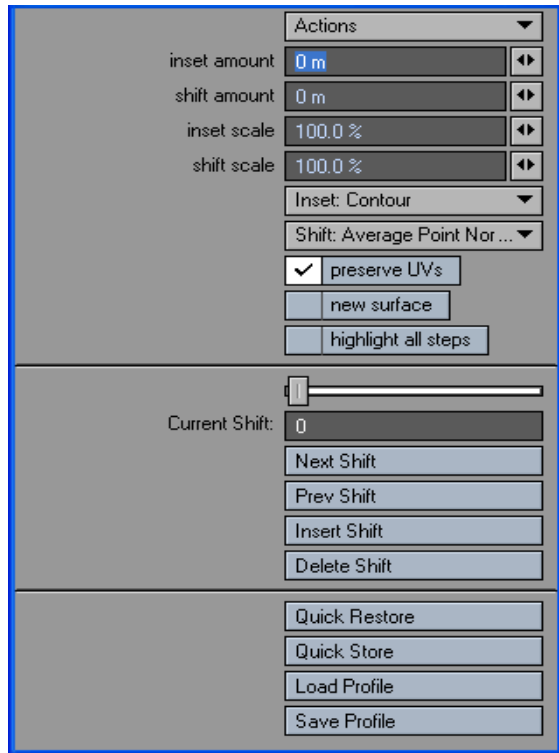
The **Scale** setting, accessible from the numeric panel only, expands or shrinks the new geometry around the center of its bounding (box) area.



**HINT:** To achieve what amounts to a group-bevel—where the selected polygons are treated as one instead of individually—use **Smooth Shift** with an Offset of 0. Then, manually move the polygons (which will still be selected).



## Multishift Tool



**Multishift** is an interactive tool similar to **Smooth Shift**, which allows you to inset and shift groups of polygons together. Unlike **Smooth Shift**, this tool lets you interactively inset the selected polygons by dragging the mouse left/right. Furthermore, the inset algorithm used can be chosen along with the shifting algorithm through the numeric panel.

Steps for using Multishift on selected geometry:

- Step 1- Select the geometry you would like to extend
- Step 2- On the Multiply>Extend Menu, choose Multishift.
- Step 3- Open the numeric panel, and toggle on Multishift Mode if it is not already activated.
- Step 4- Left-click and drag to move the selected polygons.
- Step 5- Right-click to create a new shift.

Unlike **Smooth Shift**, an average normal shifting algorithm is provided to shift groups of polygons along their average normals rather than along a local normal. Some users find that shifting along an average normal is often easier to control, as it doesn't deflate/inflate the selection as it grows inward/outward.

**Inset Amount**—The movement of the polygons towards the inner selection. Several new algorithms have been introduced for greater flexibility.

**Shift Amount**—The movement of the polygons outwards along the polygon normals.

**Inset Scale**—Controls the scale of inset during a single operation

**Shift Scale**—Controls the scale of shift during a single operation

## Inset Menu

Inset: Contour  
Inset: Relaxed Contour  
Inset: Hybrid  
Inset: Flat  
Inset: Inner Edge  
Inset: Regular Scale

**Inset: Contour**—A general inset algorithm which is intended to maintain the natural outer contours of the selection.

**Inset: Relaxed Contour**—Works like **Inset: Contour**, with a different set of rules for sharp edges. Points on an edge with an acute angle, say about 15 degrees or less, when inset will begin to move closer together faster than those on an edge with a wider angle. **Inset: Relaxed Contour** will relax these tight angles, so the points will not move closer or overlap as quickly.

**Inset: Hybrid**—A combination of the **Inset: Flat** and **Inset: Inner Edge** algorithms. When the geometry is generally flat, meaning the normals do not vary much from polygon to polygon, it will use the rules for **Inset: Flat**. When the geometry is not flat it will use the rules for **Inset: Inner Edge**.

**Inset: Flat**—A 2D shape algorithm which uses rules for 2D methods for determining the direction of moving inwards and outwards. Works best with a group of flat polygons.



**Warning:** Since the algorithm works by projecting on a 2D plane, **Inset: Flat** will likely fail and move points in the wrong direction if the geometry is not generally flat.



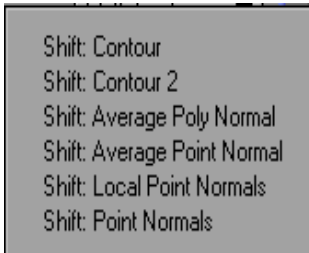
**Hint:** **Inset: Flat** is ideal for beveling flat shapes like text!

**Inset: Inner Edge**—The original algorithm for Multishift. Works best if used with quadrangles that are laid out in a grid-like fashion.

**Inset: Regular Scale**—This algorithm moves the new points toward the center of the selection.



## Shift Menu



**Shift: Contour**—Similar to the algorithm for Inset: Contour, the outward points along the normals are moved in a way that tries to preserve the original contours of the geometry.

**Shift: Contour 2**—A slight variant of Shift: Contour, in the majority of cases should give the same result, but will in some cases give a better result and in others a worse result.

**Shift: Average Poly Normal**—A basic algorithm which shifts the group of selected polygons along the averaged normal of each polygon within the group.

**Shift: Average Point Normal**—A basic algorithm which averages the normals of all points making up the selected polygons and moves the newly created polygons along those normals.

**Shift: Local Point Normal**—Moves each point along its normal.



Hint: Shift: Local Point Normal tends to be more useful in organic modeling.



Note: Unlike Shift: Point Normal, Shift: Local Point Normals will not include unselected polygons in the computation of each point normal.

**Shift: Point Normal**—Moves each point along its normal, but also includes unselected polygons in the calculation.



Note: None of the above algorithms are designed to work in all situations, although the contour algorithms are designed to be general-purpose and should work in most situations. Like any tool it is best to experiment and see which one works best for the situation.

**Preserve UVs**—Will create UV Maps for the newly created polygons. The new polygons are copied to the UV Map but not inset or shifted.

**New Surface**—Creates a new surface for the new polygons in a shift. You can create a new surface name or use a previously created surface.

**Highlight All Steps**--When checked, highlights each shift in a work session. Unchecked, only the current shift is highlighted.

## Multishift Edit Controls

**Current Shift Slider**—Adjusts the current active shift when two or more shifts are available

**Next Shift**—Selects the next shift

**Prev Shift**—Selects the previous shift

**Insert Shift**—Creates a new shift after the currently selected shift. The new shift will now be active and created halfway between the previous and next shift.

**Delete Shift**—Deletes the current shift, only if two or more shifts exist in the operation. After deletion the next shift is then selected as active.

**Multishift Mode**— When checked, Multishift Mode is activated and will record shifts and allow the user to edit and save the profile. A Right Mouse Click will insert a new shift to the operation, just like Insert Shift will do.



Note: Right-clicking with Multishift Mode Off will only save the last shift when you choose Quick Save or Save Profile. Each Insert Shift, however, will be saved in the history of the current session.

## Save Options

**Quick Restore**—The profile for the last “Quick Store” operation is activated.

**Quick Store**—Stores the current active operation profile for use with “Quick Restore”.

**Load Profile**—Loads a saved profile

**Save Profile**—Saves the current active operation to a specific file





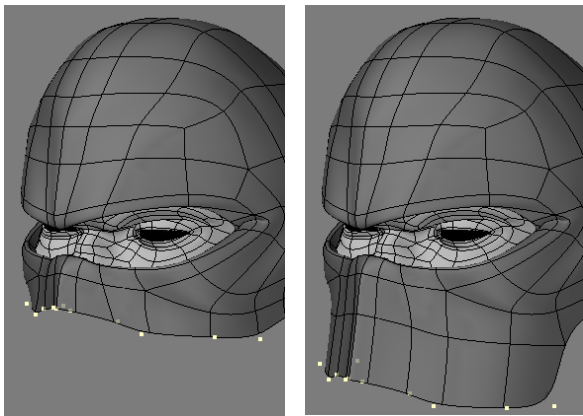
## Extender Plus

(default keyboard shortcut **E**)

**Extender Plus** (**Multiply > Extend: Extender Plus**) clones selected points and creates new geometry connecting the original points and the clones. Essentially, **Extender Plus** is like using **Smooth Shift** with 0 Offset, except with points instead of polygons. **Extender Plus** is an updated version of **Extender** and works better in many areas.

**Extender Plus** works in both **Point** and **Polygon** modes.

**Point Mode:** Point Mode allows you to select an unordered group of points, automatically reordering the polygons it extends to face the proper normals. Furthermore, it detects when it should loop and when it shouldn't, unlike **Extender**, which always loops.

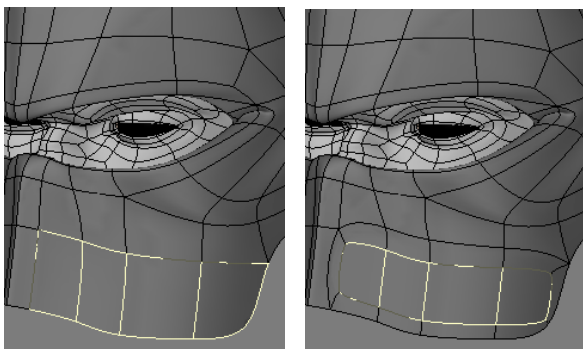


Furthermore, it uses a simpler but more efficient algorithm of determining the normals of each edge's connected polygon when creating each extended polygon as opposed to reordering the entire point selection as a whole. As a result, you don't have to select the first two points in the proper clockwise order. The entire selection can be selected with a lasso without problems. The tool works particularly well with the **Select Loop** tool when used to select borders.



**NOTE:** **Extender Plus** works in **Symmetry Mode** as well as **SubPatch Mode**, unlike **Extender**.

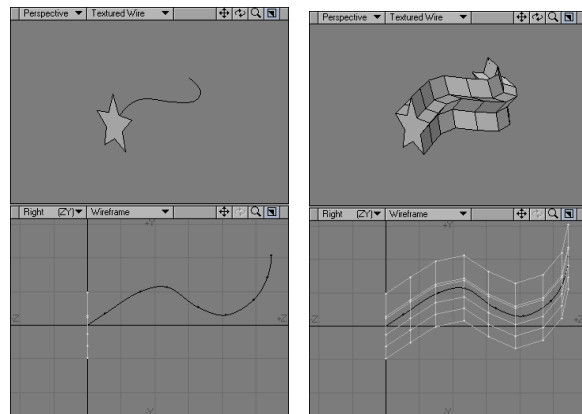
**Poly Mode:** In **Poly Mode**, **Extender Plus** performs a quick grouped polygon extrusion without providing an interactive interface. It can be a quick way to extrude while having an interactive transformation tool active such as **Scale/Move** tool as opposed to using **Smooth Shift** or **Super Shift**, dropping it, and then reactivating transformation tools.



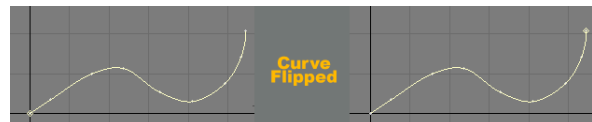
## Rail Extrude Tool

The **Rail Extrude** command (**Multiply > Extend: Rail Extrude**) will extrude a point, polygon, or object along a curved path (single rail clone) or multiple paths (multiple rail clone). It is nearly identical to the **Rail Clone** command, discussed later. The difference is that **Rail Extrude** forms a continuous skin for the object instead of making individual copies of the object.

The curve(s) must be in the background layer and the polygons must be extruded in the foreground layer. The orientation and position of the foreground polygons will affect the result. Generally, they should be at the very beginning of the curve and oriented perpendicular to it—as if the curve was a thread you were threading through the polygons.

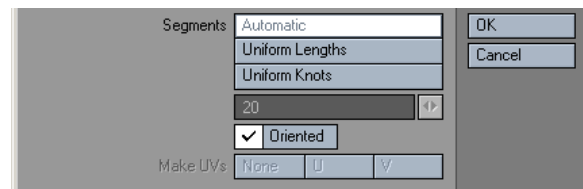


The direction that the curve faces (which end is considered the beginning) will have bearing on both the shape and direction of the cloning. The direction affects the direction of the extruding. If you don't get the results you expect, try flipping the curve's starting point with **Detail > Flip**. If you are using multiple curves, they should generally run in the same direction.



### Single Curve

The **Single** dialog will appear when a single curve is in the Background layer.



**Segments:** the number of segments in the resulting object. Select **Automatic** to distribute a number of segments along the rail curve (based on the **Curve Divisions** setting on the **General Options** panel) according to the curve's knot (point) spacing.

Select **Uniform Lengths** to distribute some number of segments (which you specify) evenly along the length of the entire curve. No matter what the knot spacing is, the segments will remain evenly spaced.

Select **Uniform Knots** to distribute some number of segments (which you specify) evenly between the knots that make up the

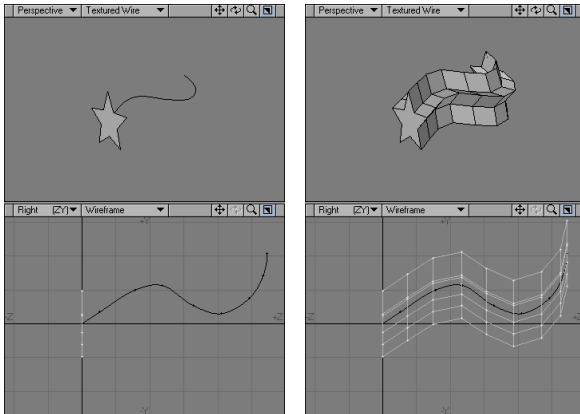




curve. Whether the knots are close together or far apart, there will be the same number of segments between each.

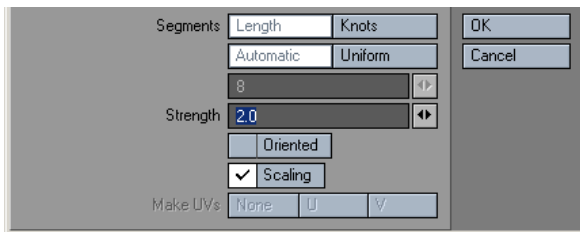
**Oriented** determines the orientation of the polygon template as it follows the curved path. When active, it causes all segments to angle themselves automatically so that they are aligned with the rail curve as it turns. When inactive, it causes all segments to remain in the same orientation as the template so that they face exactly the same way.

If a **UV Texture** is selected, the **Make UVs** options are available. You can create the **UV Map** perpendicular to the U or V axis. You can have the U or V value assigned to a range from 0 to 1. Note that a **UV Texture** must be selected or this option will be ghosted.



## Multiple Curves

The **Multiply** dialog will appear when multiple curves are in the background.



In practice, one curve will act as the main cloning curve and the other(s) act as shaping curves. The effect of the shaping curves is determined by the distance from the main curve. However, you may change the effect by moving the polygons closer or farther away from any of the curves.

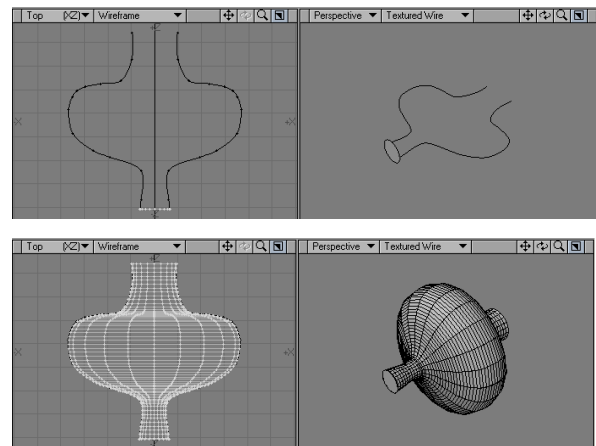
**Segments** is the number of segments in the resulting object. Select **Automatic** to distribute a number of segments along the rail curve, based on the **Curve Divisions** setting on the **General Options** panel. If you also select **Knot**, segments are distributed with regard to the curve's knot (point) spacing. If you use **Length**, knot spacing is disregarded.

To specify the number of segments, choose **Uniform** and enter the number in the input field. If **Length** is also active, the segments will remain evenly spaced no matter what the knot spacing is. If you use **Knot** instead, there will be the same number of segments between each knot, no matter how close together or far apart they are.

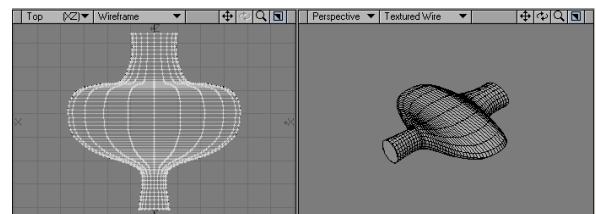
**Strength** determines how strongly the rails vie for control of the point locations. You will not notice much of a difference unless you have **Scaling** (below) deselected. The higher the **Strength** value, the closer the object will *hug* the rails.

**Oriented** determines the orientation of the polygon template as it follows the curved path. When active, it causes all segments to angle themselves automatically so that they are aligned with the rail curve as it turns. When inactive, it causes all segments to remain in the same orientation as the template so that they face exactly the same way.

**Scaling** determines how segments along an axis are sized. If the rails in the background layer spread apart along any axis or axes, then the segments will automatically stretch along those axes as well. Turn **Scaling** on to scale segments equally, rather than along those specific axes only.



For example, with **Scaling** off, if you **Rail Extrude** a polygon along the Z axis using two rails that spread further apart in the X axis, the segment's copies will be stretched on the X axis to maintain their relationship to the guide rails.



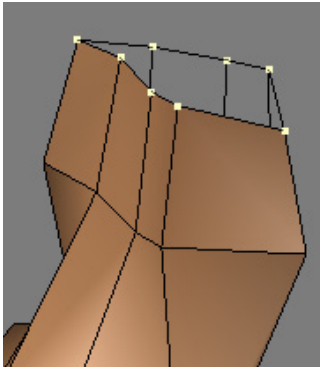


## Extender Tool

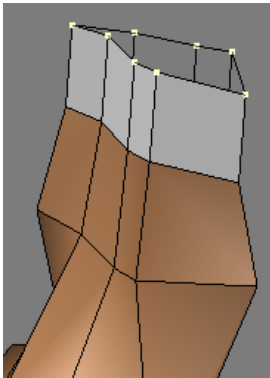
This tool has been superseded by **Extender Plus**. The **Extender** command (**Multiply > Extend: More > Extender**) clones selected points and creates new geometry connecting the original points and the clones. Essentially, **Extender** is like using **Smooth Shift** with 0 Offset, except with points instead of polygons.

### Steps for extending points (edges)

**Step 1:** Select the points that make up the edge you would like to extend. Remember to select the points in order and counterclockwise.



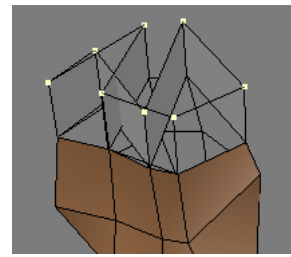
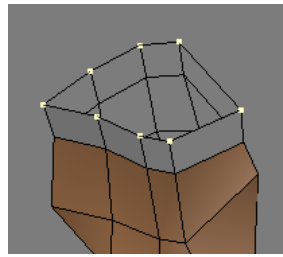
**Step 2:** Choose **Multiply > Extend: More > Extender**, then move the selected points away from original points.



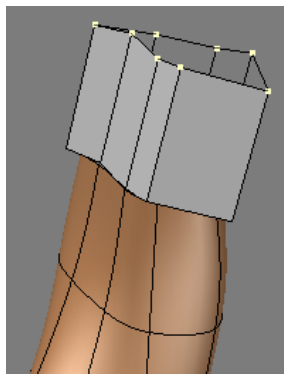
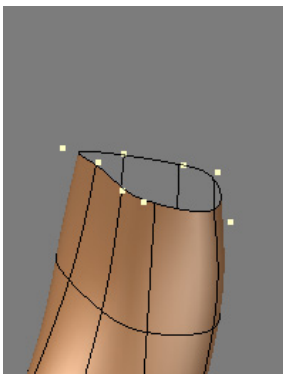
NOTE: In **Subpatch** mode, newly created geometry will be in **Polygonal** mode. **Extender Plus** doesn't have this limitation.



NOTE: Selecting the points out of order may have unwanted results.



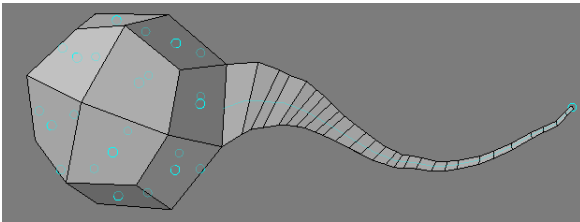
Left: Points selected in order, counter clockwise. Right: Same points selected randomly.





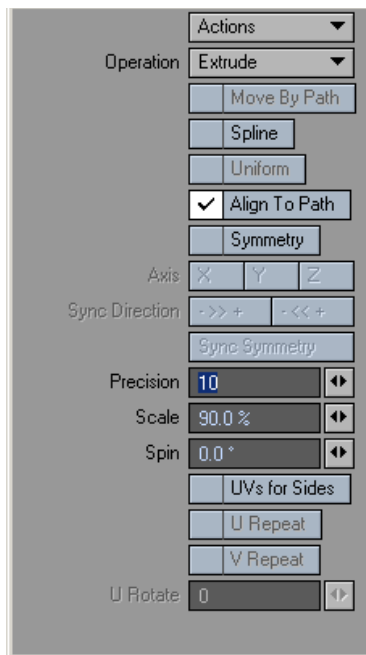
## Magic Bevel Tool

The **Magic Bevel** tool (**Multiply > Extend: More > Magic Bevel**) lets you interactively drag out a bevel along an editable path with tremendous control over how the bevels are created.



NOTE: This tool works on polygons, SubPatches, one-point polygons, two-point polygons, and Metaballs.

You create the bevels by dragging any of the blue circle handles. All polygons will get a handle regardless of what is selected. The path will be drawn along the vertical and horizontal axes of the viewport, so use a Perspective viewport to bevel off-axis.

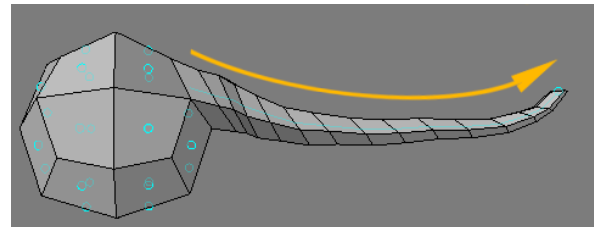
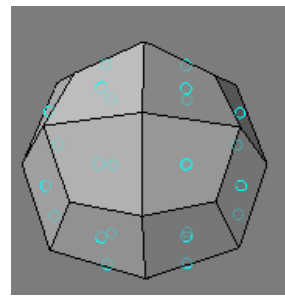


### Operation:

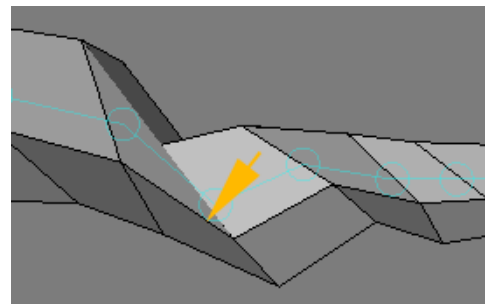
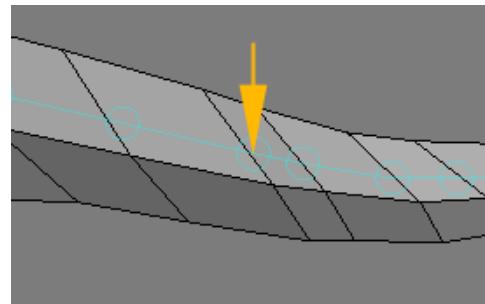
**Magic Bevel** offers the following operations:

**Extrude** — **Extrude** is selected by default and is the mode that initiates the bevel creation.

Simply left click on one of the knots and drag out new geometry.

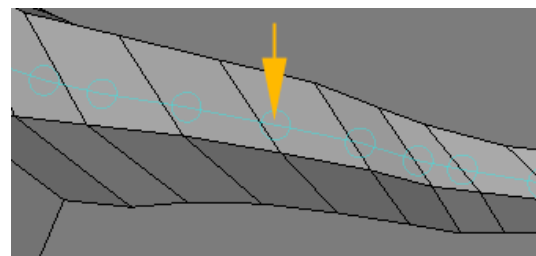


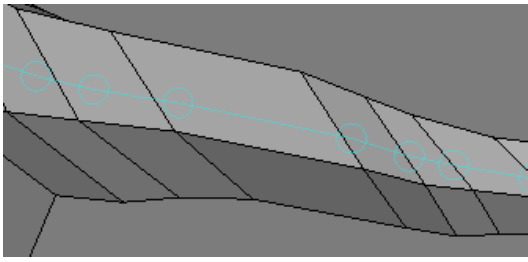
**Edit Path** — If you activate **Edit Path**, drag handles will appear at each bevel. You may drag them to change the positioning of each bevel.



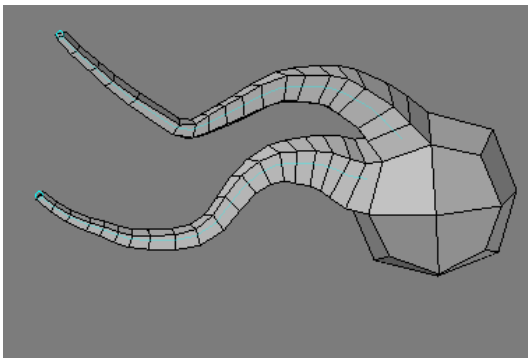
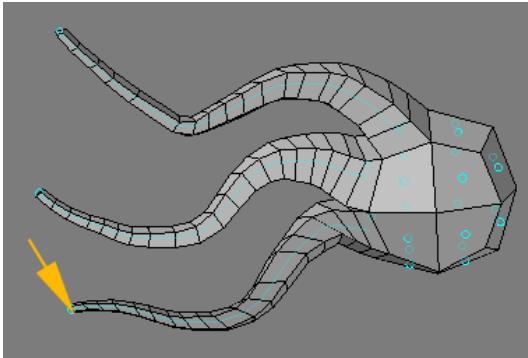
**Delete Knot** — If you activate **Delete Knot**, you can interactively remove knots from the path.

Simply Left-click on the knot you would like to remove.

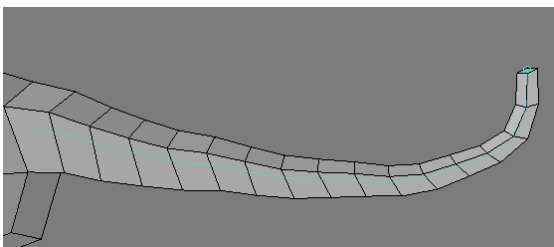
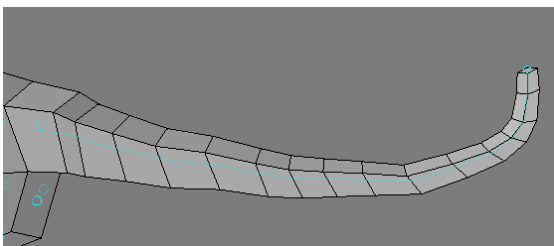




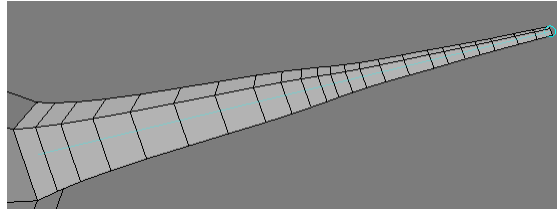
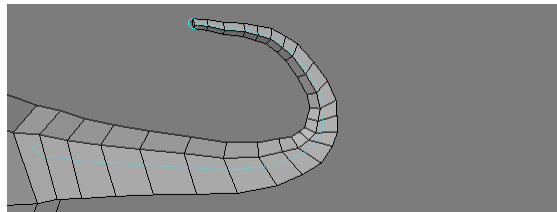
**Delete Path** — Activating **Delete Path** will give you the ability to remove an entire path.  
Left Click on the knot at the end of the path to remove it.



**Uniform Spans** — Applying **Uniform Spans** to a magic bevel path will evenly space each segment in the path.



**Straight** — Activating **Straight** and clicking on the end of a path will align the entire path on the polygons normal.



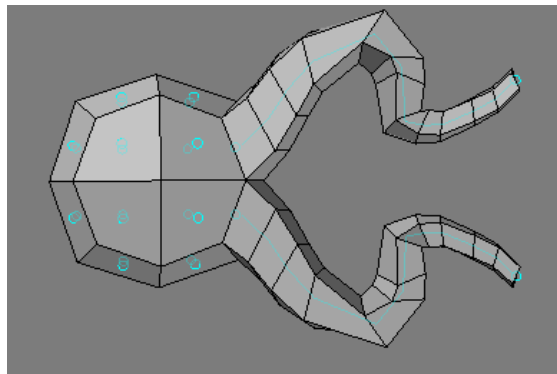
**Move by Path** — When **Edit Path** is active, you may also activate **Move By Path**. This option moves all of the bevel positions, from the current edit up to the end of the path, together at the same time.. When this is off, only the selected point moves.

**Spline** — The **Spline** option turns the path into a smooth spline path. This will change the bevel positions slightly.

**Uniform** — The **Uniform** option causes the bevels to occur evenly along the path. You will notice that the bevels will no longer occur at each edit handle. This only works when using the **Spline** option.

**Align to Path** — The **Align to Path** option aligns the bevel polygons with the tangent of the path. When this is off, the angle of the bevels matches the starting polygon.

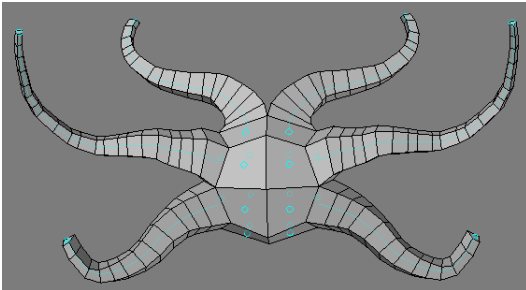
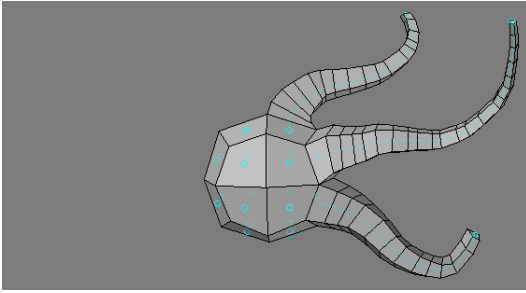
### Symmetry:



**Axis** — Choose the X,Y, or Z axis to determine which axis symmetry to work from.

**Sync Direction** — **Sync Direction** allows you to decide in which direction the sync will occur.

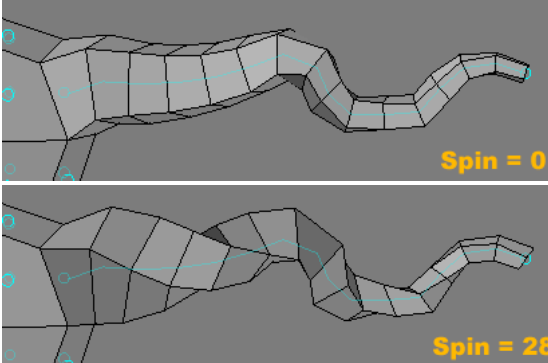
**Sync Symmetry** — Pressing **Sync Symmetry** will update the geometry in the **Sync Direction** to match the other half.



**Precision** — The **Precision** value sets mouse sensitivity and determines how closely the bevels follow the path. When your mouse pointer moves this many pixels, a bevel is created. Higher values are less precise and result in longer segments.

**Scale** — The **Scale** value sets how much to size down (or up) each successive bevel.

**Spin** — This setting will determine by how many degrees each new bevel will rotate as it is created.



**UVs for Sides** — The **UVs for Sides** option at the bottom of the numeric panel assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.

**U Repeat** — All geometry created will share the same U coordinates.

**V Repeat** — All geometry created will share the same V coordinates.

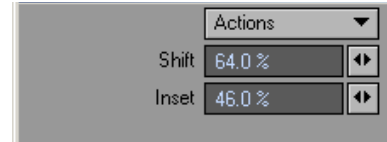
**U Rotate** — Adjusting **U Rotate** will shift the U coordinates of the map.



NOTE: If you click your **RMB** after you create a bevel path, blue handles will appear on the new bevel geometry.

## Rail Bevel Tool

Rail Bevel (**Multiply > Extend: More > Rail Bevel**) is a very fancy bevelling tool that works in a similar manner to the normal **Bevel** tool. Dragging your mouse left/right adjusts the new bevel geometry's distance from the original polygon. Dragging up/down adjusts the *inset* direction (towards or away from the polygon's center). The **Ctrl** key constrains the change to a given degree of freedom — just like the normal **Bevel** tool.



The difference between **Rail Bevel** and **Bevel** is that **Rail Bevel** uses a *profile* from a background layer to generate several bevel operations in a single step.

The profile is a series of points you create. They can be unconnected, or part of a polygon or curve. The only important factor is point order, which is normally determined by their order of creation.

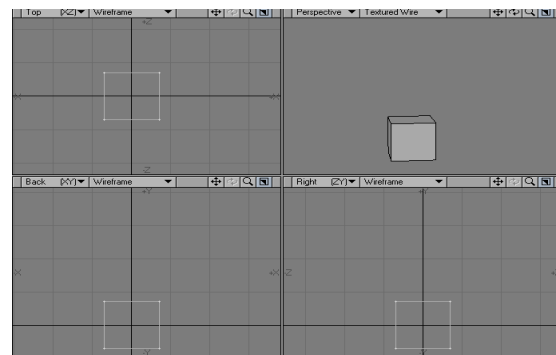
The profile is treated as if it were oriented based on the polygon's plane and extrudes along the direction of the polygon's normal.



Note: Each point in the profile is treated as a *bevel*. X=Inset, while Y=Length.

### Steps for using Rail Bevel

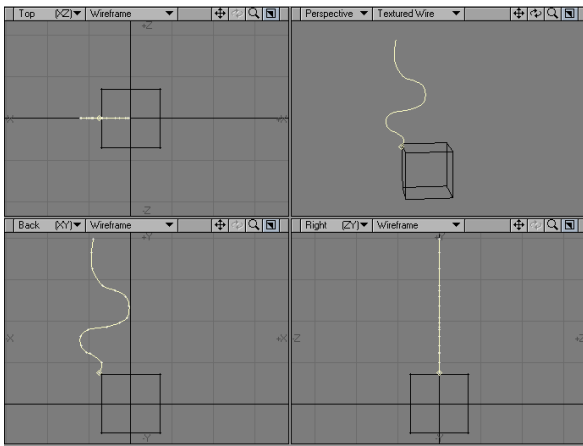
**Step 1:** Create some geometry in a layer that you plan on extending.



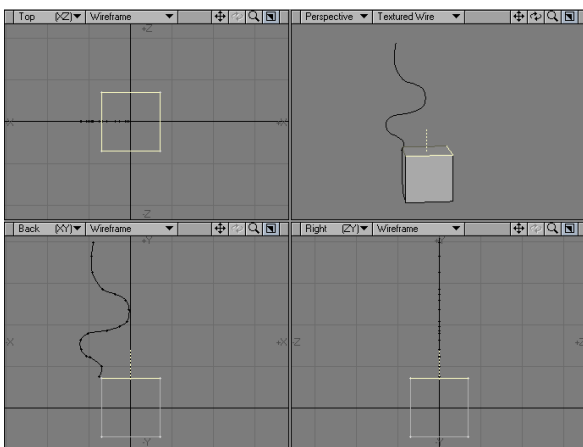
**Step 2:** In a new layer create a curve with the profile you would like to use for your bevel.



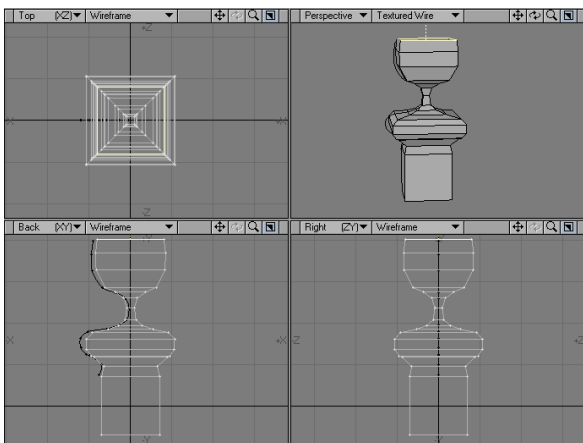
WARNING: Your profile **must** be created in the Back/Front viewport to work.



**Step 3:** Place the curve in the Background layer and your object in the Foreground layer. Select the geometry you would like to bevel.



**Step 4:** Click and drag to create your bevel. Dragging your mouse left/right adjusts the new bevel geometry's distance from the original polygon. Dragging up/down adjusts the *inset* direction.



**Step 5:** Press the **Enter** key to confirm.



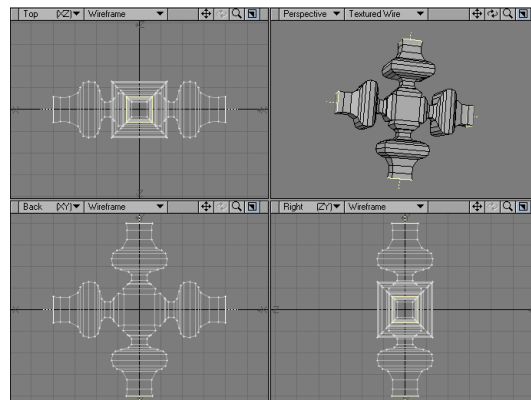
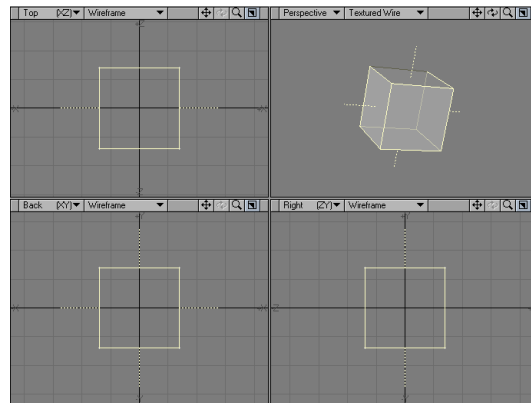
NOTE: The placement of the curve has no effect on the final outcome.



NOTE: New geometry inherits **VMap** and face/curve properties of the original polygon.



NOTE: **Rail Bevel** works with multiple polygons selected.

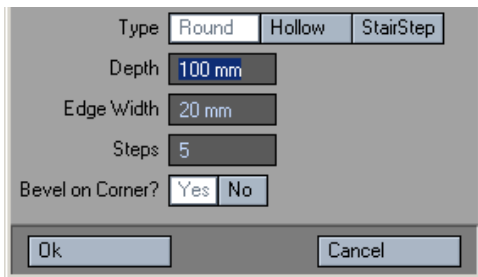


HINT: If you have multiple profiles in different layers, you can apply them one after another by selecting them (one at a time) as the background layer while using **Rail Bevel**.



## Router Tool

The **Router** tool (**Multiply > Extend: More > Router**) will create custom bevelled Edges. The numeric window will come up as soon as you select **Router** and will give you several options to work with.

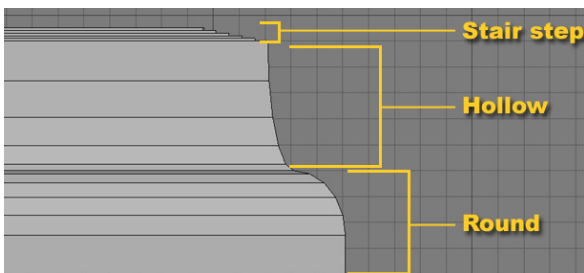


**Router** has three bevel types to choose from: **Round**, **Hollow**, and **Stair Step**.

**Round:** This option will create a “convex” curve profile.

**Hollow:** This option will create a “concave” curve profile.

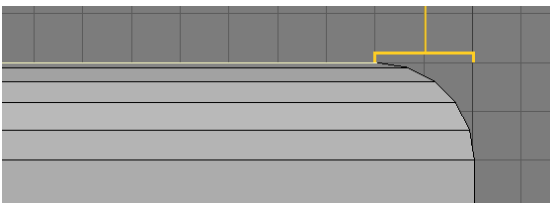
**Stair Step:** This option will cascade the newly created geometry.



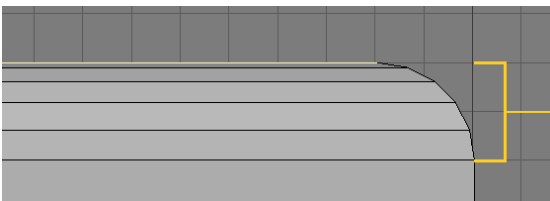
**Depth** — Adjusting the **Depth** setting changes the Height of the newly created geometry.



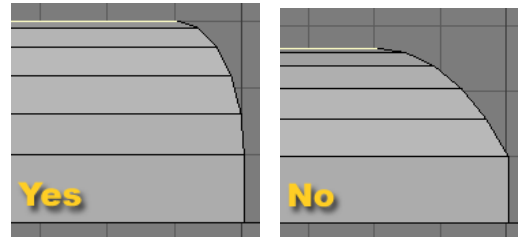
**Edge Width** — **Edge Width** controls the Inset of the bevel.



**Steps:** Adjusting the **Steps** setting will change the number of segments created.



**Bevel on Corner?:** Choosing **Yes** (Default) will constrain the bevel curve to the **Depth** and **Edge Width** settings. Choosing **No** will use the **Depth** divided by the number of **Steps** to configure the curve along the edge.







## Motion Path Extrude Tool

**Motion Path Extrude (Multiply > Extend: More > Motion Path Extrude)** will extrude geometry along a motion path saved out of Layout.



**First:** This option defines the first frame in the motion file that will be used.

**Last:** This option defines the last frame in the motion file that will be used.

**Step:** This option determines how many segments will be used in the extrusion, by setting a sampling ratio relative to the number of frames of the motion path. The lower the number, the more segments.

Step Examples:

First Frame 1, Last Frame 60

Step 0.5 - 120 Segments

Step 1 - 60 Segments

Step 2 - 30 Segments

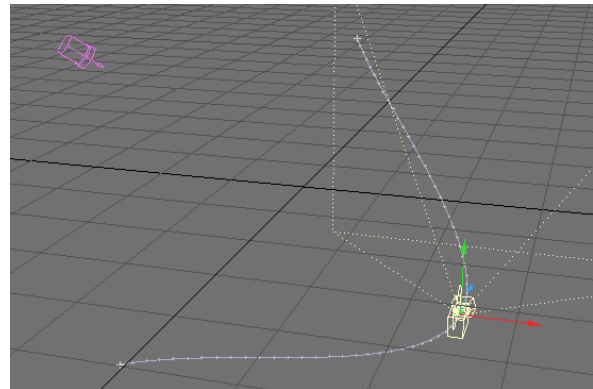
Step 3 - 20 Segments

Make UVs Option

The **Make UVs** option at the bottom of the numeric panel assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.

### Steps to extrude along a motion path:

**Step 1:** Create a motion path for an Item in Layout.

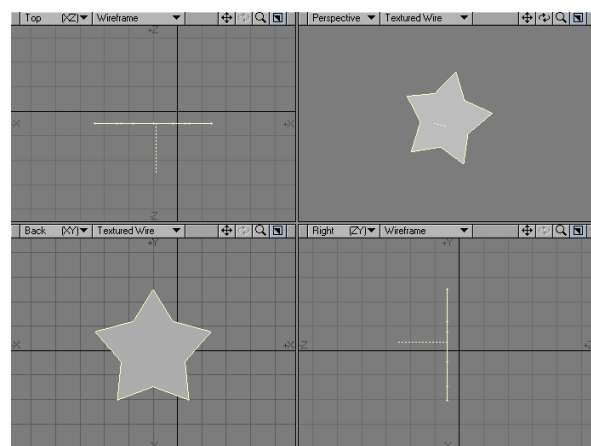


**Step 2:** With the item still selected go to **File/Save/Save motion file**.



NOTE: Be sure the file has the .mot extension.

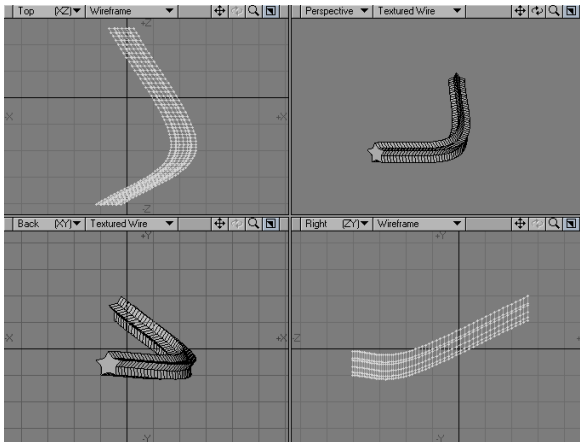
**Step 3:** Open the Modeler and create an object.



**Step 4:** Select **Motion Path Extrude (Multiply > Extend: More > Motion Path Extrude)**. Select the path saved above and change settings in the numeric window as needed.

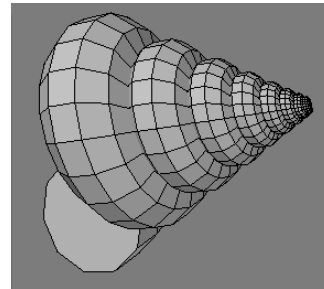


**Step 5:** Click OK.



## Seashell Tool

The **Sea Shell** tool (**Multiply > Extend: More > Sea Shell**) lets you interactively make seashell shapes. Essentially, this is like using the **Lathe** with a scaling factor on an object.



Open the numeric panel to set options.

**Axis** is the perpendicular axis around which the shell is twisted.

**# of Loops** is the number of times the source polygon is twisted fully.

**Sides per Loop** is the number of segments to use per loop.

**Shift per Loop** controls the vertical shifting of the loop.

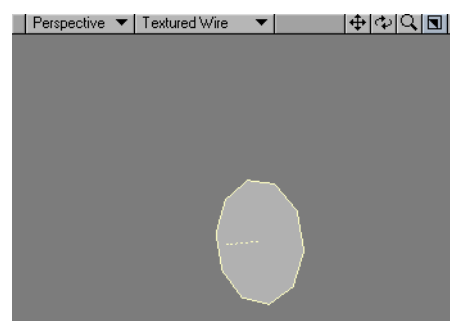
**Scale per Loop** is the scaling factor achieved after each loop.

### Make UVs Option

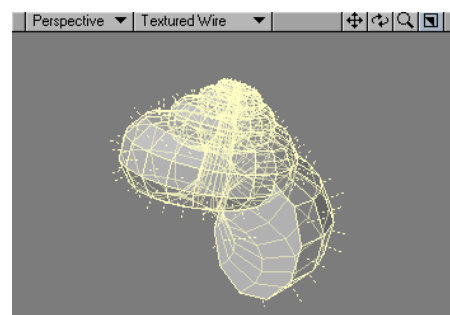
The **Make UVs** option at the bottom of the numeric panel assigns some default UVs based on the geometry of the object. Note that a **UV Texture Map** must be currently selected or this option will be ghosted.

### Steps for using the Seashell tool:

**Step 1:** Create a flat polygon that you would like to apply **Seashell** to.

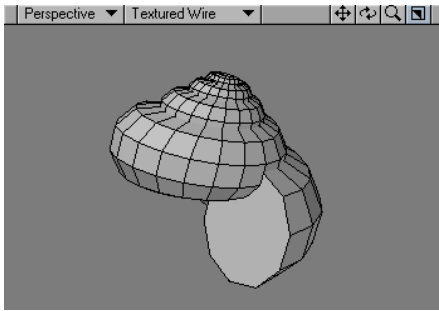


**Step 2:** Choose **Multiply > Seashell** and set the options to your liking.





**Step 3:** Press **Enter** on the keyboard.

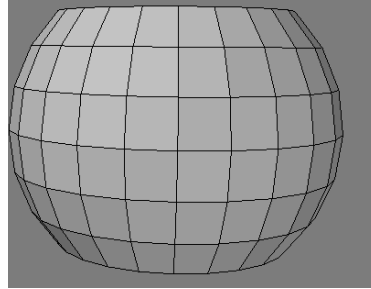


## Spikey Tool

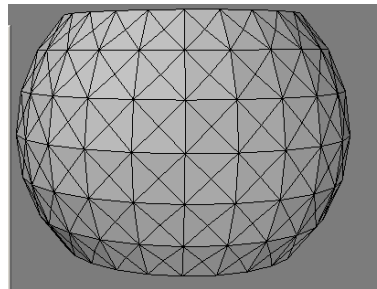
The **Spikey** tool (**Multiply > Extend: More > Spikey**) subdivides the selected polygons and moves the center point out in the direction of the surface normal giving a spiked appearance. Simply select the tool and drag your mouse pointer in a viewport.

### Steps for using Spikey:

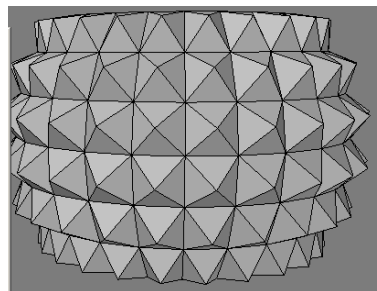
**Step 1:** Create an object that you would like to use **Spikey** on.



**Step 2:** Choose **Multiply > Spikey**, and open the numeric window.



**Step 3:** Left click and drag in the viewport or use the **Spike Factor** input field on the numeric window to adjust the spikes.



**Step 4:** Press **Enter** to confirm the changes.



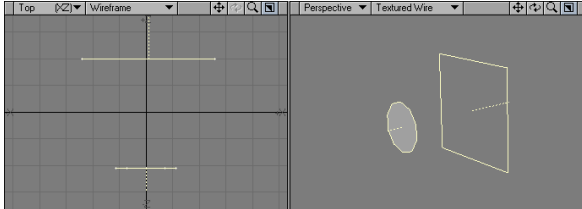
NOTE: Using Spikey with a Spike Factor of 0 is a good way to subdivide a polygon for use in ClothFX.



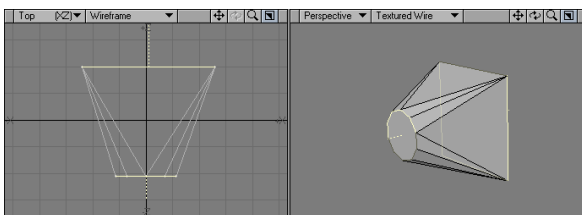
## Create Skin

With the **Create Skin** command (**Multiply > Extend: More > Create Skin**), you can cover a series of polygons or curves with a polygon *skin*. The shapes do not need to have the same number of points in common. This is also called *lofting*.

**Step 1:** Select the geometry to be skinned.



**Step 2:** Choose **Multiply > Extend: More > Create Skin**.



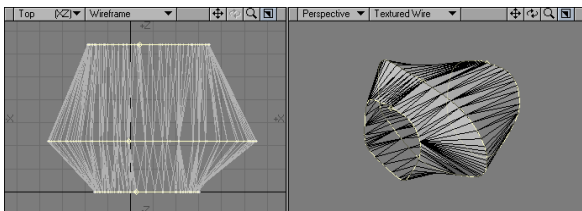
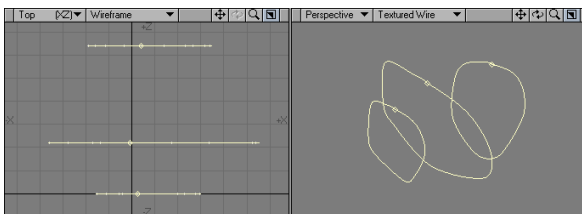
NOTE: The polygons or curves must be selected (or created) in the order that you want them connected.



NOTE: This operation can only operate with elements from the same layer.

**Create Skin** works best when you use single-sided polygons for the framework. Double-sided polygons can result in strangely skinned objects.

If you skin using curves, the curves will remain in the Modeler workspace following the operation. However, they do not become part of the resulting skinned object.



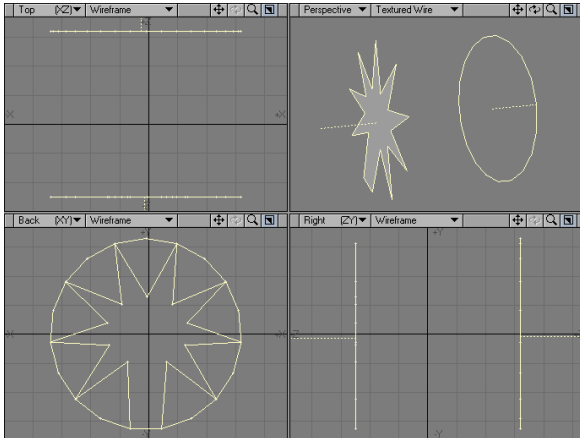
NOTE: Although **Create Skin** is similar to the **Make Spline Patch** command (**Construct > Patch**), which uses curves, the difference between the two commands is significant. The **Make Spline Patch** command tends to follow smooth contours, which maintains a rounded surface. **Create Skin** tends to connect curves with straight-line segments, which creates less smooth profiles.



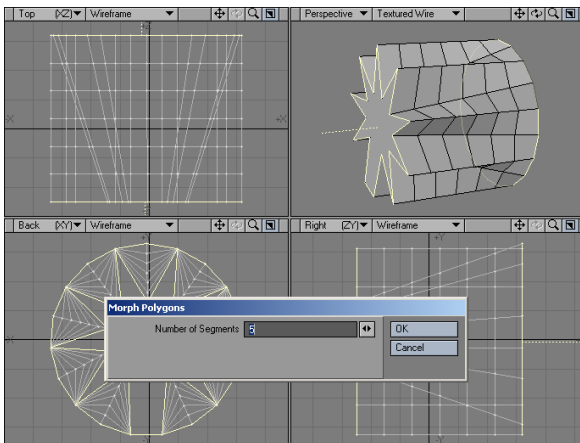
## Morph Polygons Tool

The **Morph Polygons** command (**Multiply > Extend: More > Morph Polygons**) will create a number of connected intermediate polygons between two *selected* polygons or curves that have the same number of points. They will have an outer surface, as if you also used the **Skin** command on them.

**Step 1:** Select the two polygons/curves that you would like to skin.



**Step 2:** Choose the **Morph Polygons** command (**Multiply > Morph**) and set the number of segments and click **OK**.



**NOTE:** **Morph Polygons** works best when you use it between single-sided polygons.



**HINT:** To get a twist effect, rotate the source or target polygon a little.

If you morph using curves, the curves are treated as straight-edged polygons, and will morph in a linear fashion. The curves also remain in the Modeler workspace following the operation, although they will not be part of the morphed object.

The **Number of Segments** field defines the number of morphed steps you desire between the two selected polygons.



**NOTE:** As with the **Morphing** function in Layout, the **Morph Polygons** command is dependent on point order. As such, you should create the source and target from the same polygon. You may need to flip some polygons after using this command.

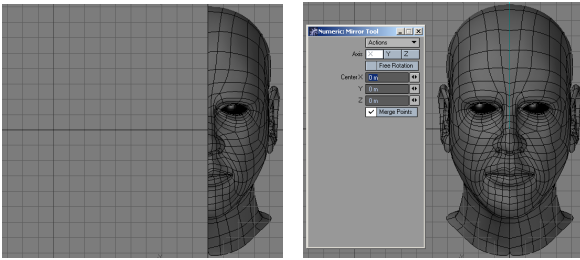


## Duplicate

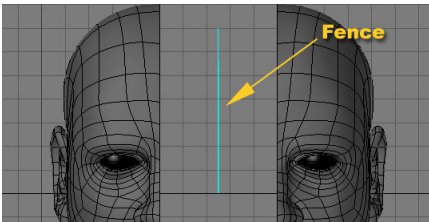
### Mirror Tool

(default keyboard shortcut **Shift V**)

Use the **Mirror** tool (**Multiply > Duplicate: Mirror**) to create an exact mirror image of an item. You can mirror the whole object, selected points, or selected polygons. Many objects you create will be symmetrical in nature. So it often makes sense to build only half of an object, create a mirror image, and join them together. At other times, your object may include certain parts that are identical, and an automated way to duplicate them is very helpful.

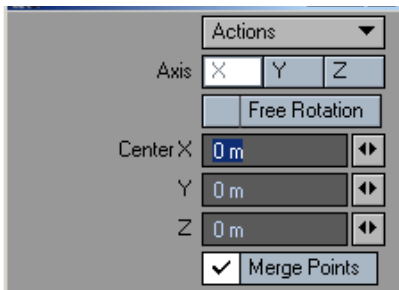


This interactive tool works using a mirroring *fence*. You drag out the fence with your mouse. This forms the main axis for the fence's length. Once you release the mouse button you can reposition by simply clicking somewhere off the fence — you are essentially repositioning the initial drag point.



You can rotate the fence using the rotation handles. The swing rotation handle *swings* the fence like a gate around one end of the fence. The axis rotation handle rotates the fence around the axis along the length of the fence.

The numeric panel defines the XYZ coordinates of the **Center** point (where the position handle is). The **Axis**, if any, is the axis perpendicular to the fence.



If **Free Rotation** is inactive on the numeric panel, the default setting, you will only be able to draw and rotate the fence across planes aligned with an axis. If you hold the **Ctrl** key down, you will be able to use 15-degree increments. If **Free Rotation** is active, you are free to draw and rotate the fence at any angle.

If the **Merge Points** option is active, any overlapping points are merged automatically. Use this if you are mirroring two halves of an object together. When you are satisfied, simply click **Multiply > Mirror** again (or select another tool). To reset, click a non-active part of the interface or click **Undo**.



NOTE: If adjusting the axis rotation handle only resizes the fence, try adjusting the handle in the perspective view.



## Mirror XYZ Tools

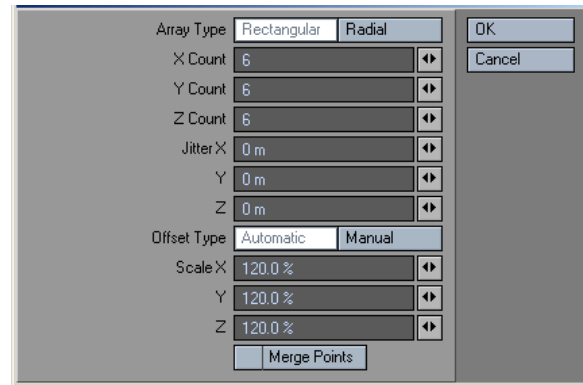
Mirrors the selected object along the zero value of the chosen axis.

## Array Tool

(default keyboard shortcut **Ctrl** **Y**)

The **Array** command (**Multiply > Duplicate: Array**) duplicates the selected item any number of times using specific increments of distance. You can create a grid out of this object, like a matrix. The shape of the array can either be **Rectangular** (box-shaped) or **Radial** (circular).

### Array Type: Rectangular



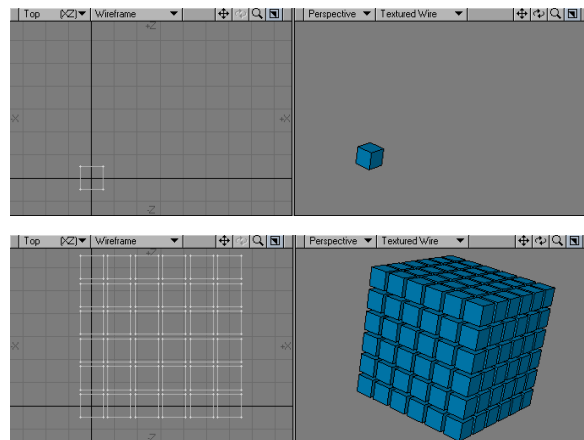
When **Rectangular** is selected, you will have the following settings available:

**Count** — Count is the number of repetitions of the original object along each of the axes.

**Jitter** — Jitter adds randomisation. A distance up to the value entered is added or subtracted to what the position would have otherwise been.

**Offset** — **Automatic** uses the object's own dimensions as the spacing value. The **Scale** fields define a percentage of the dimensions to use. A value of 100 percent in the input fields will place the objects side by side. **Manual** lets you enter specific spacing values for each axis.

Use the **Merge Points** option to automatically merge points at the end of the operation.

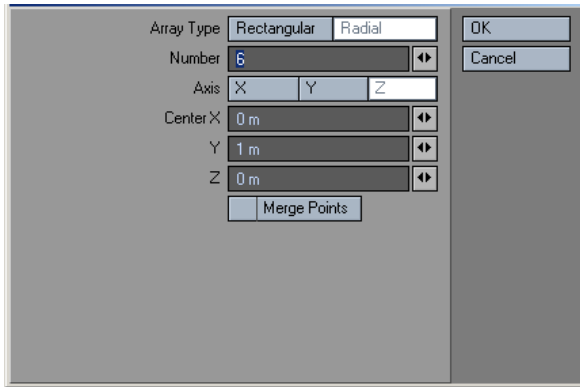


**NOTE:** Using **Automatic** on a 2D object will cause the copies to be on top of each other along the missing dimension axis.





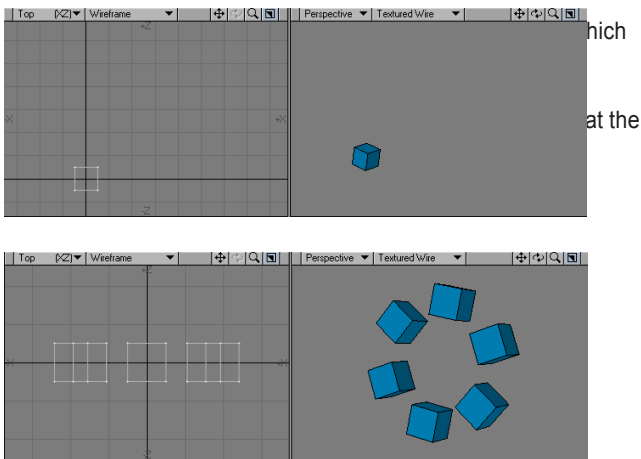
## Array Type: Radial



When **Radial** is selected, you have the following settings available:

**Number** — The number of repetitions of the original object.

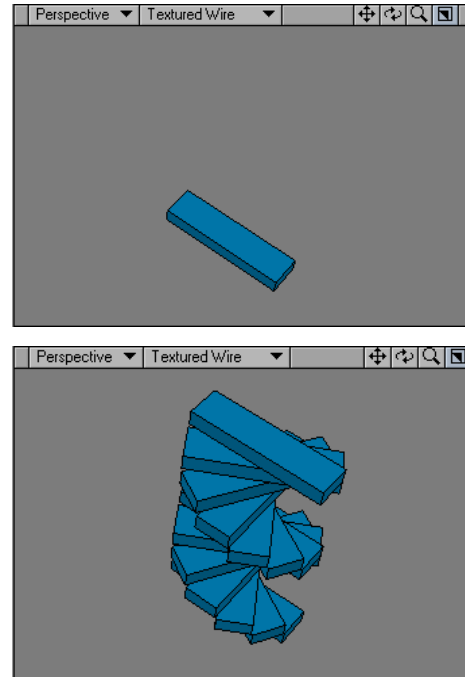
**Axis** — The axis perpendicular to the radial shape.



## Clone Tool

(default keyboard shortcut **C**)

The **Clone** command (**Multiply > Duplicate: Clone**) creates copies of an item that you can spread out in an even layout or array. You can create a series of copies along an axis at specified increments. **Clone** duplicates a selected item one or more times using specific increments of distance, rotation, and scaling. You can create a spiral staircase out of a single step, or a wood screw out of a triangular polygon outline, for example.



Enter the number of copies of the selection to create in the **Number of Clones** fields.



The **Offset** fields determine the incremental distance between one clone and the next along the **X**, **Y**, and **Z** axes.

The **Scale** fields set the amount of incremental scaling for each copy.

The **Rotation** fields set the amount of incremental rotating.



**NOTE:** The effect is progressive, so the change is applied to each clone based on the previous clone. So, for example, with scaling, the copies

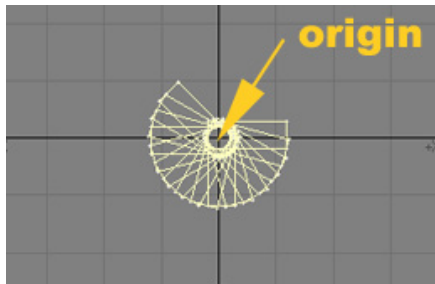


will get progressively larger or smaller.

The **Center** fields define the coordinates to use as the center point, around which the operation takes place.



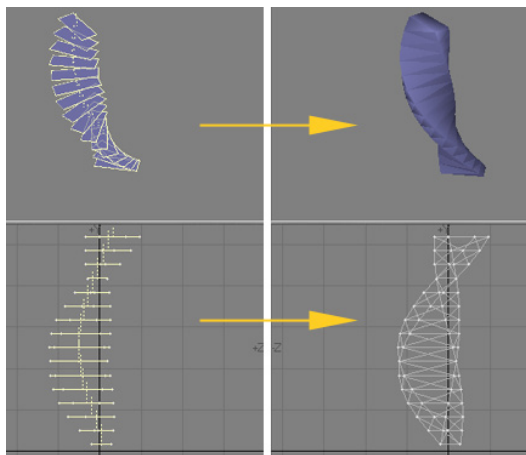
HINT: If you plan to clone with rotation, try to build your base object so that the rotation can happen around the Origin whenever possible.



HINT: To evenly space objects around an axis, take the number of total objects you want to end up with and divide it into 360 (you can even enter the sum into the input field.) This gives you the incremental rotation. Then set the **Number of Clones** to the total objects minus one.



HINT: Immediately after a **Clone** operation involving a template (i.e., 2D) polygon, you can use the lasso to encircle and select the entire group of newly cloned polygons. They are automatically selected in the order they were created. This means you can use the **Create Skin** command (**Multiply > Create Skin**) to place an outer surface around the new framework. After using **Create Skin**, the framework polygons are no longer needed (since they are inside the skin and will not render anyway). They are still selected, so you can delete them (except maybe the two ends), and leave only the skinned object remaining. This technique is useful when you create objects like a wood screw, where you no longer need the internal (unseen) polygons after you complete the outer surface.

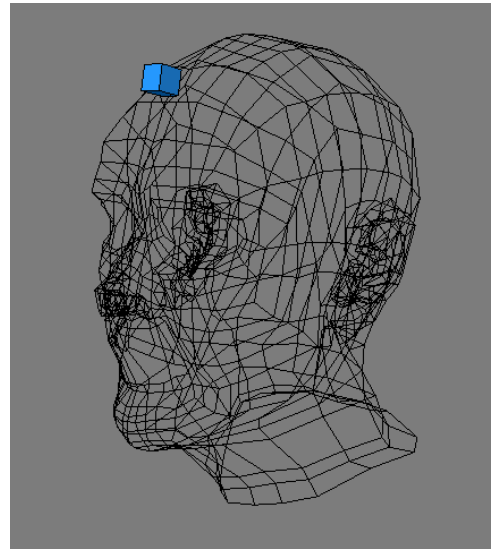


## Particle Clone

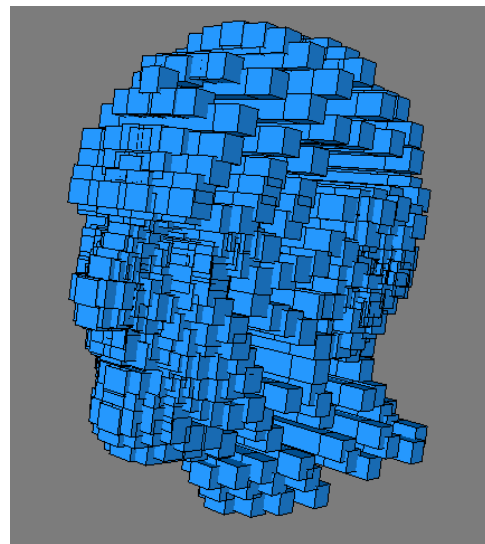
**Particle Clone (Multiply > Duplicate: More > Particle Clone)** will make a copy of the foreground object on each point that makes up the background object. It's great for asteroids, bubbles, spikes, grass, etc.

### Using Particle Clone

**Step 1:** Place the object that you would like to clone in the Foreground layer and the object that has the point information in the Background layer.



**Step 2:** Choose **Multiply > Duplicate: More > Particle Clone**, and sit back and enjoy the magic!



NOTE: If no object is in the Background layer, **Particle Clone** will clone the object in the Foreground layer using the points that make up that object.

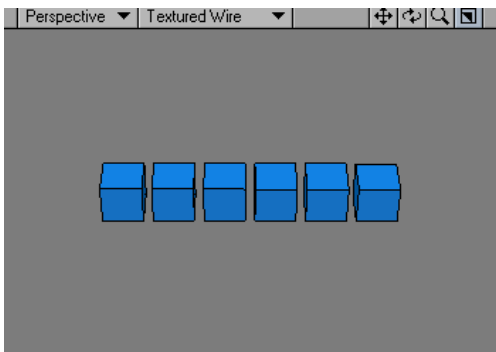
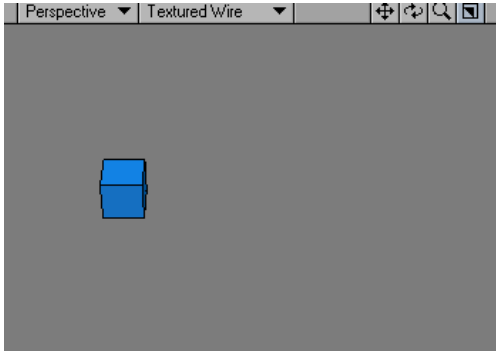


NOTE: **Point Clone Plus (Multiply > Duplicate: More > Point Clone Plus)** performs the same task but has options.



## Clone to Layer

The **Clone to Layer** command (**Multiply > Duplicate: More > Clone to Layer**) creates copies of an item that you can spread out in an even layout or array. You can create a series of copies along an axis at specified increments. **Clone to Layer** duplicates a selected item one or more times using specific increments of distance, rotation, and scaling. You can create a spiral staircase out of a single step, or a wood screw out of a triangular polygon outline, for example.



This command is almost identical to the standard **Clone** (**Multiply > Duplicate: Clone**) command, but offers the ability to place your cloned geometry in multiple layers.

### Clone to Layer Numeric Options:



**Number of Clones** — Enter the number of copies of the selection to create in the **Number of Clones** fields.

**Clone Output** — Choose between Current Layer and Multi- Layer.

**Current Layer** — This option will place the clones in the same layer as the original.

**Multi-Layer** — This option will place the clones in individual layers.

Select the **Use only empty layers** option to skip over layers that contain data when clones are being created.

The **Offset** fields determine the incremental distance between one clone and the next along the **X**, **Y**, and **Z** axes.

The **Scale** fields set the amount of incremental scaling for each copy along the **X**, **Y**, and **Z** axes.

The **Rotation** fields set the amount of incremental rotating around the **Heading**, **Pitch**, and **Bank**.



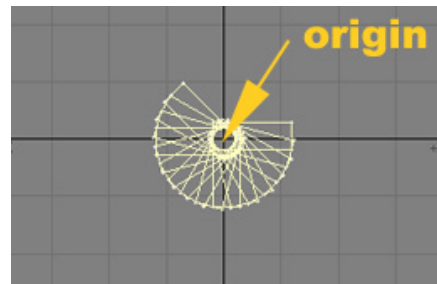
**NOTE:** The effect is progressive, so the change is applied to each clone based on the previous clone. So, for example, with scaling, the copies will get progressively larger or smaller.

The **Center Around** option consists of Origin (0,0,0), the object's center, and User Definable.

The **Center** fields define the coordinates to use as the center point, around which the operation takes place.



**HINT:** If you plan to clone with rotation, try to build your base object so that the rotation can happen around the Origin whenever possible.

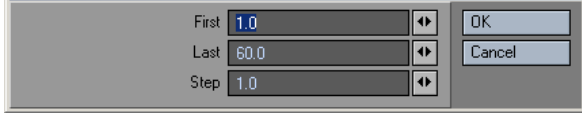


**HINT:** To evenly space objects around an axis, take the number of total objects you want to end up with and divide it into 360 (you can even enter the sum into the input field.) This gives you the incremental rotation. Then set the **Number of Clones** to the total objects minus one.



## Motion Path Clone Tool

The **Path Clone** command (**Multiply > Duplicate: More > Motion Path Clone**) works just like the **Path Extrude** command, however, instead of resulting in an object with a smooth skin, it creates *skinless* segment clones along the motion path. **Motion Path Clone** will create clones of the selected geometry along the chosen motion path.



**First:** This option defines the first frame in the motion file that will be used.

**Last:** This option defines the last frame in the motion file that will be used.

**Step:** This option determines how many clones will be used in the extrusion per frame. The lower the number the more segments.

Step Examples:

First Frame 1, Last Frame 60

Step 0.5 - 120 clones

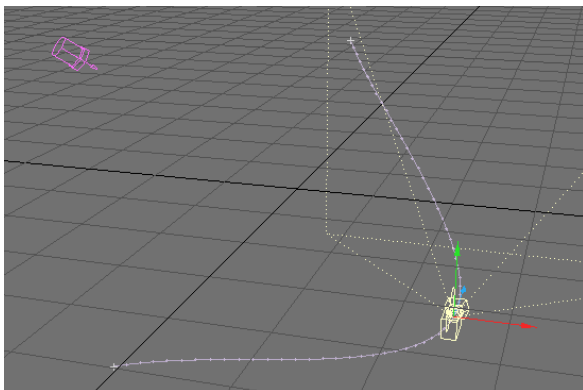
Step 1 - 60 clones

Step 2 - 30 clones

Step 3 - 20 clones

### Steps to extrude along a motion path

**Step 1:** Create a motion path for an item in Layout.

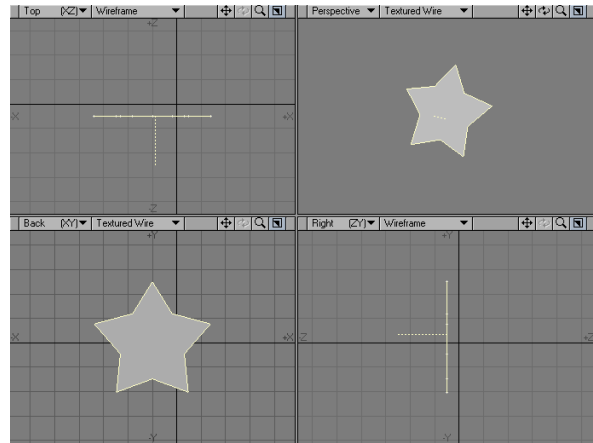


**Step 2:** With the item still selected go to **File/Save/Save motion file**.

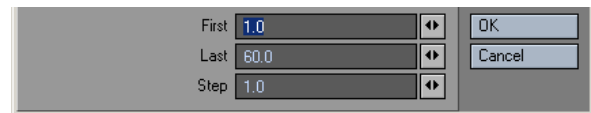


NOTE: Be sure the file has the .mot extension.

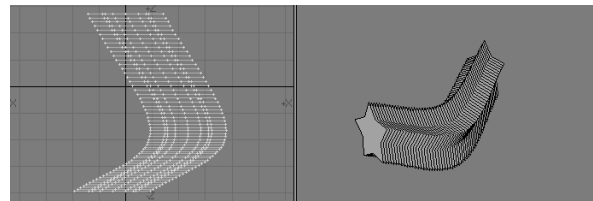
**Step 3:** Open the Modeler and create an object.



**Step 4:** Select **Motion Path Extrude (Multiply> Duplicate: More > Motion Path Clone)**. Select the path saved above and change settings in the numeric window as needed.



**Step 5:** Click OK.



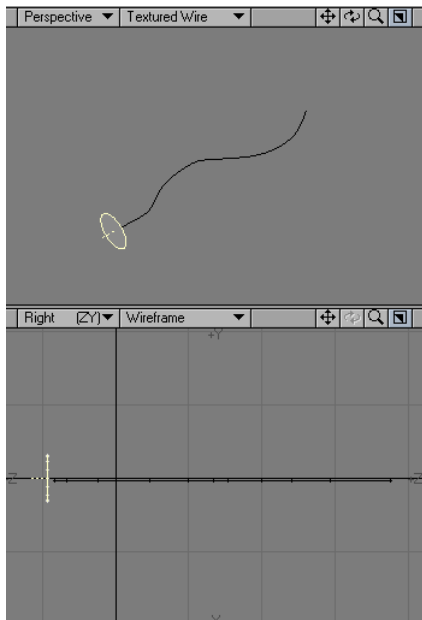


## Rail Clone Tool

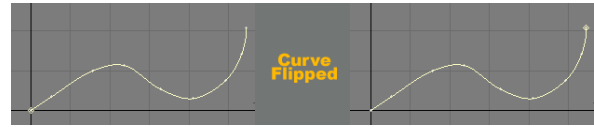
The **Rail Clone** command (**Multiply > Duplicate: More > Rail Clone**) works just like the **Rail Extrude** command, however, instead of resulting in an object with a smooth skin, it creates *skinless* segment clones along the background curve(s).

**Rail Clone** will create clone or clones of a point, polygon, or object along a curved path (single rail clone) or multiple paths (multiple rail clone). It is nearly identical to the **Rail Extrude** command. The difference is that **Rail Clone** creates individual copies of the object instead of forming a continuous skin for the object.

The curve(s) must be in the background layer and the polygons must be cloned in the foreground layer. The orientation and position of the foreground polygons will affect the result. Generally, they should be at the very beginning of the curve and oriented perpendicular to it — as if the curve was a thread you were threading through the polygons.

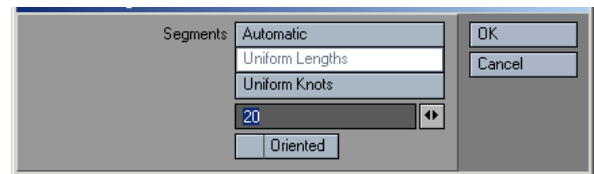


The direction that the curve faces (which end is considered the beginning) will have bearing on both the shape and direction of the cloning. If you don't get the results you expect, try flipping the curve's starting point with **Detail > Flip**. If you are using multiple curves, they should generally run in the same direction.



### Single Curve

The **Rail Clone: Single** dialog will appear when a single curve is in the background.



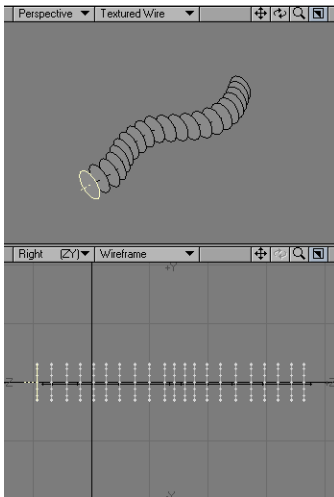
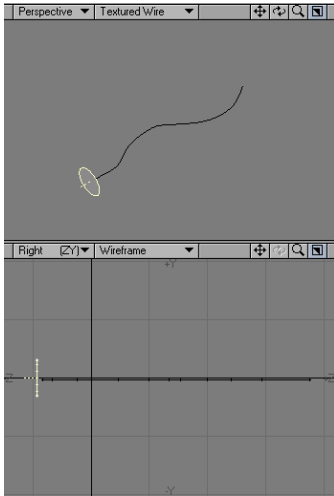
**Segments** is the number of clones in the resulting object. Select **Automatic** to distribute a number of clones along the rail curve (based on the **Curve Divisions** setting on the **General Options Panel**) according to the curve's knot (point) spacing.

Select **Uniform Lengths** to distribute some number of clones (which you specify) evenly along the length of the entire curve. No matter what the knot spacing is, the clones will remain evenly spaced.

Select **Uniform Knots** to distribute some number of clones (which you specify) evenly between the knots that make up the curve. Whether the knots are close together or far apart, there will be the same number of segments between each.

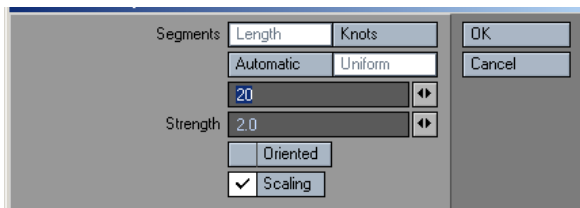


**Oriented** determines the orientation of the polygon template as it follows the curved path. When active, it causes all clones to angle themselves automatically so that they are aligned with the rail curve as it turns. When inactive, it causes all clones to remain in the same orientation as the template so that they face exactly the same way.



## Multiple Curves

The **Rail Clone: Multiple** dialog will appear when multiple curves are in the background.



In practice, one curve will act as the main cloning curve and the other(s) act as shaping curves. The effect of the shaping curves is determined by the distance from the main curve. However, you may change the effect by moving the polygons closer or farther away from any of the curves.

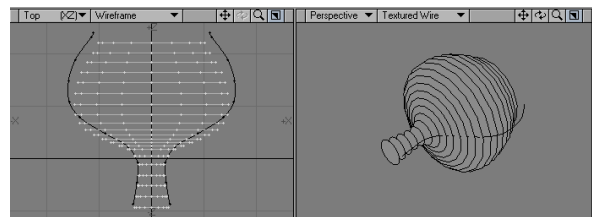
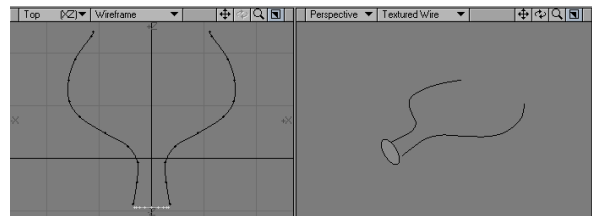
**Segments** is the number of clones in the resulting object. Select **Automatic** to distribute a number of clones along the rail curve, based on the **Curve Divisions** setting on the **General Options Panel**. If you also select **Knot**, clones are distributed with regard to the curve's knot (point) spacing. If you use **Length**, knot spacing is disregarded.

To specify the number of clones, choose **Uniform** and enter the number in the input field. If **Length** is also active, the clones will remain evenly spaced no matter what the knot spacing is. If you use **Knot** instead, there will be the same number of clones between each knot, no matter how close together or far apart they are.

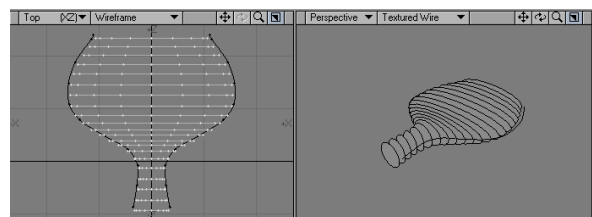
**Strength** determines how strongly the rails vie for control of the point locations. You will not notice much of a difference unless you have **Scaling** (below) deselected. The higher the **Strength** value, the closer the object will *hug* the rails.

**Oriented** determines the orientation of the polygon template as it follows the curved path. When active, it causes all segments to angle themselves automatically so that they are aligned with the rail curve as it turns. When inactive, it causes all clones to remain in the same orientation as the template so that they face exactly the same way.

**Scaling** determines how clones along an axis are sized. If the rails in the background layer spread apart along any axis or axes, then the clones will automatically stretch along those axes as well. Turn **Scaling** on to scale clones equally, rather than along those specific axes only.



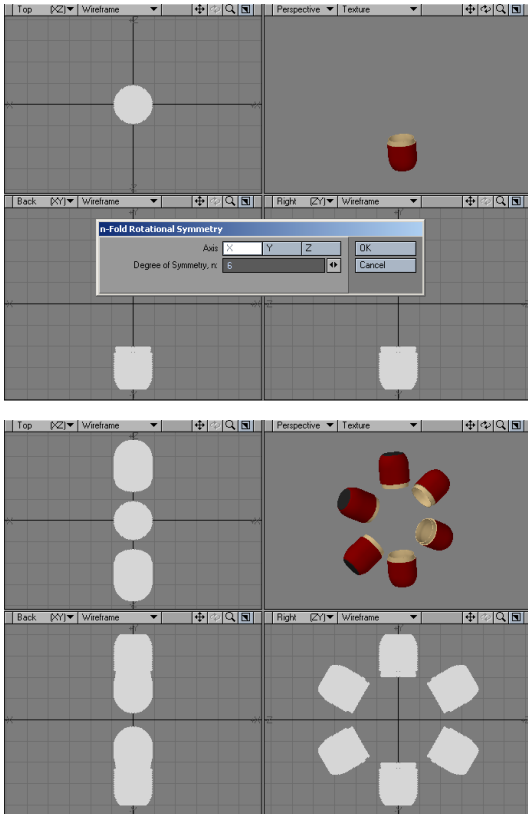
For example, with **Scaling** off, if you **Rail Clone** a polygon along the Z axis using two rails that spread further apart in the X axis, the clone's copies will be stretched on the X axis to maintain their relationship to the guide rails.





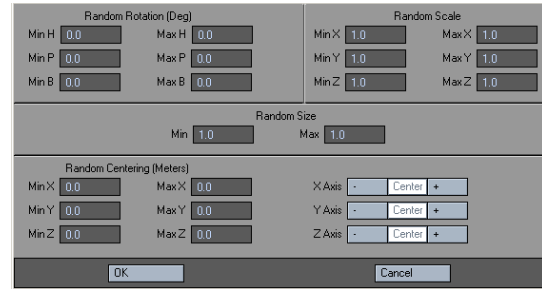
## Symmetrize Tool

The **Symmetrize** command (**Multiply > Duplicate: More > Symmetrize**) clones an object  $n$  times with a rotation each time so that the result displays  $n$ -fold symmetry about the selected **Axis**. An object is said to have  $n$ -fold symmetry if it looks the same after being rotated  $360/n$  degrees.

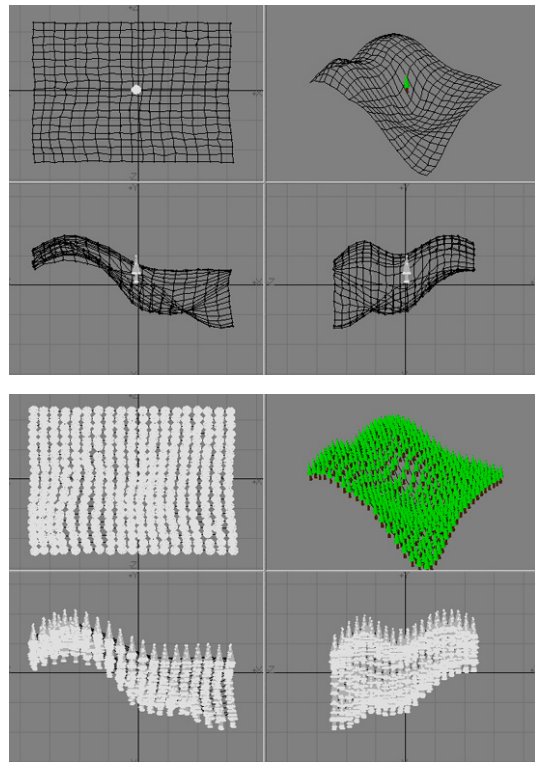


## Point Clone Plus Tool

The **Point Clone Plus** command (**Multiply > Duplicate: More > Point Clone Plus**) will clone objects in the foreground to the locations of points in the background. You can randomise the rotation, scale, and centring independently for every axis between two values. **Random Size** will randomly change the overall scaling of objects. The **XYZ Axis** options specify whether the objects should be centered on the points or flush on one side or another. To specify static values, enter the same number in the **Min** and **Max** fields.



This command is great for cloning and placing, say, tree objects on a landscape or putting hair on a head.

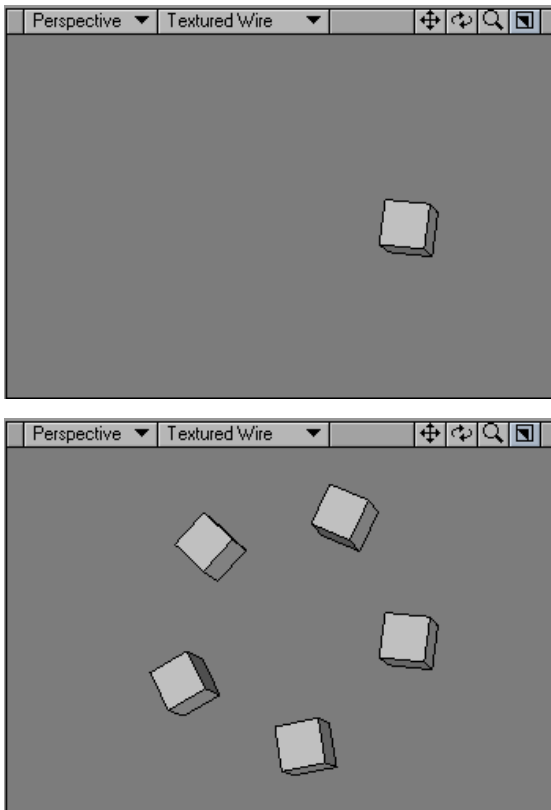




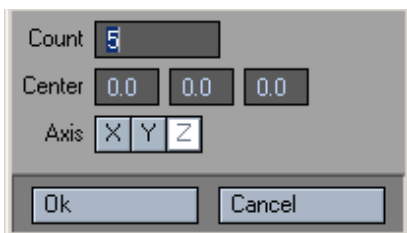


## Radial Array Tool

Radial Array (**Multiply > Duplicate: More > Radial Array**) creates an array of geometry around the **Center** position using the specified **Axis**. The **Count** setting sets the number in the final array.

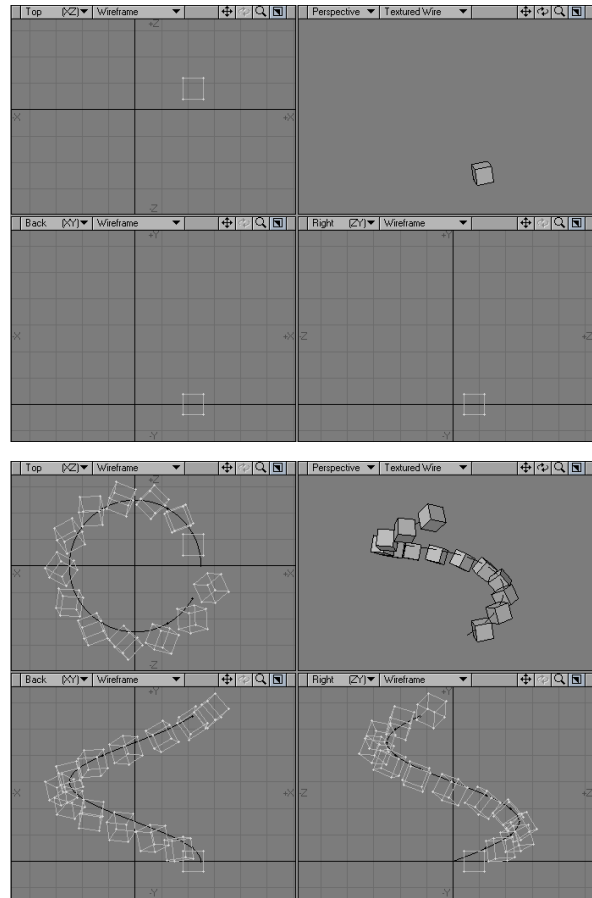


NOTE: This is a non-interactive command so you will need to numerically input the settings.



## Helix Tool

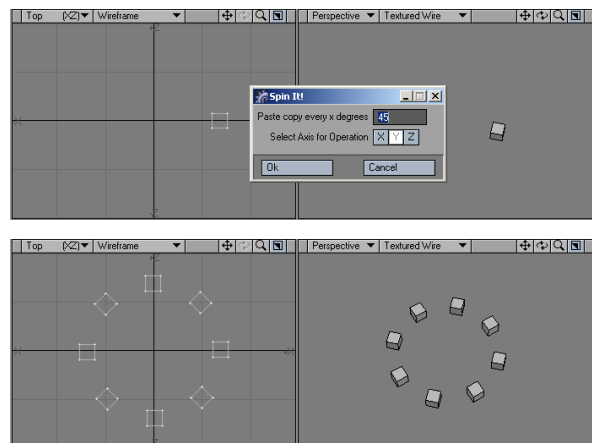
The **Helix Object** function (**Multiply > Duplicate: More > Helix**) will create a one meter spiral curve in a background layer and clone the geometry in the foreground layer along it.



NOTE: There are no settings for this command so when using **Helix** keep in mind that the **Spiral Curve** will be 1 meter in size.

## Spin it Tool

The **Spin it** command (**Multiply > Duplicate: More > Spin it**) will create an array of geometry around the Origin using the specified **Axis**.





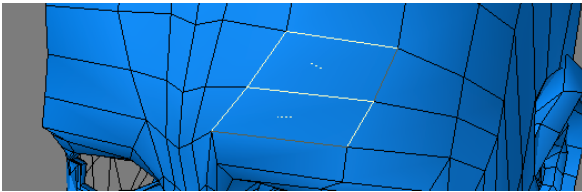
## Subdivide

### Add Points

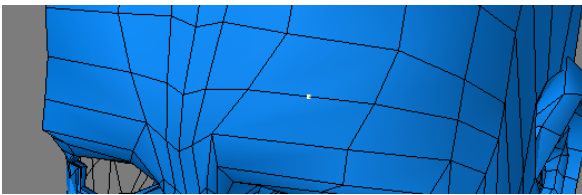
The **Add Points** operation (**Multiply > Subdivide: Add Points**) will give you the ability to manually increase the point count on a selected polygon. Sometimes you will need to add a point to a polygon and then move the point to change the shape of the polygon. You might also want to add multiple points so you can manually subdivide a polygon into smaller polygons.

#### To add a point to a polygon:

**Step 1:** First, select the target polygon. If your selected edge is shared with another polygon, select that one as well. If you do not, the edge will split and the polygons will no longer be joined.



**Step 2:** Choose **Multiply > Add Points**. Click the black plus sign at your desired insertion point along the edge.



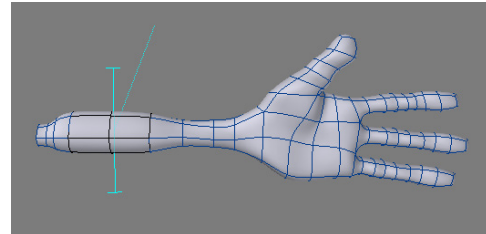
NOTE: Switch to the Point Selection mode and deselect any points *before* using **Add Points**.

Adding points while there are already points selected will cause the added points to snap and weld to the selected points.

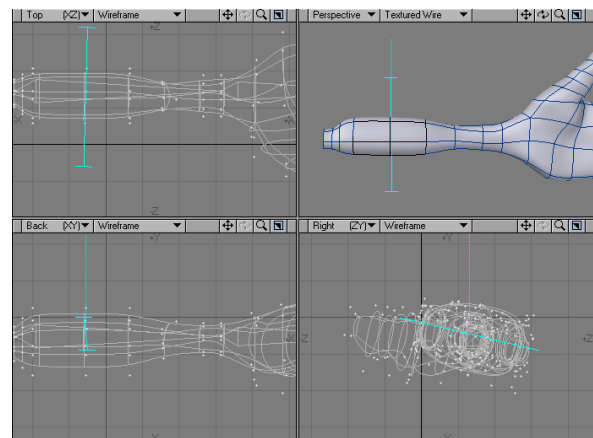
## Knife Tool

(default keyboard shortcut **Shift K**)

The **Knife** tool (**Multiply > Subdivide: Knife**) lets you interactively get results similar to the **Drill** function's **Slice** option; however, it requires no background object. You simply drag out a *slicing line* in any viewport — nothing needs to be in a background layer. The line is infinite along the axis perpendicular to the view you initially drag in. This tool is perfect for adding segments to your geometry.

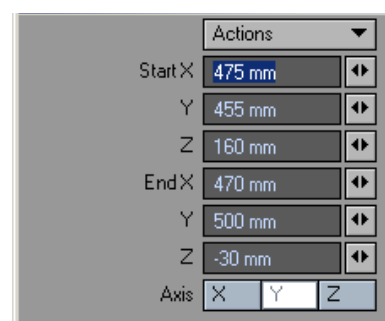


The other viewports will let you see the line from a different angle and a small dotted line will be visible that shows you the direction of the slice. After you have released the mouse button, you can drag the *T-handles* at the ends of the line (in any viewport) to reposition them or drag the center of the line to move the entire line. The tip of the dotted line can also be dragged to change the direction in increments of 90 degrees. This is an interactive tool so you will see the resulting new geometry as you position the line.



Like most tools, drop it or select another to make the geometry, or click your **RMB** to create the geometry without dropping the tool. Click on an open area of the interface to cancel the operation before it is made or use **Undo**.

The **Knife** tool's numeric requester indicates the **Start** and **End** XYZ coordinates, defining the end points of the line. The **Axis** settings define a target point, relative to the base of the dotted direction line, that the line goes through.

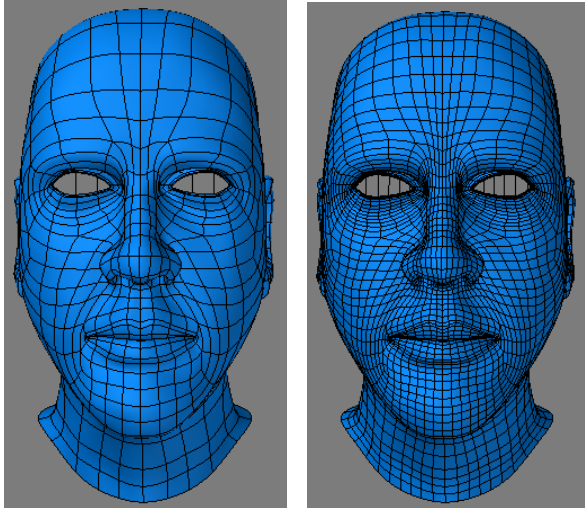




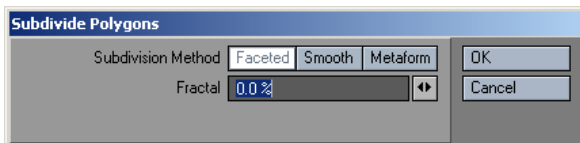
## Subdivide

(default keyboard shortcut **Shift D**)

The **Subdivide Polygons** operation (**Multiply > Subdivide: Subdivide**) divides any selected polygons with three or four sides into smaller polygons. This makes an object appear smoother, but also makes the model more complex.



The **Subdivide Polygons** panel appears when you choose **Multiply > Subdivide: Subdivide**. This panel provides several commands that let you automatically increase the detail of existing polygons. To use this panel, you must select the target polygons first.

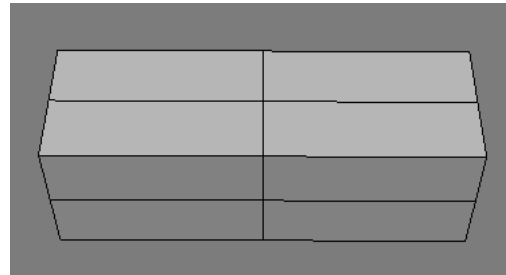


**NOTE:** You cannot use **Subdivide Polygons** on polygons with more than four sides.

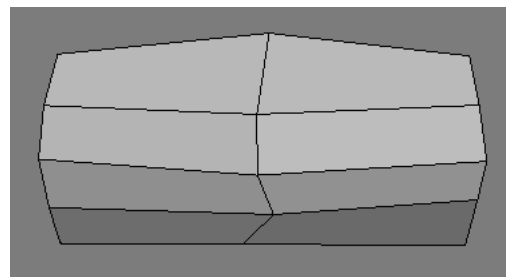
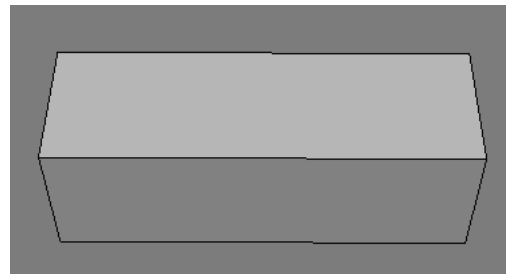
## Randomising Options

All **Subdivide** options let you enter a **Fractal** factor that will randomly jitter newly created points. The result is a subdivided object with its points randomly jittered. A factor of 0 will not jitter points and higher values will jitter points more.

The jitter radius is a fraction of the edge length, times the **Fractal** factor. A factor of 1 will jitter the new points by at most 50 percent of the length of the edge that they will subdivide. A factor of 0.1 is five percent jitter and a factor of 2 will move points 100 percent of the edge length. Points added to the center of four-point polygons are jittered a comparable amount, although the formula for them is more complicated.



0 Fractal

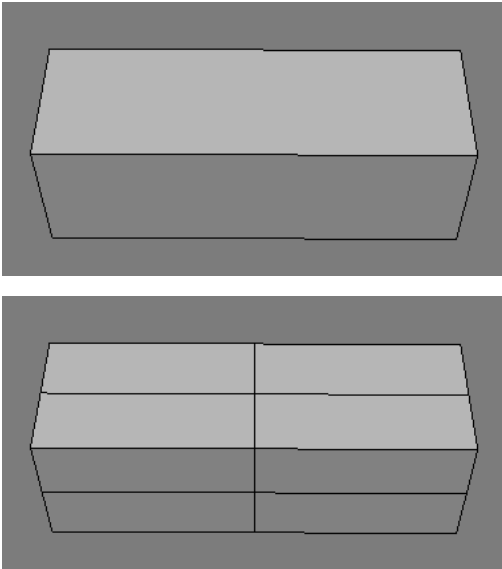


50% Fractal



## Faceted Subdivide

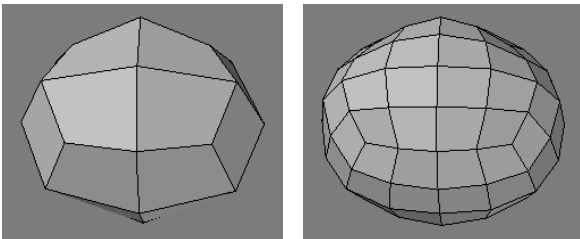
The **Faceted** option adds more segments to existing polygons, but doesn't change the overall shape. Let's say you made a box, but just wanted one side to have multiple segments. In this case you could use **Faceted**.



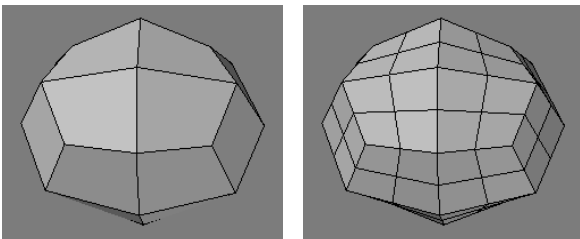
NOTE: **Faceted Subdivide** works on triangles, when the **Triple** command is used.

## Smooth Subdivide

**Smooth** will subdivide objects that have curved, smoothed, or rounded areas. Modeler assumes any two polygons are part of a smooth surface and will extrapolate where subdivided detail should be placed to maintain (even enhance) the curvature of the original form.



Smooth Subdivide

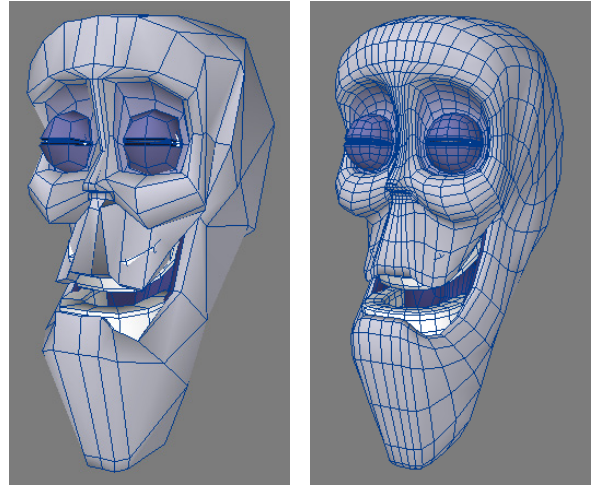


Faceted Subdivide

The value entered in the **Max Smoothing Angle** field is used to determine whether adjoining polygons should be smoothed or not. Adjacent polygons whose normals form an angle above this value will not be smoothly subdivided. The default is 89.5 degrees, so polygons at right angles or sharper will not be affected.

## Metaform Subdivide

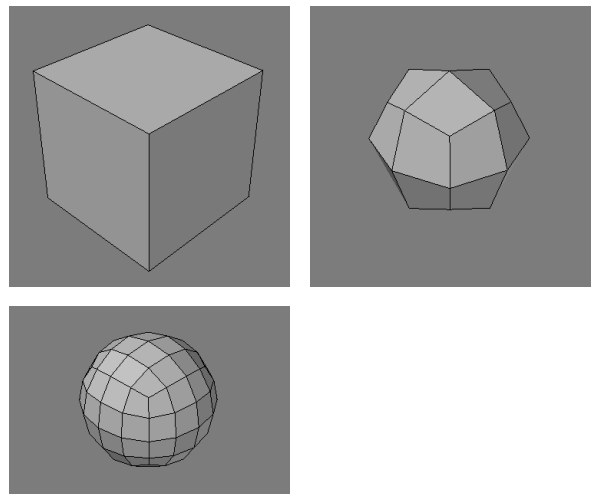
**Metaform** is another method of subdividing. The object will be smoothed dramatically, with the original object acting as a kind of bounding box template for a slightly smaller, more rounded form. **Metaform** is very useful for creating smooth organic-looking objects out of simple geometric structures.



**Metaform** is an adaptive process, meaning that locations that contain greater detail (i.e., more points/polygons) will have more detail in the smoothing process. It is easy to create rounded forms with **Metaform**. Moreover, if you use **Metaform** on part of an object by selecting polygons, you can create specific areas of greater detail on an object without subdividing the entire object again and again.

Adjoining polygons with surface normal angles greater than the **Max Smoothing Angle** will not be subdivided in a smooth manner. The default angle is 179 degrees, so only polygons that are nearly parallel with each other are not affected.

Metaform is often used one or more times on a box to create a rounded starting point for SubPatch cages. It is superior to using, say, a ball primitive, since all polygons are quads.



NOTE: **Smooth** and **Metaform** can achieve similar results using the same maximum smoothing angle; however, **Smooth** will tend to smooth outward, and make a slightly larger object. **Metaform**, on the other hand, smooths inward, and makes a more compact object.



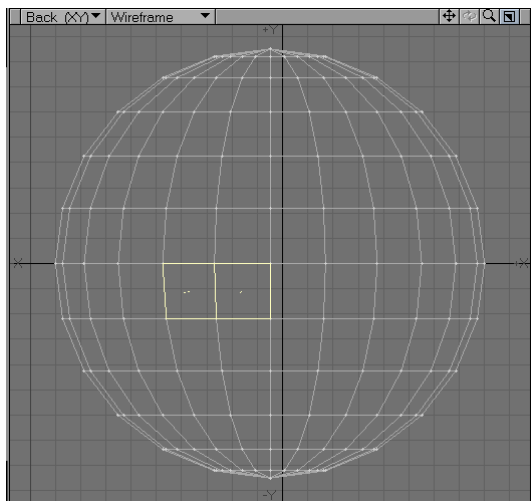
## Band Saw Pro Tool

**Band Saw Pro (Multiply > Subdivide: Band Saw Pro)** will slice through a contiguous band of four-point polygons (or patches). While the images below use a simple sphere, **Band Saw Pro** is capable of navigating a more complex object – note the illustrations used in the later section on **Multiply > Subdivide: More > Band Saw**.

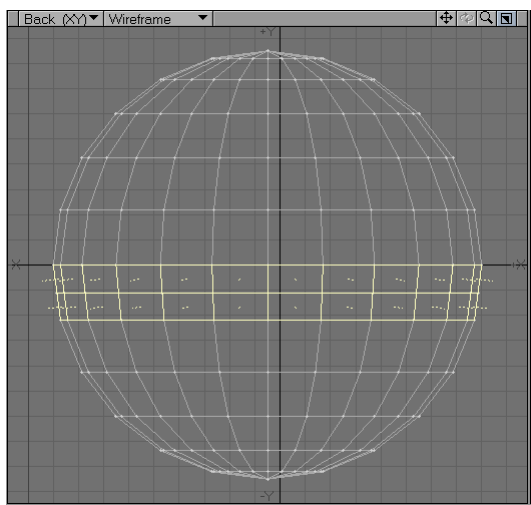
To use the **Band Saw Pro** tool, be sure you are in Polygon selection mode, then just click on an edge to select two polygons in the “band” that you wish to divide and activate the **Band Saw Pro** tool (**Multiply > Subdivide: Band Saw Pro**). The tool will select the band of polygons going either direction from the first two selected, and will indicate the slice(s). The selection band and the slice(s) will continue all the way around until the operation returns to the point of origin or encounters a non-four-point polygon. The tool's Default setting is for a single slice right in the center. The **Numeric** panel options can be used to create multiple slices, with a variety of controls, and these can be saved as custom presets.



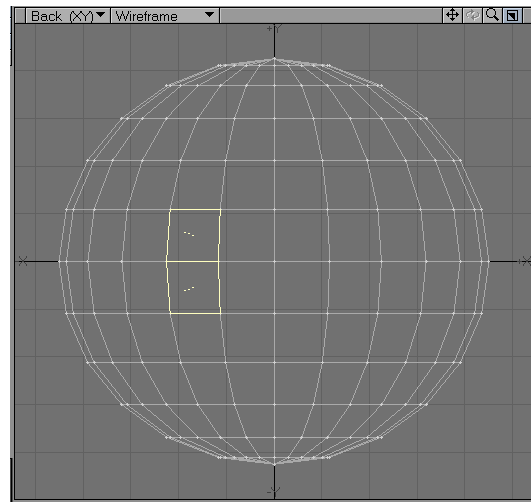
**NOTE:** **Band Saw Pro** defaults to the last preset used, even across sessions.



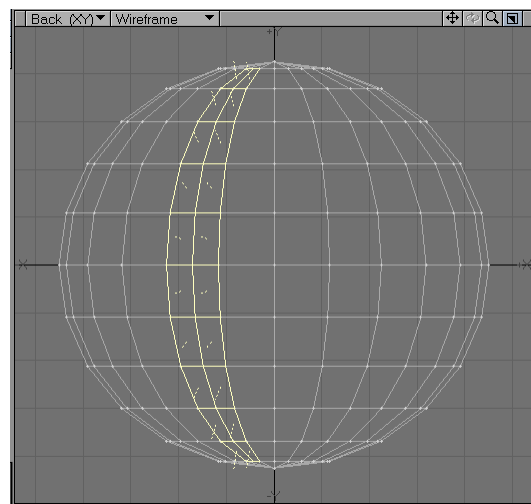
Selecting polygons for a horizontal band.



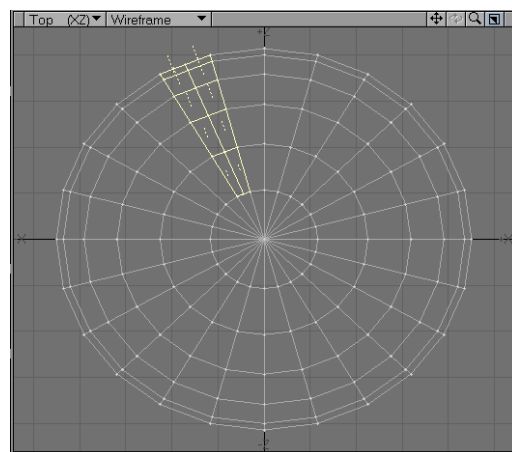
When activated, Band Saw Pro selects rest of band and indicates the slice.



Select polygons for a vertical slice.



When activated, Band Saw Pro selects the rest of the band and indicates the slice.



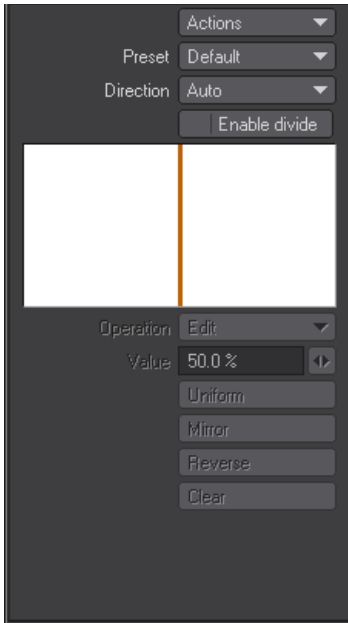
In the Top View, note that Band Saw Pro's operation stops when it encounters a non-four-point polygon.



**NOTE:** You can select just one polygon, and use the Numeric panel if you need to adjust the selected band. When you select more than one polygon, the first selected polygon controls the operation.



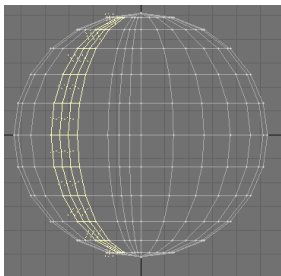
## Band Saw Pro Numeric Options



Band Saw Pro Numeric Panel and Pop-ups

The **Actions** pop-up lets you **Reset** the tool or **Activate** it to make the slice.

The **Preset** pop-up lets you apply the default settings for the tool, or one of up to 10 custom settings. A custom setting can be created by selecting one of the **Custom** options on the **Preset** button, entering the settings for **Enable Divide**, **Value** to place the relative position of one or more slices, the **Operation** pop-up to add or change the location of slices, and, where needed, by applying **Uniform**, **Mirror**, or **Reverse** to set or fine-tune the final version of the slice(s). Press the space bar to accept the entries and execute the slice, and the settings will also be saved as the options for the **Custom** setting that was selected.



The **Direction** option determines the direction of the band and the slice. With a quad, there are only two directions to proceed and these are described as **Even** or **Odd**. **Auto** is the default setting, wherein **Bandsaw** will pick the direction for you.

**Enable Divide** activates the slices that you have set up. However, you can also use **Bandsaw** to just select the band of polygons by not activating this option.

The **Divide Window** shows the slices that you have set up for the current operation. You can also make changes or additions in this window using the **Operation** pop-up, discussed next. When there are multiple slices shown in the window, the one shown in red is selected, and the ones shown in blue are not selected.

The **Operation** pop-up lets you make changes and additions to the slices.

**Add** allows you to create new slices by clicking with the **LMB** in the **Divide Window**.

**Edit** allows you to select a slice using the **LMB** and move it using the **Value** slider.

**Delete** allows you to select a slice using the **LMB** and remove it.

**Value** lets you set the location of the selected slice by entering a value in the numeric field or by using the slider.

**Uniform** equalises the spacing between slices.

**Mirror** will mirror the selected slice across the center.

**Reverse** inverts the order — handy if you placed the slice(s) on the wrong side.

**Clear** will remove all slices and set it back to default.



NOTE: The **Band Saw** tool (**Multiply > Subdivide: More > Bandsaw**) is similar to **Band Saw Pro** in function, but is not interactive and includes a point selection option. Both tools are tailored for use on SubPatch objects in areas where you need more control points.



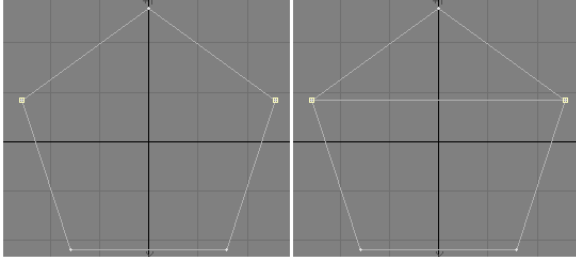


## Split Polygons Tool

(default keyboard shortcut **Shift** L)

Choosing **Multiply > Subdivide: Split** will divide a polygon into two smaller polygons using its existing points. If used on a curve, the curve will be split into two distinct curves.

Select a polygon and switch to the **Point** edit mode. Select two points that do not share any edges. Clicking **Split** will subdivide the polygon along a line between those two points.



NOTE: In **Point** mode, the **Bridge** tool (**Construct > Bridge**) works identically to the **Split Polygons** command, except with one convenient modification: you don't have to select the polygons you want to cut ahead of time.

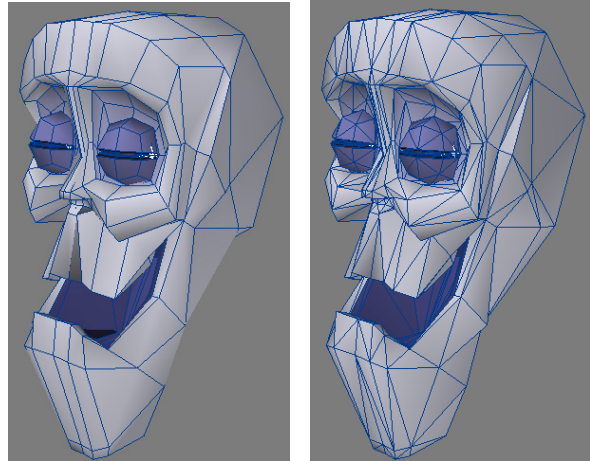


HINT: When working with Splines, repeating the **Split Polygons** tool when there are multiple points selected will split the spline into smaller splines.

## Triple Tool

(default keyboard shortcut **Shift** T)

The **Triple** command (**Multiply > Subdivide: Triple**) converts the selected polygon(s) into triangular polygons. For example, a square polygon becomes two triangular polygons. Triangles cannot by definition become non-planar; therefore, this tool is useful for objects you wish to animate using bones, or objects that will have **Displacement Maps**. Tripling non-planar polygons makes them planar.



NOTE: Tripling does not change the general shape of the object/polygons. Also, once a polygon is a triangle, tripling it again has no effect.





## Cut Tool

(default keyboard shortcut **Shift U**)

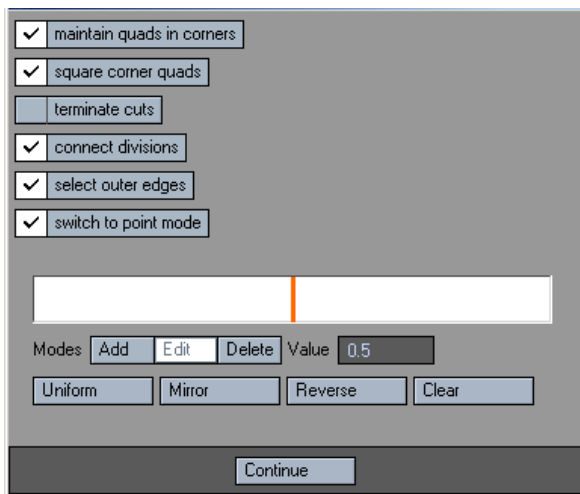
The **Cut** command (**Multiply> Subdivide: Cut**) is a general-purpose, interactive cutting tool. The tool cuts from edge to edge. You can select edges using the point selection or a quicker yet less flexible method of selecting edges is to select a sequential set of polygons. The edges in between the polygons and the opposing edge of each end poly will be selected.



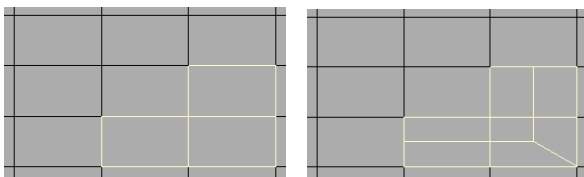
NOTE: The best way to see how this works is to simply experiment and practise.

**Cut** is capable of cutting through polygons with any number of points. The ability to terminate cuts and form corner-quads is still restricted, however, to 4-point polygons only. However, you can safely select n-point polygons even with those features turned on.

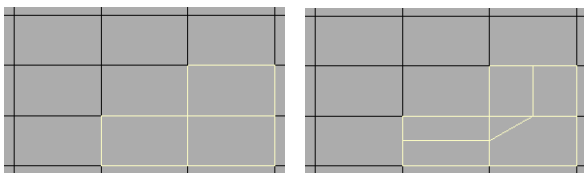
Choosing **Multiply> Cut** brings up the **Cut** panel.



The **Maintain quads in corners** option will create the necessary geometry to maintain 4-point polygons in corner geometry.

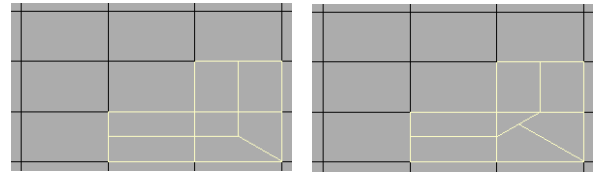


Maintain quads in corners: On



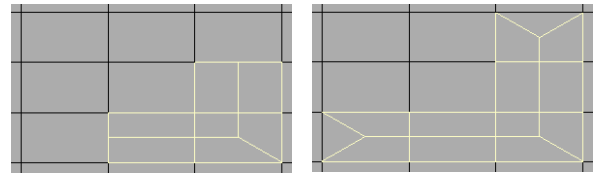
Maintain quads in corners: Off

The **Square Corner Quads** option will avoid creating 4-point triangle shapes in corners when the **Maintain quads in corners** is selected.



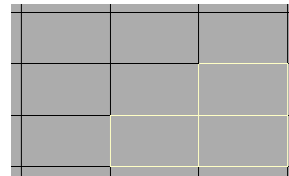
Left: Square Corner Quads: On, Right: Square Corner Quads: Off

The **Terminate Cuts** option will create extra geometry at the ends of the selection to avoid creating Ngons.

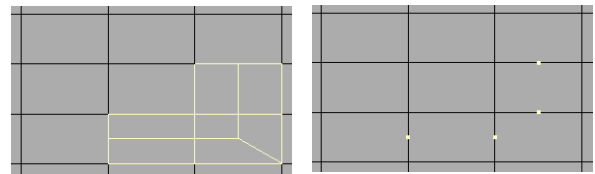


Left: Terminate Cuts: Off, Right: Terminate Cuts: On

When the **Connect Divisions** option is activated, **Cut** will split the polygons at the newly created points.

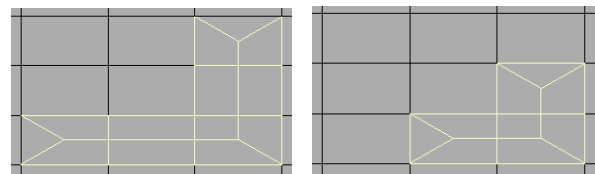


### Selection



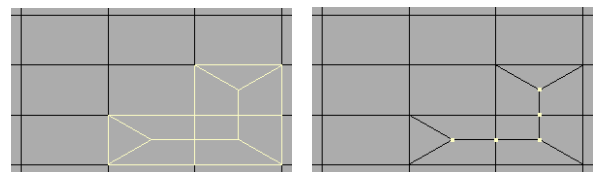
Left: Connect Divisions: On, Right: Connect Divisions: Off

The **Select Outer Edges** option will define whether polygon edges outside the selection will be affected.



Left: Select Outer Edges: On, Right: Select Outer Edges: Off

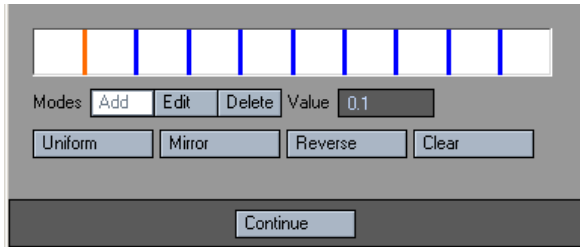
When the **Switch to Point Mode** option is active the newly created points will automatically be selected after the operation.





By default, there will be a single default slice right in the center. This is the positioning of the slice that will be made on the polygons.

The **Value** field will display the location of the selected slice. You can move it by selecting the **Edit** mode and dragging it with your mouse. You can also add more slices in the **Add** mode or remove them in the **Delete** mode.



**Uniform** equalizes the spacing between slices. **Mirror** will mirror the selected slice across the center **Reverse** inverts the order — handy if you placed the slice(s) on the wrong side. **Clear** will remove all slices and set it back to default.

Clicking the **Continue** button will apply the operation and close the **Cut** panel.



NOTE: For faster, non-interactive versions of **Cut** see **Quick Cut** and **Quick Saw** on page 238.

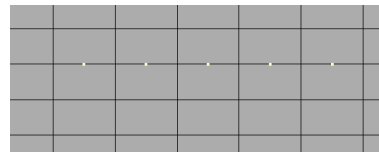
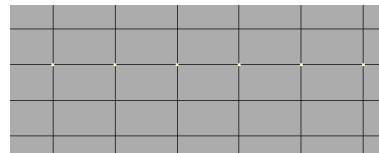
## Divide Tool

The **Divide** command (**Multiply> Subdivide: Divide**) is very similar to the **Add Points (Multiply> Subdivide: Add Points)** command in that it will give you the ability to increase the point count on a selected polygon. The main difference between the two is that the points are automatically added in the center of the polygon's edge when using **Divide**.

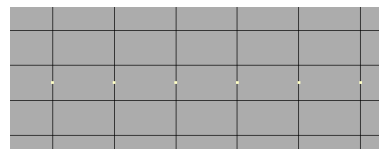
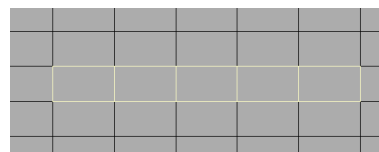
Sometimes you will need to add a point to a polygon and then move the point to change the shape of the polygon. You might also want to add multiple points so you can manually subdivide a polygon into smaller polygons.

The **Divide** command works in both **Point** and **Polygon** mode.

**Point Mode:** In **Point** mode, **Divide** will divide edges into two segments based on your point selection. Each pair of adjacent points you select will be treated as a selected edge to be divided. The tool will linearly search for adjacent point pairs - every possible linearly ordered pair it finds is used. This allows for the selection of adjacent edges without running the tool twice. However, it might take a little time to grow accustomed to the selection style, but it becomes natural over time.



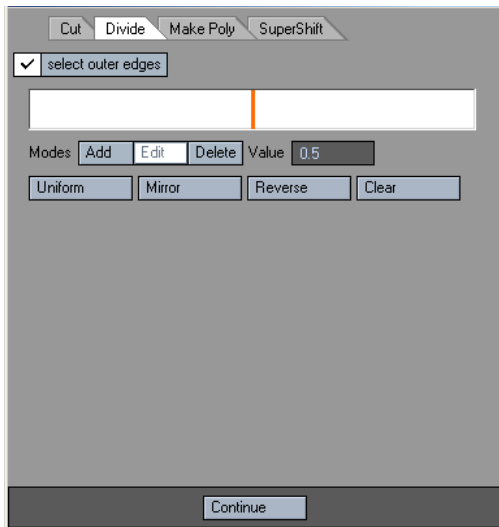
**Polygon Mode:** In **Polygon** mode, the tool will divide edges between polygons. It's a lot easier, quicker, and more straight forward this way than through point selections, but isn't as flexible for doing multiple edges within the same poly. Additionally, the tool can divide the edge at the starting and ending polygons, provided that they are not adjacent. If they are adjacent, then a loop is formed and the shared edge between the first and last polygon will also be divided. The best way to see this is to experiment with the tool.





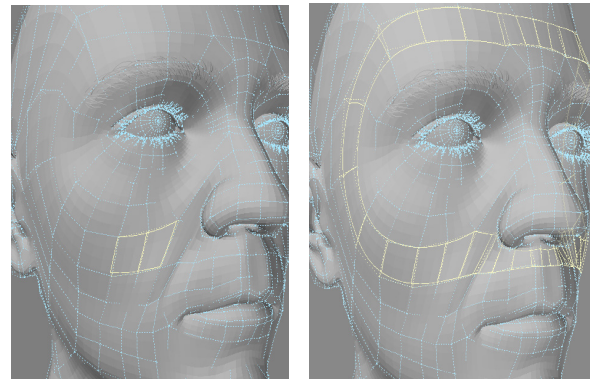
## Configuring Divide:

To configure the **Divide** settings, simply choose **Edit > Edit Modeler Tools** and select the **Divide Tab**.



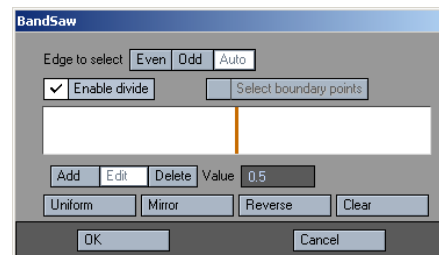
## Band Saw Tool

**BandSaw** will slice through a contiguous band of four-point polygons (or patches). To use the **BandSaw** tool, just select a polygon and activate the **BandSaw** tool (**Multiply > Subdivide: More > Bandsaw**).

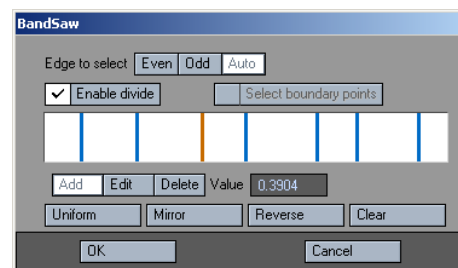


**NOTE:** If you select more than one polygon, the first selected polygon controls the operation. This tool is tailored for use on SubPatch objects in areas where you need more control points.

The operation continues all the way around until it returns to the point of origin or encounters a non-four-point polygon.



The **Edge to select** option determines the direction. With a quad, there are only two ways and these are described as **Even** or **Odd**. Select **Auto** for **Bandsaw** to pick the direction for you.



By default, there will be a single default slice right in the center. This is the positioning of the slice that will be made on the polygons. You can move it by selecting the **Edit** mode and dragging it with your mouse. You can also add more slices in the **Add** mode or remove them in the **Delete** mode.

To actually perform the slice, you need to activate the **Enable divide** option. However, you can also use **Bandsaw** to just select the band of polygons by not activating this option.



The **Select Boundary Points** option will select the points that make up the selected polygons. This option is only available when **Enable Divide** is unselected.

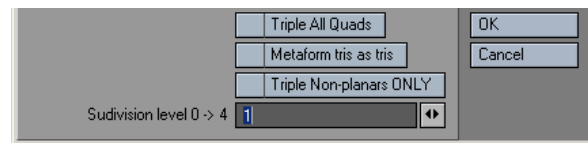
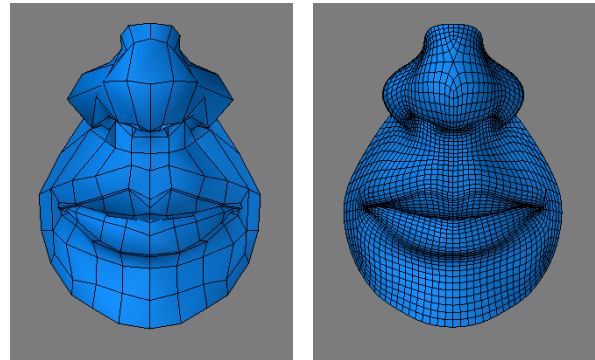
**Uniform** equalises the spacing between slices. **Mirror** will mirror the selected slice across the center. **Reverse** inverts the order—handy if you placed the slice(s) on the wrong side. **Clear** will remove all slices and set it back to default.



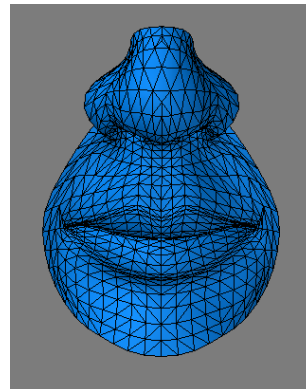
NOTE: The **BandSaw Pro** tool (**Multiply > Subdivide: Bandsaw Pro**), discussed previously in this section, is similar to the standard **BandSaw** function, but works interactively.

## Metaform Plus Tool

The **Metaform Plus** (**Multiply > Subdivide: More > Metaform Plus**) command will subdivide polygons with more than four sides (Ngons).

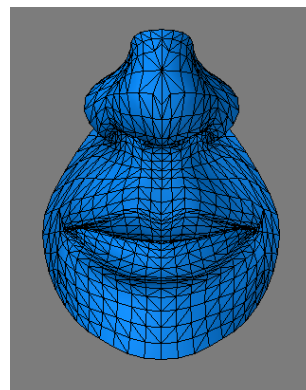


**Triple All Quads** option will triple all quads after the **Metaform** operation.



**Metaform Tris as Tris** will subdivide any 3-point polygons selected and keep the newly created geometry limited to 3-point polygons as well.

**Triple Non-planars Only** will triple only non-planar polygons after the **Metaform** operation.





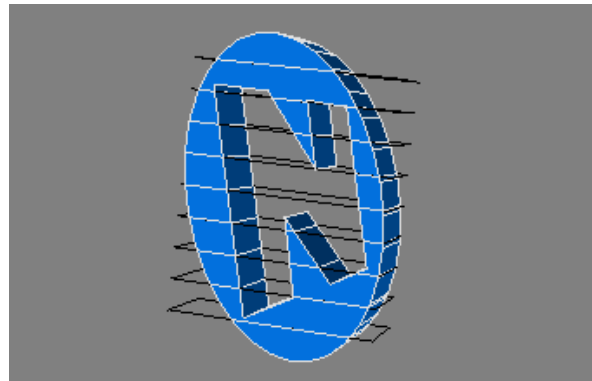
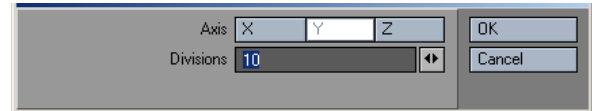
**Subdivision Level** field allows multiple levels of **Metaform** to take place at once. A value of 3, for instance, Metaforms the object three times.



NOTE: **Subpatch** mode is a real-time version of Metaform Plus.

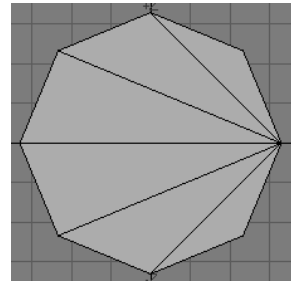
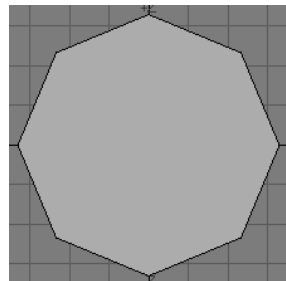
## Julienne Tool

The **Julienne** command (**Multiply > Subdivide: More > Julienne**) slices an object into sections that are regularly spaced along a given axis. **Julienne** is very useful for subdividing objects that will be bent or used with a **Displacement Map**. Use the **Division** setting to determine how many slices you would like to make on the selected **Axis**.



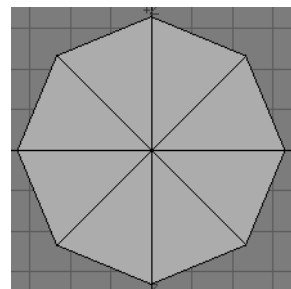
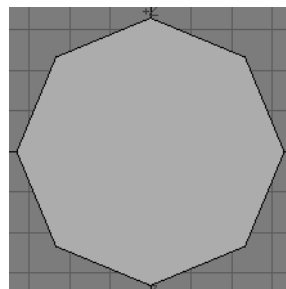
## Fast Triple Fan Tool

The **Fast Triple Fan** command (**Multiply > Subdivide: More > Fast Triple Fan**) triples polygons, with new polygons sharing a common point in a fan shape.



## Make Pole Tool

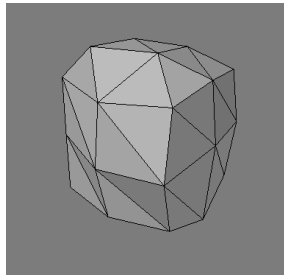
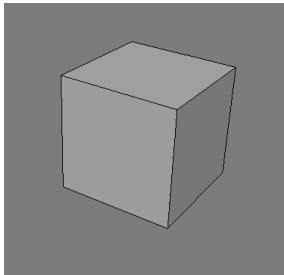
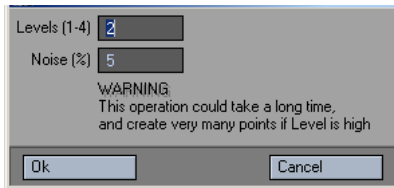
The **Make Pole** command (**Multiply > Subdivide: More > Make Pole**) triangulates each poly you have selected with a pole at the center. Normally, poles are generally avoided in a model, but if you use the **Make Pole** tool with the **Fix Poles** tool (**Detail > Fix Poles**), they can be a convenient way to close open geometry without too many smoothing abnormalities. Furthermore, merging the triangles in an octagon can result in a nice set of quads.





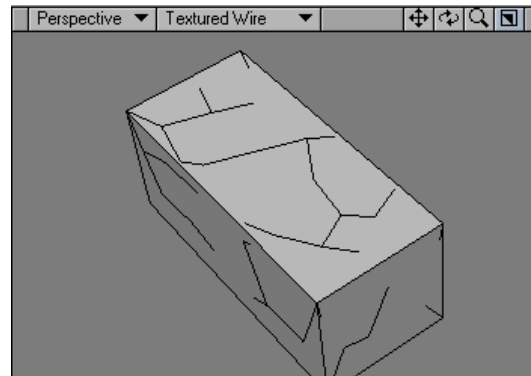
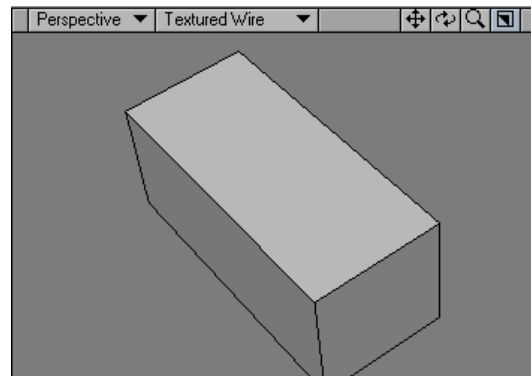
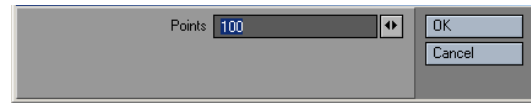
## Fractalize Tool

The **Fractalize** command (**Multiply > Subdivide: More > Fractalize**) will subdivide geometry unevenly in a fractal pattern. Set the Subdivision Level (1-4) and amount of Fractal noise on the **Fractalize** panel. This is great for creating landscapes or rocks.



## Random Pricks Command

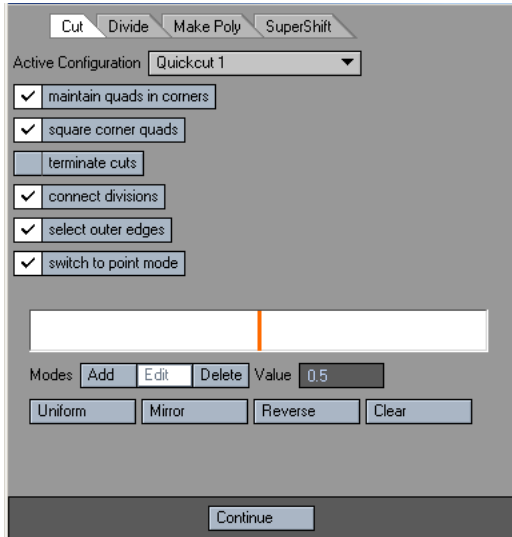
The **Random Pricks** command (**Multiply > Subdivide: More > Random Pricks**) creates random surface points on polygons. Use this to add detail to an object where you will later kill all of its polygons.





## Quick Cut 1, 2 and 3 Tools

**Quick Cut** (**Multiply > Subdivide: More > Quick Cut**) is simply a set of pre-configured **Cut** variants that allow you to cut quickly without going through a panel. They are designed to be used for commonly used cuts. They default to all having the same functionality - they need to be configured before the three start doing different things. To configure each **Quick Cut**, use the **Quick Cut Config** tool (**Edit > Edit Modeler Tools**) to configure each one to your preferences.

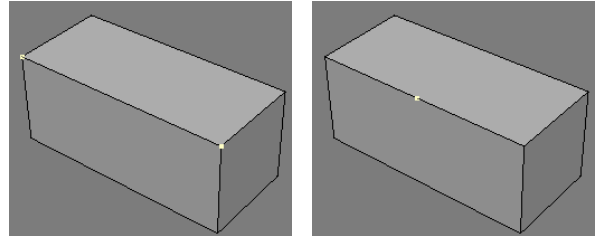


NOTE: See the **Multiply > Cut** tool for full details on QuickCut and QuickSaw, starting on page 801.

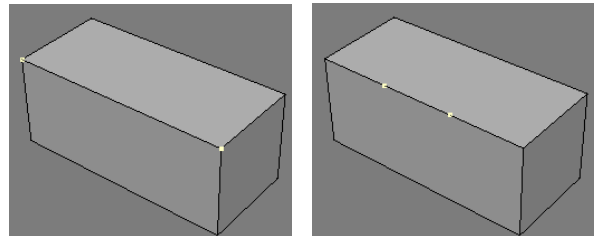
## Quick Saw Tools

The **Quick Saw** tools (**Multiply > Subdivide: More > Quick Saw**) are simply pre-configured variants of **Cut** (**Multiply > Subdivide: More > Cut**). Like the **Quick Cut** tools (**Multiply > Subdivide: More > Quick Cut**), these tools are quick in the sense that they do not require going through a panel interface to make cuts. Unlike the **Quick Cut** tools, these tools have hard-coded customisations and cannot be customised for new functionality.

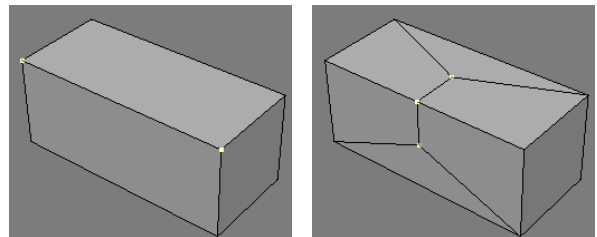
### Quick Saw:



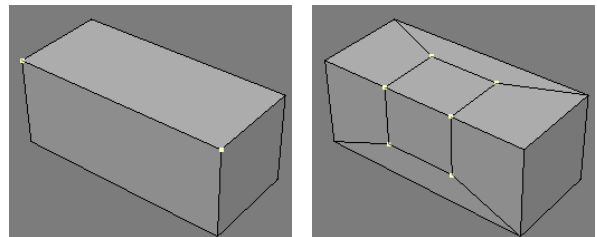
### QS Double:



### QS Term:



### QS Double Term:

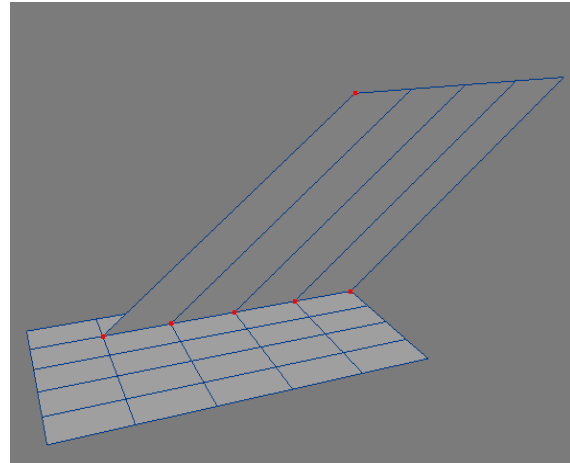
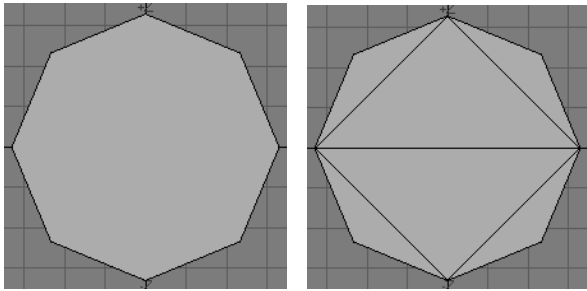






## Fast Triple Traverse Tool

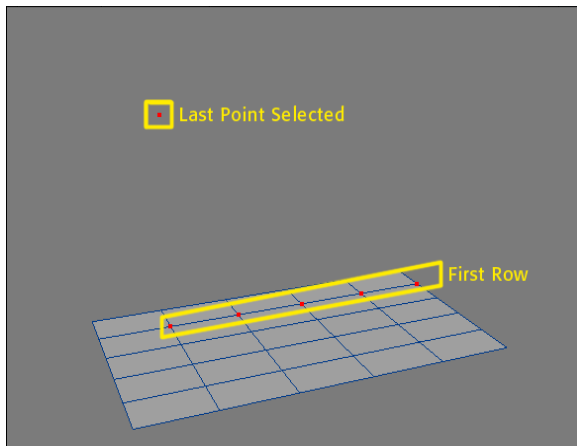
The **Fast Triple Traverse** command (**Multiply > Subdivide: More > Fast Triple Traverse**) triples polygons, with new polygons being created along the outside edge, moving inward.



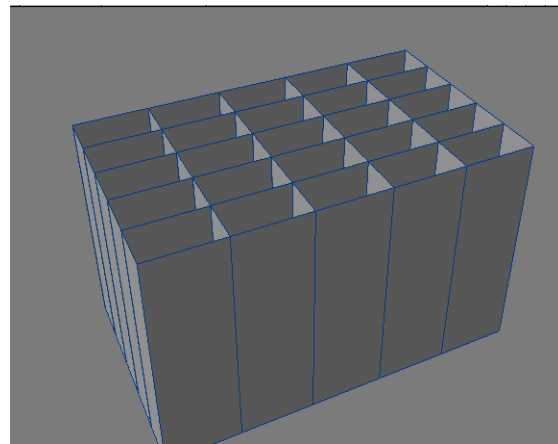
Here, the last point selected was at a different angle.

## Create Rows

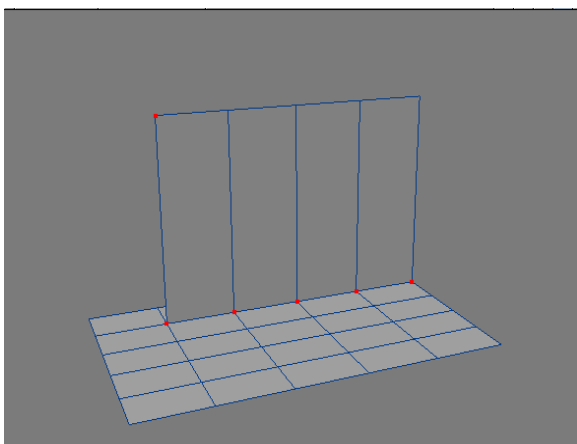
Creates a row of polygons based on the first row of selected points and the last selected point. The order of selected points on the first row does not affect the tool.



The first row of points is selected then the last point. The first row determines the number of polygons created. The first and last point selected determine the edge of the first polygon.



Multiple rows can be selected. Here, all points on the original grid were selected before the last point.





---

## Chapter 6: Construct Tab

---



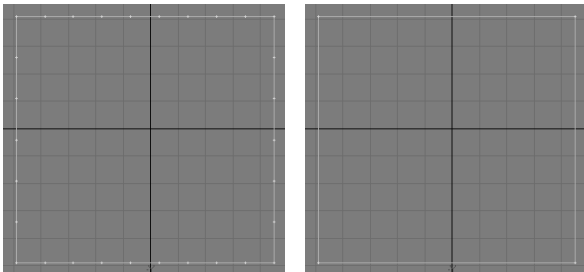
## Construct Tab

The Construct Tab contains a variety of tools for modifying geometry. These include ways to reduce the number of points and polygons in all or part of the geometry, and ways to shape the geometry by drilling holes or stenciling the mesh with new points and edges, or connecting meshes. Tools for patching splines, for converting metaballs and metaedges, for toggling Subpatch mode, and for freezing subdivision surfaces into a polygonal mesh are also located here.

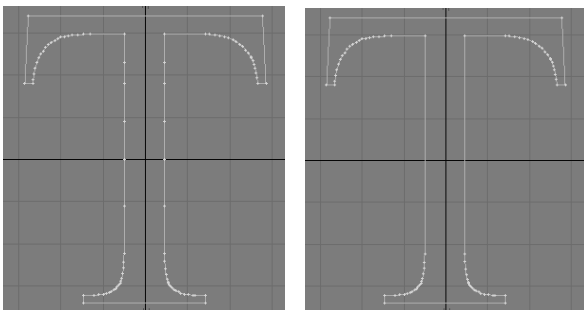
## Reduce

### Reduce Points Command

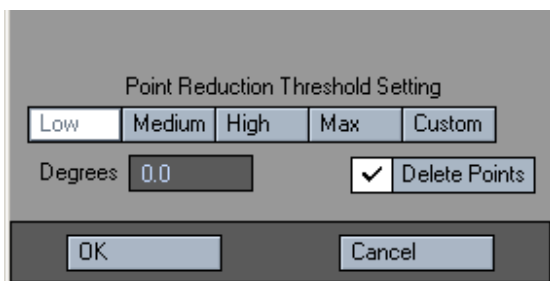
The **Reduce Points** command (**Construct > Reduce-Points**) traverses polygon edges and removes any points (from the polygons) that form an angle with its adjacent points that is less than the **Degrees** setting.



This command is especially useful to reduce the number of points and polygons generated by extruding or bevelling text objects.



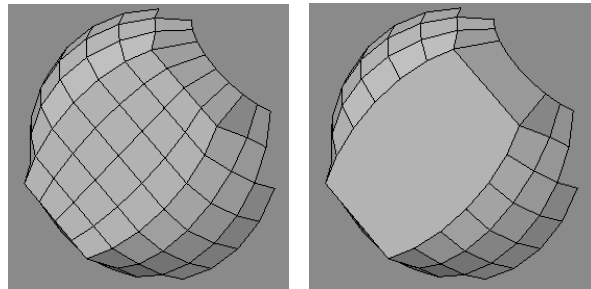
Simply select a **Point Reduction Threshold** setting and the **Degrees** will be set automatically.



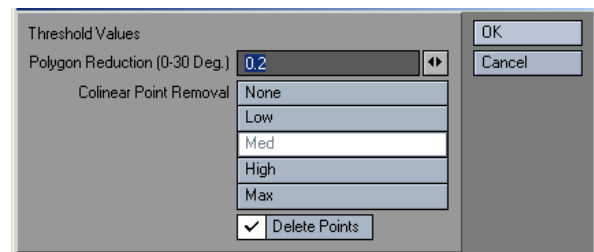
You can also enter your own **Degrees** setting. Activate **Delete Points** to delete the removed points.

## Reduce Polygons Command

The **Reduce Polygons** command (**Construct > Reduce-Polygons**) merges co-planar polygons with three or more vertices into a single polygon and can optionally remove any stray points resulting from the polygon removal.



Polygons are merged if the relative angle between two polygons is less than the **Polygon Reduction** setting, they share an edge, and have the same surface name.



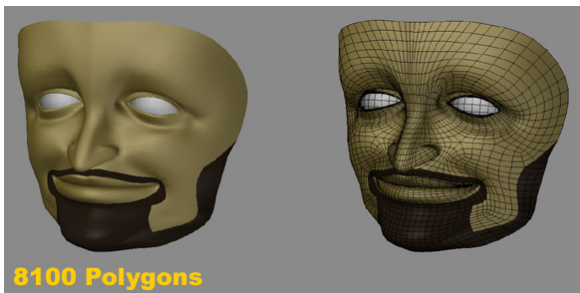


## Reduce Polys + (Qemloss2)



**WARNING:** Always save your objects before you run **Reduce Polys +!**

The **Reduce Polys +** function (**Construct > Reduce Polys**) uses a surface simplification algorithm in an attempt to reduce the number of polygons in an object. The plugin lets you rapidly produce good quality approximations of excessively detailed polygonal models. **Reduce Polys +** is an excellent way to create low resolution *stand-in* objects for scene layout or for multi-resolution models. You can use the object replacement capability in Layout to substitute a low resolution object in place of an object with a high polygon count when it is far away from the camera.

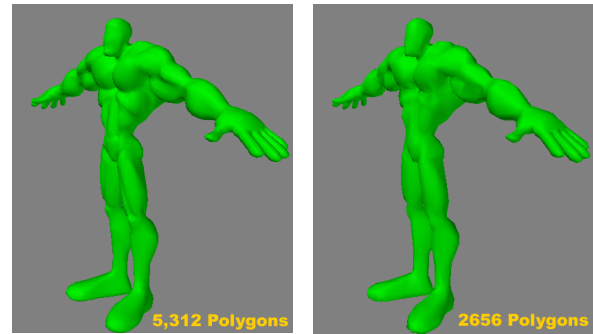


This plugin uses routines adapted from Michael Garland's public domain QSLim Simplification Software. The algorithms used in this software are described in the papers written by Michael Garland and Paul S. Heckbert, *Surface Simplification Using Quadric Error Metrics*, SIGGRAPH 97, and *Simplifying Surfaces with Color and Texture using Quadric Error Metrics*, IEEE Visualization 98.

### Introduction

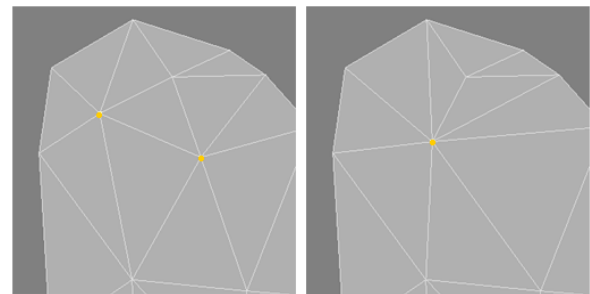
Many 3D models contain a large number of polygons, especially algorithmically created models (such as those created by implicit surface construction techniques) and models created with 3D scanners and digitisers. Obviously, rendering 3D scenes is much faster when the models contain a minimum number of polygons. Also with the current trend towards sharing 3D worlds over the Internet, level of detail models (LOD) are becoming absolutely necessary for creating worlds in which you can browse and interact in a reasonable manner.

This tool provides polygon reduction on objects within Modeler. Only one parameter, the reduction **Goal**, must be set by you, the remaining default values should provide good reduction for many objects with a high polygon count. However, to get the best results, you must understand some of the basic concepts behind the algorithm.



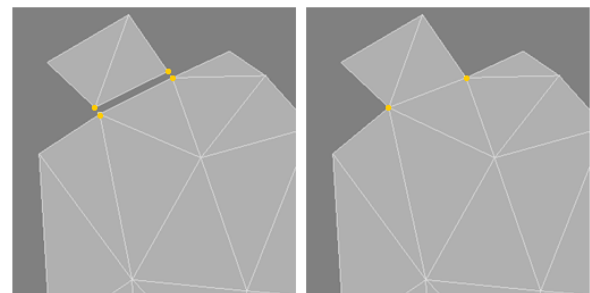
### Terminology

The simplification algorithm is based on contractions of vertex pairs. It supports two types of contractions: **edge** and **non-edge contractions**. An **edge contraction** occurs when the vertex pair shares an edge. This is the primary type of contraction that occurs during the reduction stage (in fact, **non-edge contractions** are turned off by the default parameters). The following figure shows an example of an **edge contraction** where vertex v1 and vertex v2 are joined to form a new vertex v3. Since v1 and v2 share an edge (highlighted below), one or more triangles will always be removed during this contraction. In this example, two triangles are eliminated from the mesh.



Edge Contraction

**Non-edge contractions** let the algorithm join unconnected areas of the object together. The next figure shows an example of a **non-edge contraction** (also called *aggregation*).



Non-Edge Contraction

In fact, there are two **non-edge contractions** taking place: v1 and v2 contract to form v5; v3 and v4 contract to form v6. Since by definition the vertices in the contraction pairs do not share an edge (there is no edge between v1 and v2, or the v3, v4 pair), there is no actual reduction in the polygon count of the object. However, since this feature joins previously unconnected areas of the object together, the potential exists for future reductions. Many times this can also provide a better low resolution approximation of an object that has many disconnected regions.

As the algorithm proceeds through each iteration, an approximation of geometric error accumulates at each vertex of



the object. If this geometric error is less than the user defined **Maximum Error Tolerance**, the vertex is marked as a viable candidate for another contraction. Once the geometric error for a vertex becomes greater than the maximum threshold value, it will no longer be considered in any more contractions. During each iteration, the vertex pair with the smallest combined geometric error is chosen for the current contraction.

The algorithm proceeds until the simplified object is reduced to your targeted **Goal** (number of polygons) or until all the vertex errors have become greater than the **Maximum Error Tolerance**. These two parameters control how much reduction will take place, the remainder of the parameters control various aspects of the vertex contractions.

Two types of vertices receive special consideration by this plugin: **surface border** and **boundary vertices**. Parameters are provided for you to weigh the geometric error for these two special types of vertices:

- A **surface border vertex** is a point that is shared by two or more surfaces.
- An edge that exists in only one triangle is a **boundary edge**, and determines two endpoints that are called **boundary vertices**.

The **Surface Border Weight** and **Boundary Preservation Weight** parameters let you weigh the geometric error at these points. The higher the weight, the less likely the vertex will be replaced. Clever use of these parameters (along with some equally clever surfacing) can provide quite a bit of control over the contraction process. The **Polygon Area Weighting** parameter, causes every vertex's geometric error to be weighted by the area of the polygons that contain the vertex. Once again causing larger values (triangles with larger areas) to be less likely to be removed.

To help preserve the shape of these shared borders, weighting of the surface border vertices, using **Surface Border Weight**, has been added to the algorithm to help restrict movement/replacement of those vertices.

## Using Reduce Polys +

Make sure you have an object in the current foreground layer(s) of Modeler. You also need at least one empty layer, because the existing object remains unchanged, and the reduced object is placed in the first available empty layer. Choose **Construct > Reduce Polys +** and you are presented with the following panel.



polygons.

NOTE: **Reduce Polys +** ignores any polygon selections and works only on the entire object in the foreground layer, including any hidden

**Goal** lets you set the final number of polygons you would like in the simplified object. You may enter either a desired polygon count (an integer such as 1000), or a percent based on the number of polygons found in the original object (a real number with a percent sign at the end, such as 65.2%). If you enter a percentage, it will simply calculate the polygon goal by multiplying that percentage by the total polygon count in the original object. So 100% will mean no reduction takes place, and 0% means the object will disappear completely.

The first major step that Reduce Polys + takes is to triple your polygons, which will increase the polygon count, if your model contains non-triangular polygons. A **Goal** of 100%, however, will reduce the number of polygons back to the original number, but now they will all be triangles. If that reduction doesn't preserve your model's shape well enough, don't hesitate to try percentages over 100%, those models will still be smaller than the tripled original.



NOTE: Other **Parameter** settings may cause the algorithm to fail to reach the reduction goal.

**Maximum Error Tolerance** sets the geometric error threshold. A large value (such as the default 1,000,000) almost guarantees that you will achieve the reduction goal in the previous parameter field. Smaller values may preserve the shape of the original object better, however you may not achieve the desired reduction goal entered above.

**Surface Border Weight** gives you some control over how often and far surface border vertices are relocated during the reduction process. A large value restricts the relocation of points that lie along surface borders, thereby preserving the shape of the border fairly well. Smaller values let the points move farther from their original location, possibly causing the border to change its shape. A value of 0 will not constrain the surface borders at all, and vertices along a border may not end up where you expect when the simplification is finished.

**Boundary Preservation Weight** allows you some control over how boundary vertices are relocated during the contraction process. A small value lets the boundary points move farther than larger values. A value of 0 will not constrain the boundary at all, and the object may not look at all like you expect when the simplification is finished.

If you end up with unexpected gaps between non-contiguous polygon surfaces, you can try to preserve those boundaries by increasing the **Boundary Preservation Weight**. However, you might get better results by merging the edge points together before running **Reduce Polys +**. You can always just cut and paste the polygons afterward, if the separation was essential.

**Pair Selection Tolerance** determines whether non-edge contractions are performed during the simplification process. If this value is 0, non-edge contractions are turned off, and only edge contractions will take place during the simplification. Any value greater than 0 will cause non-edge contractions to be possible during the reduction. If you enter a negative number, Reduce Polys + will automatically use 5% of the radius of the



object's bounding sphere. This is the value you should use if you want to start experimenting with this parameter.

Take care when changing this value to anything but 0! It is strongly recommended that you leave this at 0 for all complex models with lots of polygons. If you use this, first reduce the model to a fairly small polygon count with it turned off (=0), then reduce the reduction again with a carefully chosen **Pair Selection Tolerance**, or better yet, just use a negative number. It is a very memory-intensive operation.

**Vertex Placement Policy** is best left at **Optimal**. When a pair of vertices is contracted, the algorithm must decide where to locate the new vertex. The algorithm can use any one of the three final options listed in this selection of buttons. **Optimal** calculates the new placement based on the location with the least amount of geometric error. It doesn't hurt to experiment with these options though.

Sometimes vertex pair contractions do not preserve the orientation of the faces in certain areas of the contraction. If **Preserve Mesh Quality** is changed to **Yes**, the normal of each neighbouring face is compared before and after the contraction. If the normal flips, the contraction is penalised greatly by making the geometric error for that contraction very large (so that contraction will probably never take place). In most cases, this will not be a problem, and the plugin will work faster if this parameter stays set to **No**. It's not terribly slow, so it certainly doesn't hurt to experiment with it.

If you change **Polygon Area Weighting** to **Yes**, the area of the triangle containing the vertex is used to weigh the geometric error. This will cause larger triangles to increase the geometric errors of its vertices, so it is less likely to be chosen for simplification.

The **Reduce Polys +** function displays a progress monitor while it creates the reduced object in the first available empty layer, following these steps: first, the object in the foreground layer(s) is copied to the first available empty layer, then all its polygons are converted to triangles using Modeler's **Triple** command. Next, all vertices and polygons are converted into the necessary data structures needed for the simplification routines, and the copied object is subsequently removed. Once the simplification routine finishes, the reduced polygon object is placed in the previously empty layer.



NOTE: For additional information, check out <http://amber.rc.arizona.edu/lw/>.

## Dissolve

Similar to Merge Polys, but works on all elements. Selected edges then applying Dissolve will merge the polygons used by the selected edges.

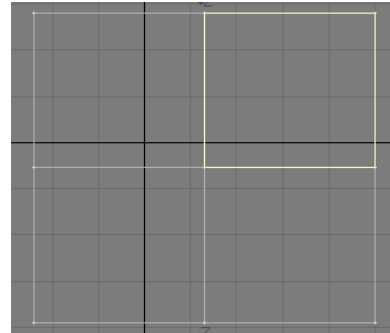
## Remove

### Remove Vertices from Polygons

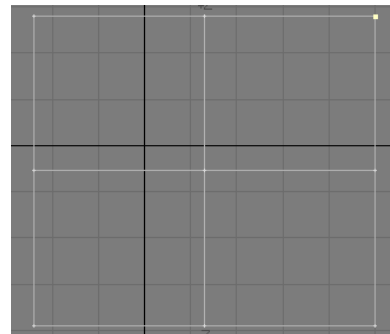
The **Remove Points** command (**Construct > Remove Vertices From Polygons**) *disassociates* the selected point(s) from the selected polygon, but does not delete the point. Non-selected polygons that used that point remain unchanged.

**To remove a point from a polygon:**

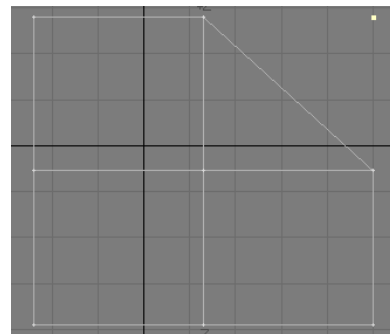
**Step 1:** Select the polygon(s) containing the point to be removed (**Polygon Selection** mode).



**Step 2:** Select the points to be removed (**Point Selection** mode).



**Step 3:** Choose **Construct > Reduce: Rem Points**. To actually delete the points, use the Cut function.







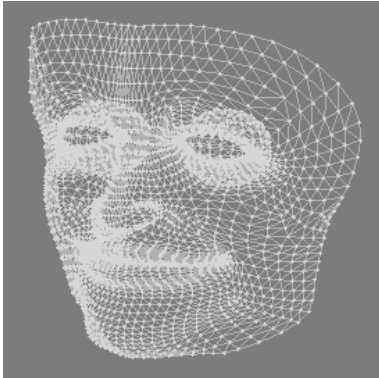
## Remove Polygons Command

(default keyboard shortcut **K**)

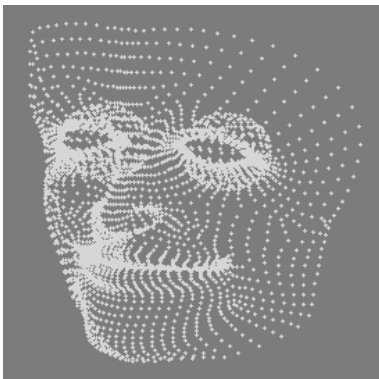
Frequently, you want to use some of the points of an object (often primitives) in creating your objects. A good example is using part of a disc's points to model a rounded corner. The way to do it is to use the **Remove Polygons** command (**Construct > Remove Polygons**).

*To remove polygons but not points:*

**Step 1:** Select the target polygons.



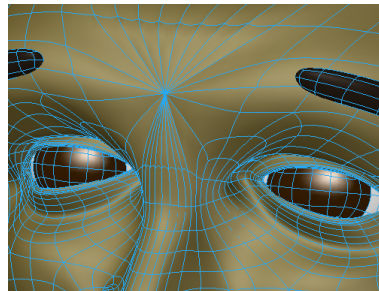
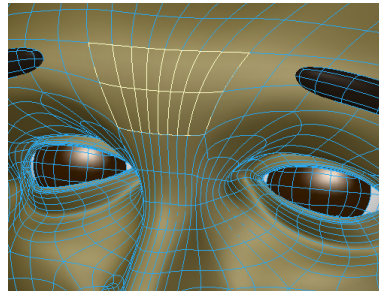
**Step 2:** Choose **Construct > Reduce: Rem Polygons** or press the **K** key.



HINT: An easy way to remember this keyboard shortcut is to think of it as “killing” (K) the polygons.

## Collapse Polygons

The **Collapse Polygons** command (**Construct > Collapse Polygons**) replaces selected polygons with a single point at their average position. It is like moving all of the points for the selected polygons to their center and then merging the points.



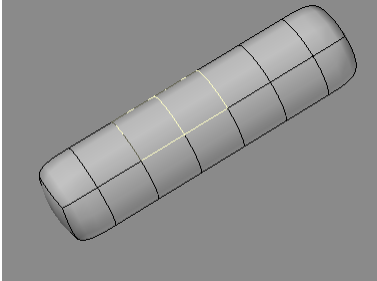




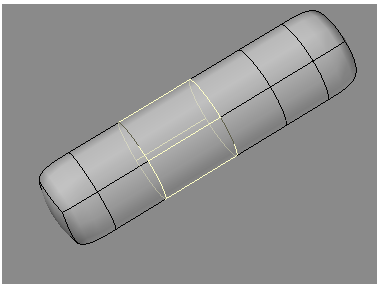
## Bandglue

The **Bandglue** command (**Construct > Reduce : Bandglue**) merges a contiguous row of four-point polygons into one polygon. This is a great tool to use to clean up extra rows of geometry. **Bandglue** is the opposite of **Bandsaw**.

To use, first select a contiguous row of (two or more) polygons.



Then execute the command. Like **Bandsaw**, the operation continues all the way around the geometry until it returns to its point of origin, or hits a non-four-point polygon.



## Combine

### *Faster and Better Drills and Booleans*

Complex **Drill** and **Boolean** operations require a lot of computing. Selecting the affected polygons in the foreground layer before using either command will greatly speed things up. This lets Modeler concentrate on only those polygons that need to be affected. Also, if your initial result is not satisfactory — particularly with complex objects — try it again after nudging one of the objects on an axis a little.

## Boolean Command

(default keyboard shortcut **Shift B**)

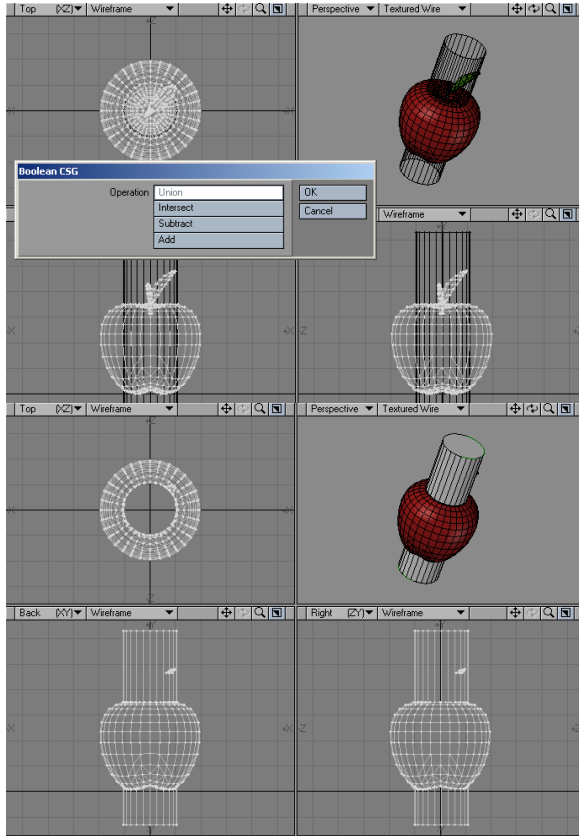
The **Boolean** command (**Construct > Combine: Boolean**) is similar in some respects to the **Solid Drill** command, discussed later. However, **Boolean** operations treat objects as solids with volume, instead of hollow 3D objects. They will merge, split, carve, and join objects in a variety of ways. Like the drilling tools, you must have an object in the background layer. The major distinction of **Boolean** is that it treats the foreground object as a solid mass and thus creates inside surfaces along cut edges.

With **Boolean**, some portion of the foreground and background objects must physically overlap in 3D space.



## Union

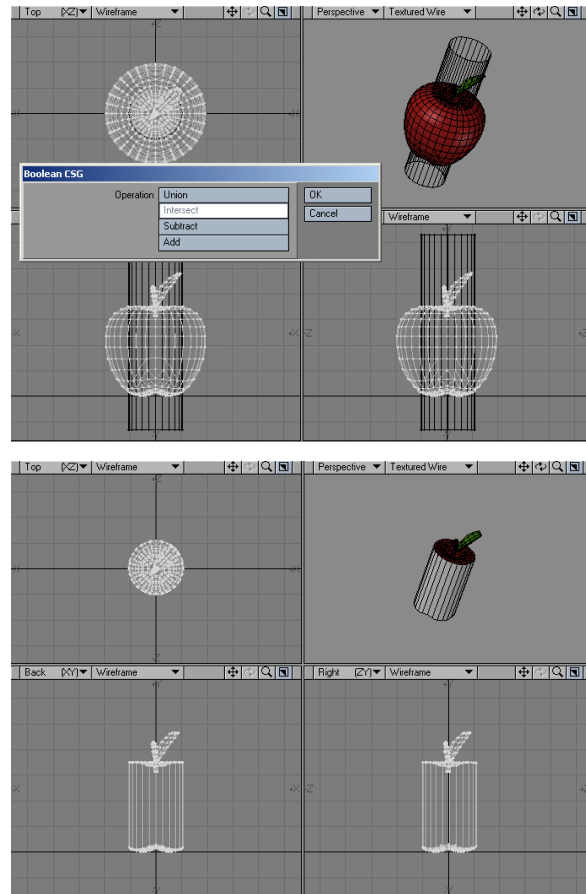
**Union** joins the background layer with the foreground layer, taking the solid objects and combining them into a single object. Interior faces are removed and no new polygons or surfaces are created; therefore, all surfaces will retain their original names. To work properly, both objects must be closed 3D solids.



From the rendering perspective, you could get a similar result by just pasting both objects into one layer; however, **Boolean** results in the fewest polygons and smaller object files, which will use fewer system resources and render faster. There may be times, however, that due to the complexity of objects, **Boolean** will not produce acceptable results.

## Intersect

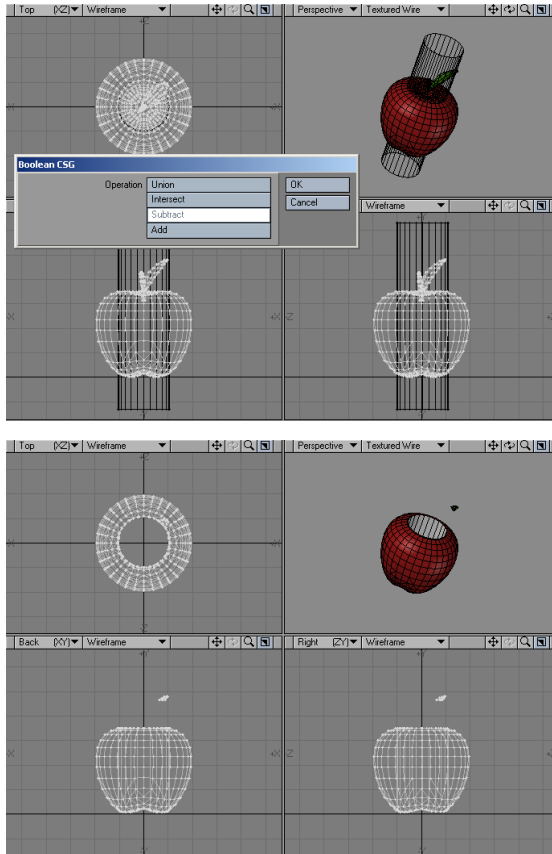
**Intersect** takes solid objects and leaves behind whatever portion was common between them. It leaves behind only those portions of the background layer and foreground layer that were overlapping. (This corresponds to the portion that the **Union** operation discards.) No new polygons or surfaces are created; therefore all surfaces will retain their original names. To work properly, both objects must be closed 3D solids.





## Subtract

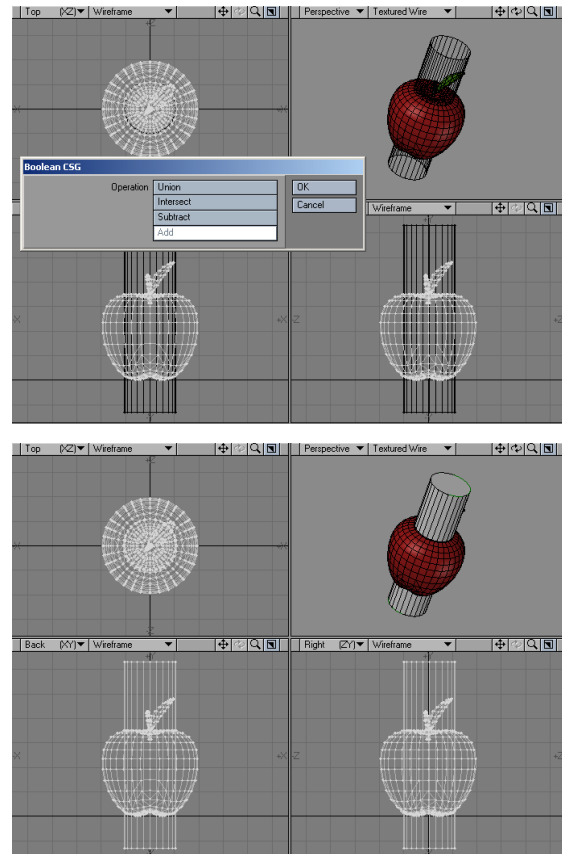
**Subtract** removes the background drilling template object from the foreground object, leaving a copy of itself embedded there. Essentially, it carves the background layer out of the foreground layer, leaving new inside surfaces in its wake. New surfaces will take on the names of the surfaces that carved them. To work properly, both objects must be closed 3D solids.



The results of this operation will vary depending on whether the polygons being drilled consist of single-sided or double-sided polygons (see subsequent discussion).

## Add

**Add** joins the background layer with the foreground layer so that the overlapping shapes are fully combined as one. You can use combinations of 2D and 3D objects. This is different from simply saving two overlapping objects from two different layers, since the polygons that physically overlap are merged in a form of mutual drill operation. No new surfaces are created, although some polygons may be subdivided. Therefore all surfaces will retain their original names.



The result is similar to just cutting and pasting the tube object into the apple layer. The difference is that where surface polygons from the two objects intersect they are sliced by each other to form a common edge. All interior polygons and surface names are retained.

You can use more than two objects at a time, but it's often better to work with a pair at a time. The math involved in **Boolean** operations is complex, and using multiple objects in a single operation is quite time-consuming.



### To use the Boolean command:

**Step 1:** Make the object that you are going to use as the cutter.

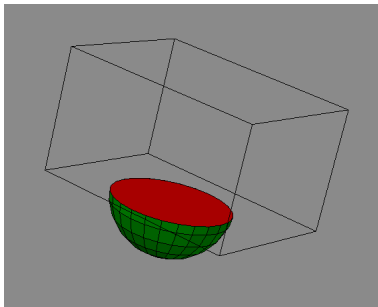
**Step 2:** Place the object to be cut in another layer and put the object from step 1 in the background.

**Step 3:** Click Construct > Boolean (or use the keyboard shortcut **Shift B**).

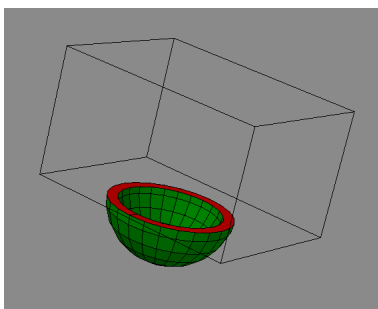
**Step 4:** Select the type of Boolean Operation and click OK.

### Single vs. Double-sided Polygons

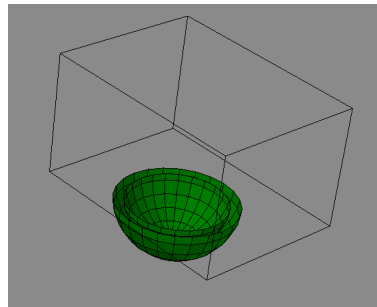
Single-sided and double-sided polygons play a unique role in the **Boolean Subtract** operation. Imagine that you are a knife blade, slicing through a 3D sphere made of single-sided polygons. Every time you cut through a polygon layer, you pass from the outside of the sphere to the inside. If you passed through only one polygon, you are now inside the 3D sphere. Accordingly, a Boolean knife blade slicing through a one-sided sphere would leave solid halves.



Now say there are two spheres made of single-sided polygons, one sphere being smaller than the other, with the smaller one inside the other (normals facing in). You pass through the first polygon — from air to solid — then through a second polygon — from solid back to air. You would now be inside a hollow core inside the inner sphere. A Boolean knife slicing through this sphere would leave a hollow core.



Now, if our two spheres were double-sided instead, Modeler would treat each sphere as if it were constructed with paper-thin walls (double-sided polygons are just back-to-back polygons). Remember, normally the sphere would be treated as a solid ball. Upon encountering a double-sided polygon, Modeler will think that it has passed through two polygons. In the paper-thin space between the front-facing and back-facing polygon, the Boolean knife blade sliced through air, then solid, then air again.



**NOTE:** **Boolean** will usually not operate as expected if the foreground and background objects are both double-sided.



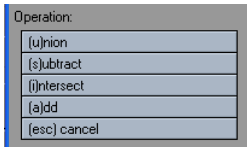
**HINT:** Boolean commands such as **Subtract** will produce much better results if your objects contain plenty of geometry to begin with. If a boolean does not appear to produce favourable results, or generates an error when run, then consider increasing the amount of polygons with tools such as **Multiply > Subdivide: Subdivide** before performing the boolean operation.



## Speed Boolean

Speed Boolean (Utilities>Plugins>Additional) works like the Boolean command except you work in one layer.

For Booleans to work properly in LightWave, you need fully connected, closed objects, and on two separate layers. What Speed Boolean does is take your polygon selection, expand it to Select Connected, then it Cuts the selected mesh, Pastes it on a free background layer and executes the desired Boolean operation, Deletes the background layer then switches back to foreground.



Note: Speed Boolean requires that you have both meshes in the same layer to function. You must also select at least one polygon on the mesh doing the Cut.

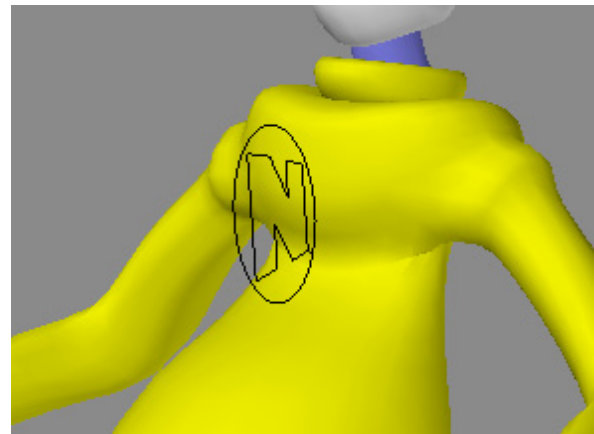
## Drill Tools

(default keyboard shortcuts: Solid Drill **Shift** C Drill **Shift** R)

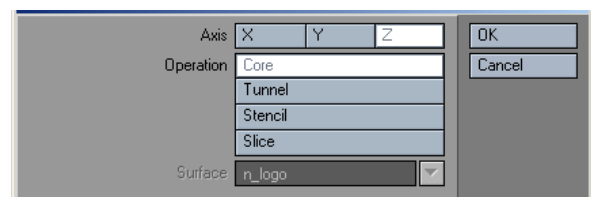
There are two *drill* commands: **Drill (Construct>Combine: Drill)** and **Solid Drill (Construct>Combine: Solid Drill)**. The difference is that with **Drill** you use a two-dimensional object as your drill bit. The bit extends infinitely along the drilling axis, so it has an imaginary third dimension. Think of it as looking straight down along the length of a drill bit. Forgetting the fact that you know it has length, it would look like a disc, right? With the **Drill** command, you just need a disc-shaped polygon and the length is assumed.

**Solid Drill** is essentially identical to **Drill** except — as you might have guessed — it uses an actual three-dimensional object as your drill bit.

When you use a drill in the real world, you must first place a drill bit in the drill. This same concept applies to Modeler. Modeler uses an item in the background layer as the drill bit that will cut the item in the foreground layer(s).



NOTE: If your drill bit is an open curve, Modeler will treat it as a closed curve with its endpoints joined.

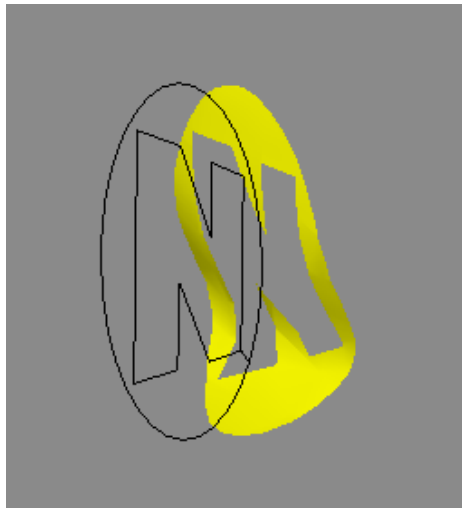




Both **Drill** tools have the same four options:

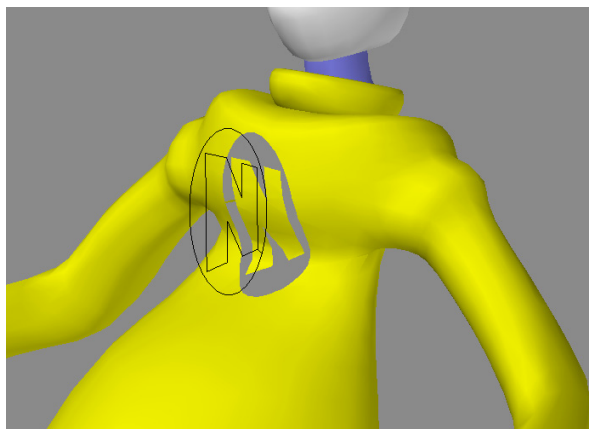
### Core

The **Core** operation will include all polygons that reside inside the 3D space of the drill bit, plus portions of the polygons that straddle the borders.



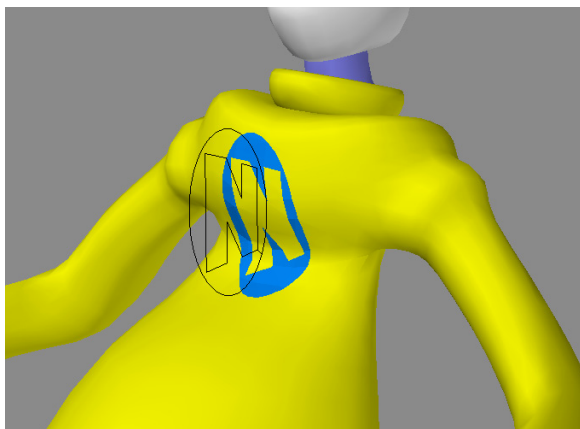
### Tunnel

**Tunnel** is the opposite of **Core**. It creates a *tunnel* in your object. All polygons that reside outside the 3D space of the drill bit will be included, plus the trimmed polygons that straddle the borders.



### Stencil

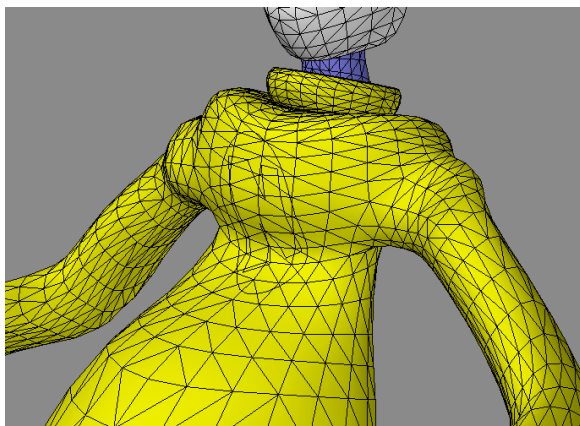
The **Stencil** option stencils the drill bit's shape onto any intersecting polygons. It also names all of the polygons that fall within the bit's shape. You can either enter a surface name in the field or select an existing name from the pop-up menu.



This option is great for stenciling text or logos onto objects—an alternative to using surface color textures. To help draw attention to a stenciled area, you might wish to use the **Bevel** tool (**Multiply > Bevel**) or **Smooth Shift** tool (**Multiply > Smooth Shift**) on the stencil.

### Slice

**Slice** is the same as **Stencil**, except it does not rename any surfaces.



### To drill an object:

**Step 1:** Place the target object in a layer.

**Step 2:** Place your bit object in another layer and put the target object in the background.

**Step 3:** Position the bit object as desired. Use the viewport whose perpendicular axis (i.e., the axis you can't affect) is the one you want to drill along. For example, if you wanted to drill along the Z axis, position the bit object in the Back type viewport.

**Step 4:** Swap foreground and background layers using the apostrophe key (') shortcut.

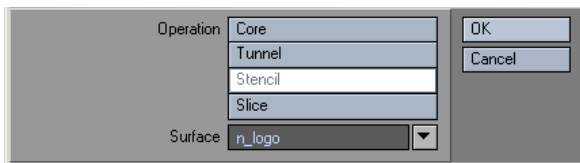
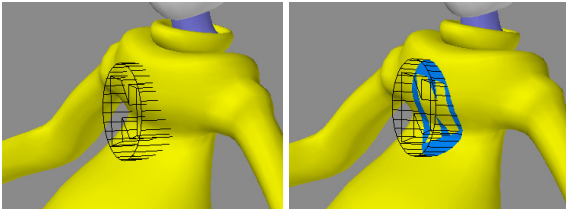
**Step 5:** Choose Construct > Drill.

**Step 6:** Select the drilling **Axis** and the type of drilling **Operation**. Click **OK**.



## The Solid Drill Tool

The **Solid Drill** command (**Construct > Combine: Solid Drill**) will work just like the **Drill** command, but you have control over the effect along the drilling axis. There is no **Axis** option. It is unnecessary since you use a 3D object that has no *implied* third dimension.



NOTE: The objects in the foreground and background layers must physically overlap for the tool to work properly. They cannot be just lined up on an axis.



NOTE: Both the drill bit and the item to be drilled must be closed solids. A hemisphere with one open side, or a hollow tube, would not qualify, but a closed hemisphere or a tube with sealed ends would.

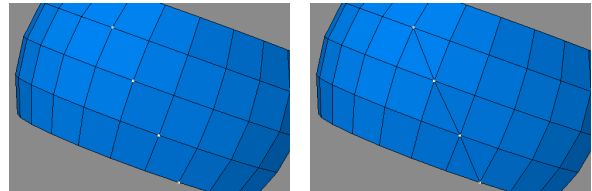
## Bridge Tool

(default keyboard shortcut **L**)

The **Bridge Tool** is located under **Construct > Combine: Bridge**.

### Point Mode:

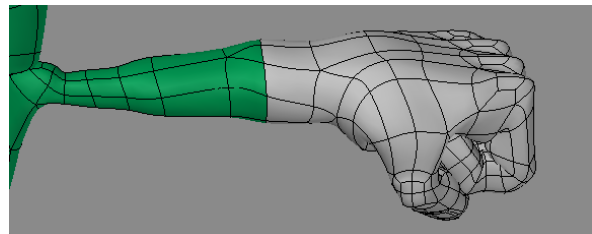
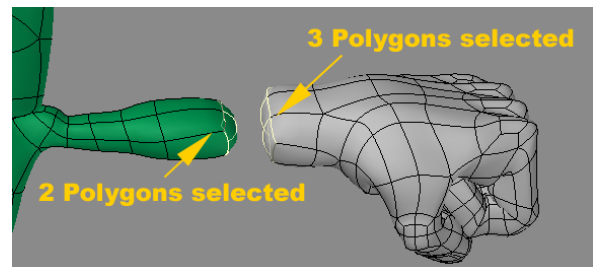
In **Point Mode**, the tool works identically to LightWave's **Split Poly** tool, except with one convenient modification: you don't have to select the polys you want to cut ahead of time.



NOTE: When in **Point Mode** the **Bridge** tool stores your poly selection, select all polys, run split polygons, then restore the previous selection. As such, the tool has the same problems with split polys when trying to do elaborate cuts — it's recommended that you keep it simple and use the tool multiple times rather than try to make a very complicated series of cuts at once. This tool works well in combination with **Divide**.

### Polygon Mode:

In **Polygon Mode**, the tool serves as a connect tool to form bridges/tunnels between polygons. Multiple polys can be selected and bridged together, with or without matching border shapes/point counts.



It is recommended that the borders have point counts that are fairly close, otherwise the tool will perform heavy triangulation to bridge the polygons which will often give undesired smoothing results without modifications. When bridging together polygons, the tool will automatically use a seam point, which will result in the minimal amount of twisting and turning of edges bridging the two polys together. If the point counts don't match, the seams are fitted at an angular tolerance level with triangles bridging the additional points on the border with the higher point count.





## Patches

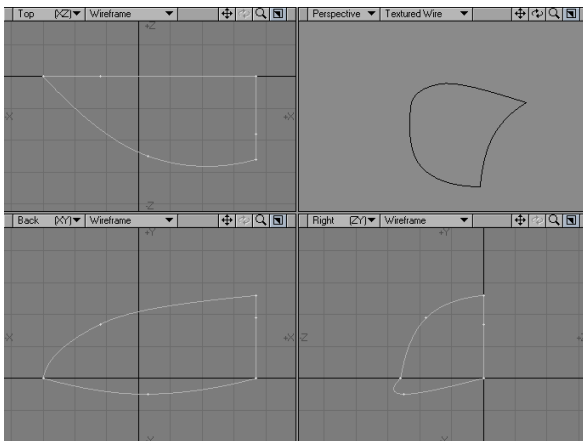
### Patch

(default keyboard shortcut **Ctrl** **F**)

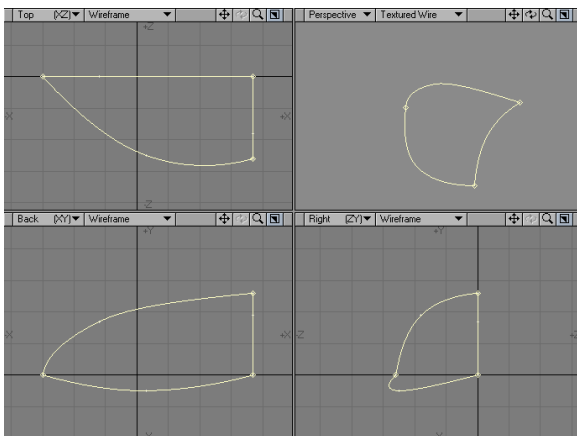
Modeler's feature for **Spline Patching** lets you model with curves and then smoothly spread a mesh of polygons over them.

#### To use Spline Patching:

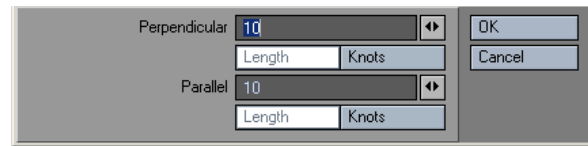
**Step 1:** Make three or four curves that create one or more fully enclosed areas. Note that the curves must share end points, and that they must be "head to tail" meaning that the origin point of one curve is the same point as the end point of the preceding curve.



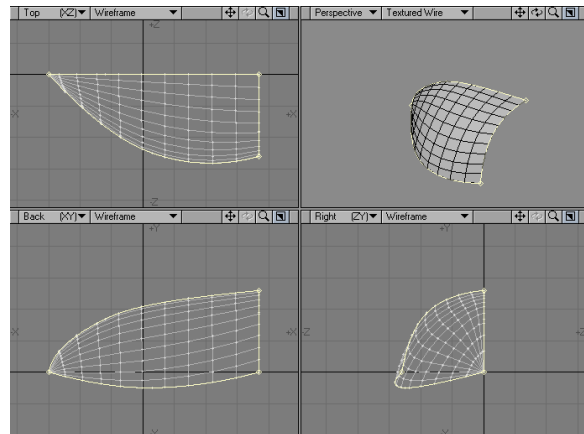
**Step 2:** In the **Polygon Selection** mode, select the curves that surround the enclosed area. Do this in a clockwise or counterclockwise fashion. Remember which curve you selected first! It's okay if you also select some polygons in addition to the curves, they will be ignored in the patching operation.



**Step 3:** Choose **Construct > Patches: Patch**. The fields for **Perpendicular** and **Parallel** refer to the number of polygons to put along the side that is perpendicular or parallel to the first curve you selected. **Knots** and **Length** let you choose to space your polygons equally according to the overall length or relative to the knots or points on the splines.



**Step 4:** Click **OK**. You should see a polygon mesh added to the area within the three curves you selected.



Sometimes a spline cage will be patched unexpectedly. There are some fixes you can try. All of the selected curves should flow in the same direction, based on their heads and tails. If they don't, you can flip them and try the patch again. You can also try selecting the curves in the opposite direction.

Sometimes the resulting surface normals face the wrong way. Just select at least one polygon on that side. Choose **View > Selection: Connected** to select all of the connected polygons in the patch. Then, press the **F** key to flip the polygons.

If there are not enough curves or they do not share points properly, an error message appears. If this happens, check the joining points carefully to make certain they are not merely overlapping. Choose either **Detail > Points: Merge Points** or **Detail > Points: Weld** to correct.

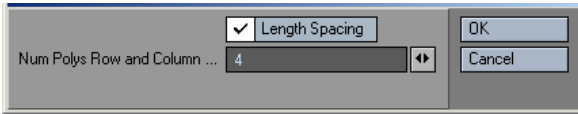


**NOTE:** When patching a 3-sided spline patch, one corner will always contain a fan of triangles. The corner where these triangles will be created is determined by the connecting corner of the first and second spline selected.



## Auto Patch

The **Auto Patch** command (**Construct > Patches: Auto Patch**) automatically patches a spline cage. You can set the number of rows and columns of the patch between 0 and 20.



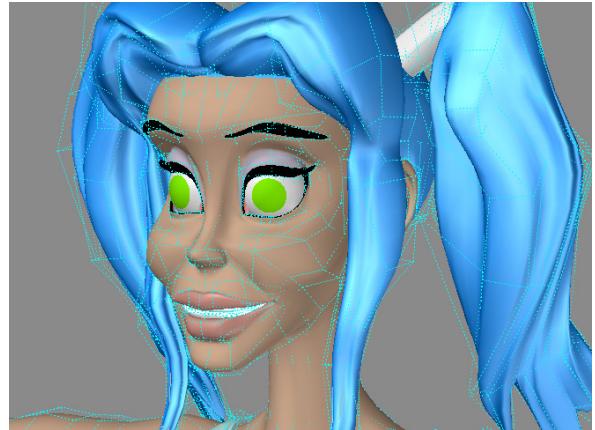
To space the polygons according to the length of the curve, select the **Length Spacing** checkbox. To ensure success, never have more than four knots in a curve.

## Convert

### Subpatch

(default keyboard shortcut **Tab**)

Standard polygonal surfaces need many polygons to approximate a smooth surface. Even so, a smooth surface made up of polygons will eventually reveal its inherent sharp-edged nature, if it's examined closely enough. Although you can create extremely polygon-heavy objects to attempt to compensate for this, it is often difficult to manipulate and manage them from the perspective of memory-consumption and editing.



A “SubPatch” - an abbreviation for subdivision patch - is a “bi-cubic patch”. The idea behind **SubPatching** is to repeatedly refine the control mesh until you achieve a smooth surface, called the “limit surface.” The **SubPatch** Modeling mode is not unlike a real-time implementation of the **Metaform** (**Multiply >Subdivide**) command.

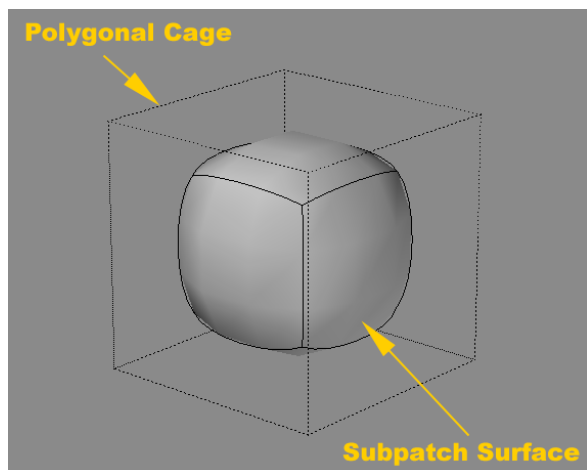


You activate **SubPatch** mode by pressing the **TAB** key or choosing **Construct > Convert: SubPatch**. In most instances you want to affect the entire object, so no polygons are selected when you press the key. If polygons are selected, only those will have the **SubPatch** mode turned on. This can get confusing because the polygons become unselected. If you press the **TAB** key again, the **SubPatch** state of all polygons is reversed. If you do this by accident, use the **Statistics** window to select/re-select just the **SubPatch** surfaces.

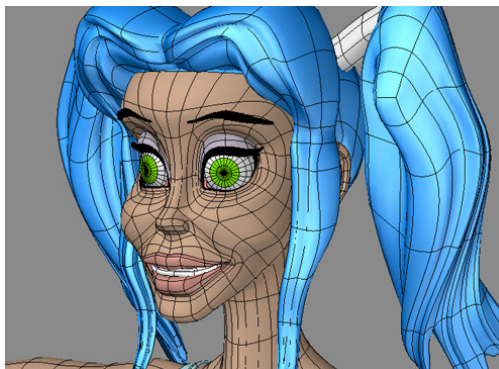
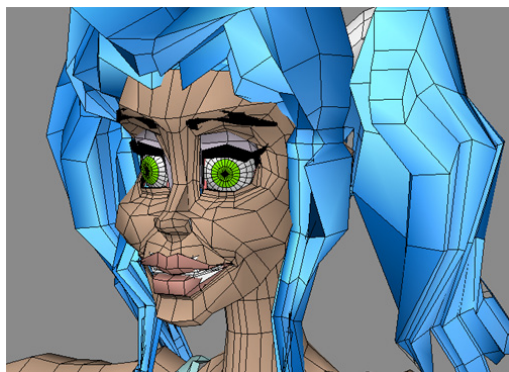


	Name	Num
+ -	Total	4636
+ -	Faces	628
+ -	Curves	0
+ -	SubPatches	3980
+ -	Skelegons	28
+ -	Metaballs	0
+ -	1 Vertex	0
+ -	2 Vertices	28
+ -	3 Vertices	110
+ -	4 Vertices	4498
+ -	>4 Vertices	0
+ -	Non-planar	0
+ -	Surf: (none)	0
+ -	Part: (none)	4608
+ -	Col: (none)	0

To change the shape of the **SubPatch** object, you manipulate the shape of the polygonal cage (i.e., control mesh) using most of the standard Modeling tools, including **Bevel**, **Smooth Shift**, **Metaform**, and so on, or by just dragging points and polygons around. Don't use tools that may create greater than four-point polygons, like **Boolean**.



You can select the **SubPatch** vertices by clicking either on the true (polygon cage) vertex location or the location of the vertex when mapped on the **SubPatch** surface.



When you use **SubPatch**, you often manipulate the polygons into what would normally be non-planar, degenerate, or *cattiwompus* polygons. Since the polygonal cage is merely a reference for the **SubPatch** object, this is perfectly acceptable.



**HINT:** If you end up editing a very complex cage, the display can become confusing. In such a case, the cage is often close in shape to the **SubPatch** surface. Thus, you may be able to edit the normal polygons and just flick the **SubPatch** mode on to check the results.



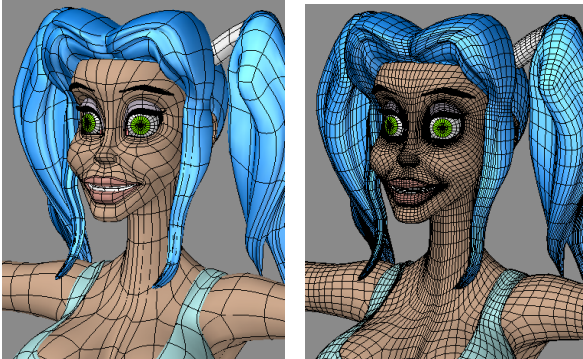
**HINT:** Since it is possible to adjust vertices on a **Subpatch** surface by selecting them directly, deactivating the **Show Cages** and **Show Guides** options on the **Viewports** tab of the **Display Options** panel (**Edit Menu > Display Options**) to hide the polygonal cages and point guide lines can improve the performance of Modeler.



## Freeze Command

(default keyboard shortcut **Ctrl D**)

The **Freeze** command (**Construct > Convert: Freeze**) is used to convert curves, **SubPatch** objects and **Meta-primitives** into polygons.



### To convert a curve into a polygon:

Select the curve(s) and choose **Construct > Freeze**. Each independent curve will become a distinct polygon. An open curve will be closed (between its end points) before it is converted to a polygon.

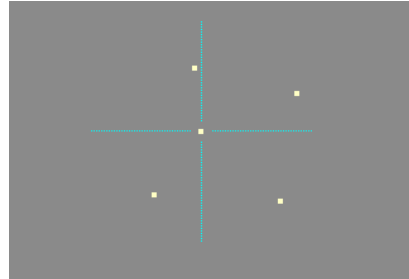
The number of points that Modeler will use to approximate the curve depends on the setting for **Curve Division** on the **General Options** panel (**Modeler > Options > General Options**).

## Meta

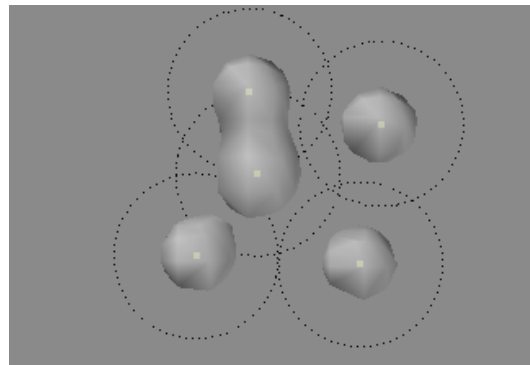
### Convert Metaballs

You can convert *points* into plain **Metaballs** using the **Convert Metaballs** command (**Construct > Convert: Meta > Metaballs**).

**Step 1:** Create a point or multiple points.

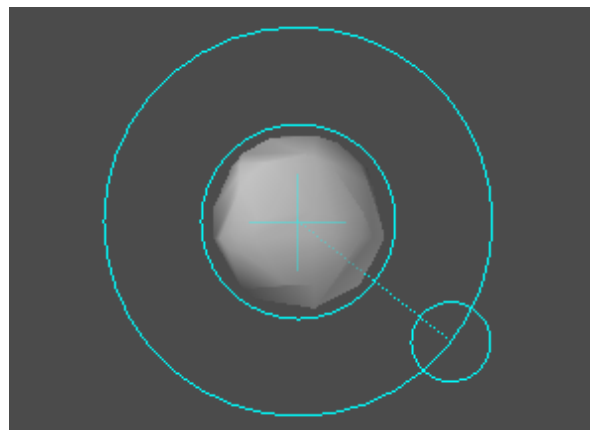
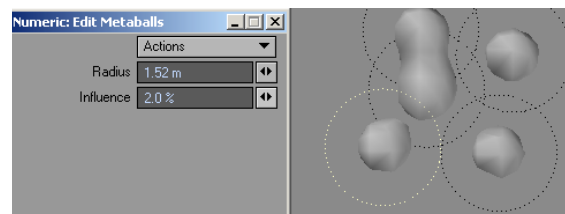


**Step 2:** Select **Construct> Convert: Meta > Convert Metaballs**.



### Edit Metaballs

Select the Metaball you wish to edit and then activate the **Edit Metaballs** tool (**Construct > Convert: Meta > Edit Metaballs**). On the numeric panel, you can adjust the **Radius** and **Influence** with the numeric panel or interactively with the handles in the viewport.





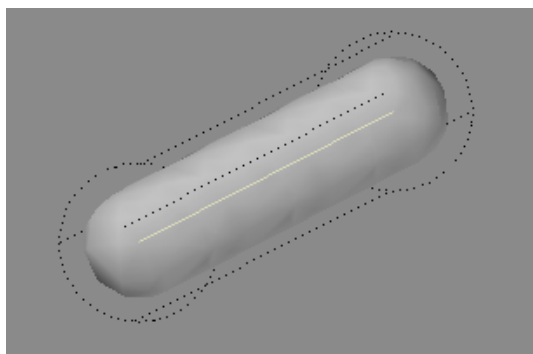
## Convert Metaedges

**Meta edges** can be created from two-point polygons or curves. Simply select the polygon/curve and choose **Construct > Convert: Meta > Convert Metaedges**.

**Step 1:** Select a two-point polygon or curve.



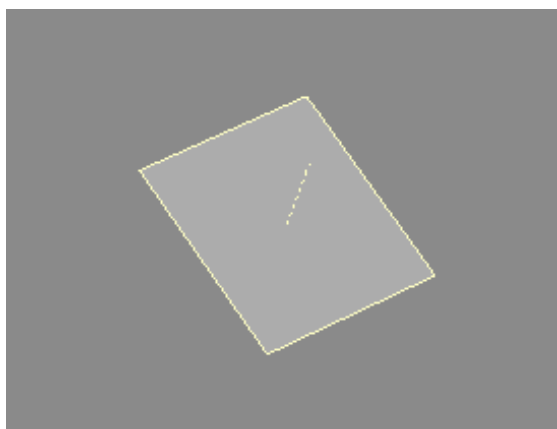
**Step 2:** Use **Construct > Convert: Meta > Convert Meta Edges** to convert the selection to a Meta Edge.



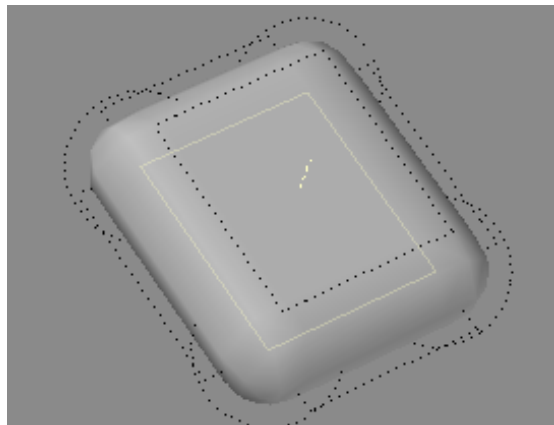
## Convert Metafaces

There is no **Draw Metafaces** tool. They must be created from three- or four-point polygons or curves. Just select the polygon/curve and choose **Construct > Convert: Meta > Convert Metafaces**.

**Step 1:** Select a polygon or curve.

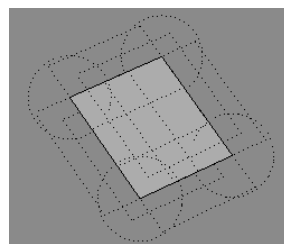
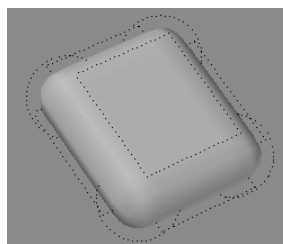


**Step 2:** Choose **Construct > Convert: Meta > Convert Metafaces**.



## Toggle Metamesh

You can also toggle the Meta-primitive mesh on/off by choosing **Construct > Convert: Meta > Toggle Metamesh**. You may want to do this if you have a lot of Meta-primitives and your display refresh is too slow. This is not just a display change and is similar to toggling off **SubPatches**.



Left: On, Right: Off



**NOTE:** If you save a Meta-primitive with the mesh off, you will not be able to see the surface in Layout.



---

## Chapter 7: Detail Tab

---





## Detail Tab

The Detail Tab contains a variety of tools for fine-tuning the points, polygons, edges and curves of your geometry.

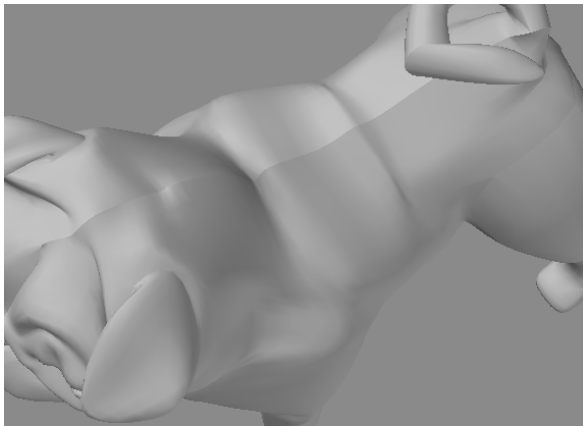
## Points

### Merge Points Command

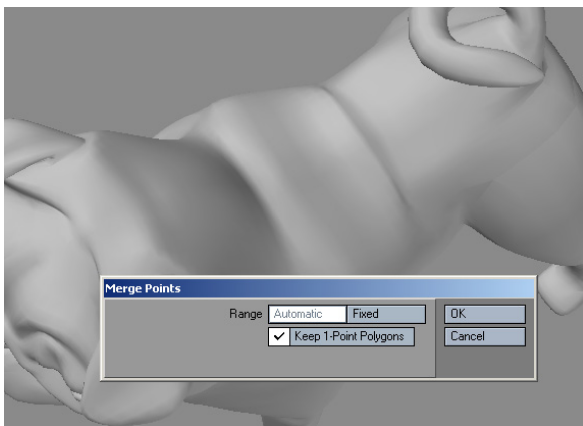
(default keyboard shortcut **M**)

Use the **Merge Points** command (**Detail > Points: Merge Points**) to remove points that occupy the same or very nearly the same space. Remember, if all of the points along a polygon's edge are shared, LightWave's surface smoothing can smooth over the two polygons. Many circumstances exist where you may create unnecessary duplicate points, but here is a list of some common ones:

- After you cut polygons to a separate layer and then paste them back.
- After you use the **Mirror** tool and **Drill** operations, because the complex mathematics involved often generate extra points.



Before Merge



After Merge

If the **Keep 1-Point Polygons** option is disabled, polygons are automatically deleted if they become single points (due to their vertices merging together). The default is for these polygons to be retained.

The **Merge Points** command has two **Range** modes: **Automatic** and **Fixed**. These basically determine the criteria for when points should be considered *mergeable*.

**Automatic** merges the points that share the same space. For objects or portions of objects that you have cut from the main object with the intention of merging later (back into their original positions), use **Automatic**.

**Fixed** merges any points within the specified **Distance** of one another. When merged, one point will move to the location of another — **Merge** will not average the distance between points to create a new point midway between them. As such, the related polygon may be stretched as necessary.



**HINT:** To speed the operation, select the points you wish to merge before using this command. (It's OK if you select more than you need.)



**NOTE:** See also the **Weld** command, described next.





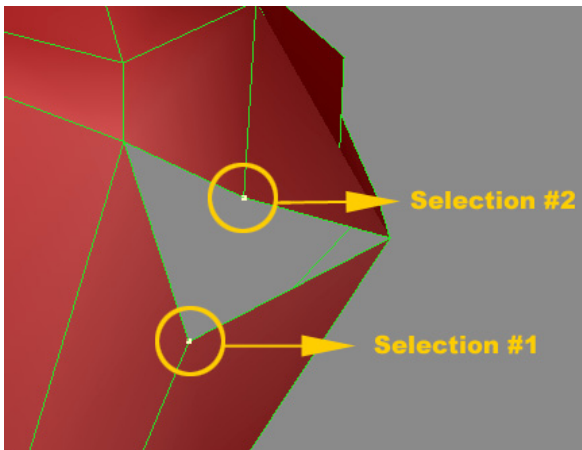
## Weld Command

(default keyboard shortcut **Ctrl** **W**)

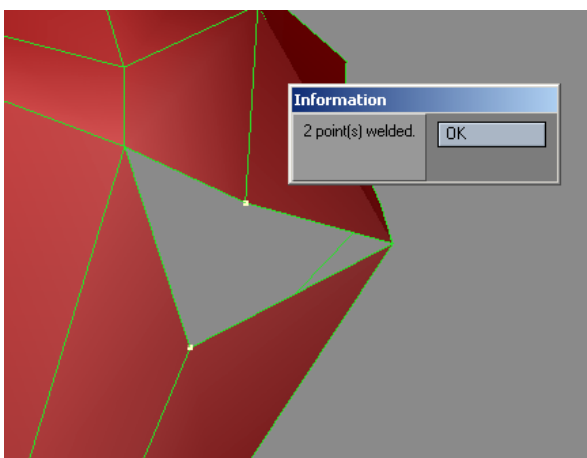
The **Weld** command (**Detail > Points: Weld**) is similar to the **Merge Points** command except that the selected points are welded to the last point selected.

### Steps for Welding:

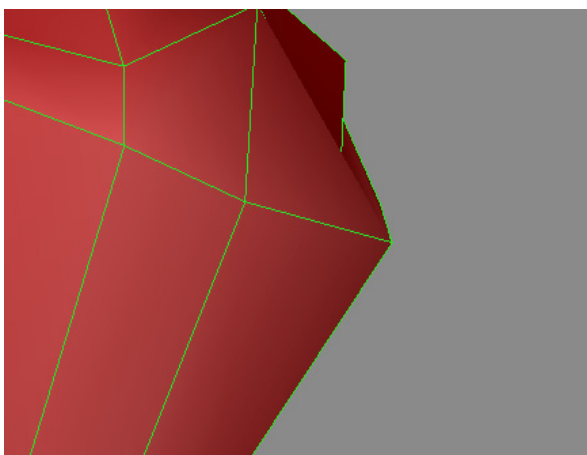
**Step 1:** Select the points that you would like to weld. The last point selected is where the points will be welded to.



**Step 2:** Select **Detail > Points: Weld**. A small dialog box will appear letting you know how many points will be welded.



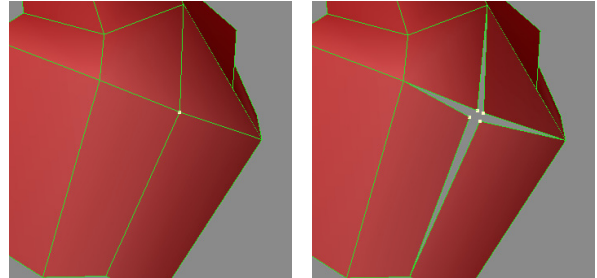
**Step 3:** Click **OK**. Selection #1 will be welded to Selection #2.



## Unweld

(default keyboard shortcut **Ctrl** **U**)

The **Unweld** command (**Detail > Points: Unweld**) lets you create multiple copies of the selected points so that none are shared by two polygons at once—effectively reversing an executed **Weld** command.



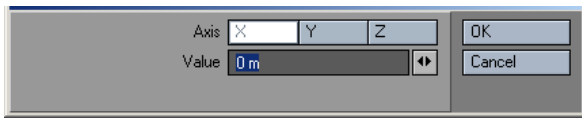
Left: Point selected before Unweld Right: After Unweld and points spread out.



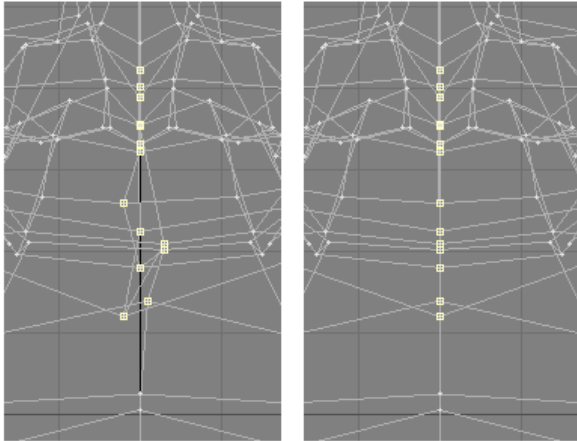
## Set Value

(default keyboard shortcut **V**)

The **Set Value** command (**Detail > Points: Set Value**) sets the X, Y, or Z position values for the selected points or polygons.

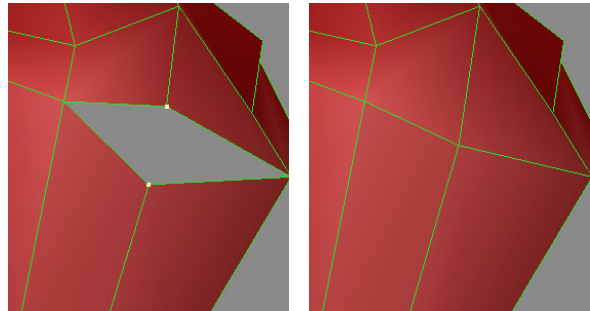


This is a great tool for aligning all of the points along a strict axis. You may find a need for this when you are trying to merge two halves of a mirrored object together and are having trouble merging all points along the mirroring axis.



## Weld Average

The **Weld Points To Average** command (**Detail > Points: Weld Average**) is similar to **Weld**, but averages the distance between points before welding.



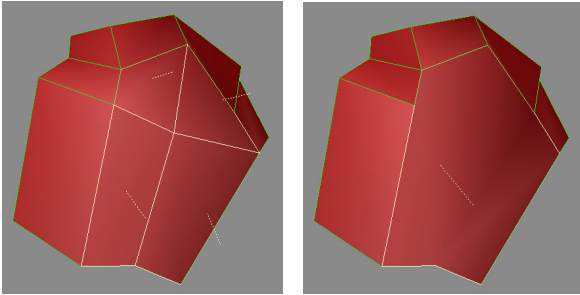


## Polygons

### Merge Polygons Command

(default keyboard shortcut **Shift** **Z**)

**Detail > Polygons: Merge Polygons** joins two selected polygons that share at least one common edge. A common edge is one where two or more polygons share all of the points along that edge.



The **Merge Polygons** command also works on curves, if they share a common terminating point. It doesn't matter if you use the start or end of the curve. They will be merged into one continuous curve.

Common problems are edges that don't share all points. Try merging points first to eliminate overlapping points. After using this command, check for non-planar polygons, which often result.

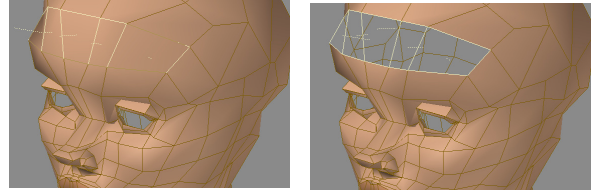
### Flip Polygons

(default keyboard shortcut **F**)

Often as you create polygons or use various tools with single-sided polygons, their surface normals will point in the wrong direction. Blender provides some tools to help you solve that problem.

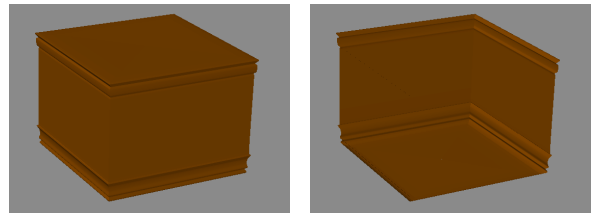
#### To flip polygons:

Simply select the polygon(s) and choose **Detail > Polygons: Flip**. This action reverses the direction of the surface normal for the selected polygons.

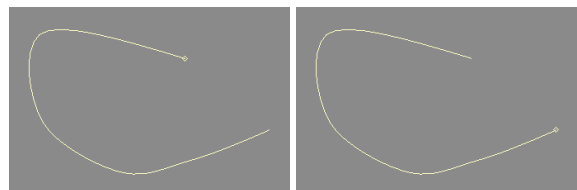


NOTE: You should get in the habit of checking regularly for polygons facing the wrong direction as it is a common problem.

Sometimes you may choose to have the polygons facing the other way (flipped). A room is a good example.



NOTE: This command also reverses the direction of selected curves.

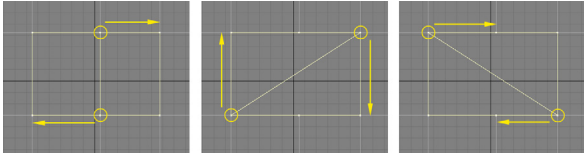




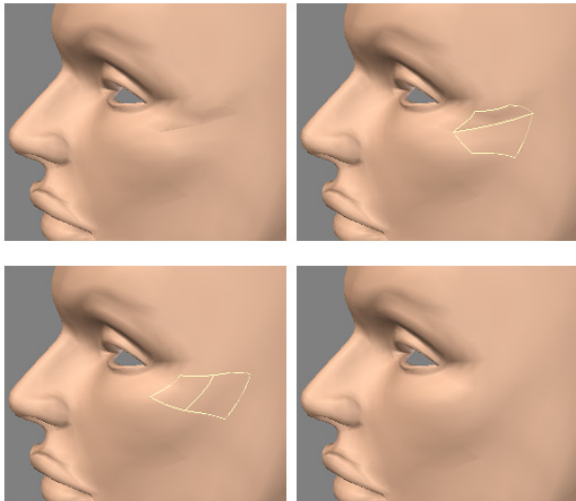
## Spin Quads

(default keyboard shortcut **Ctrl** **K**)

The **Spin Quads** command (**Detail > Polygons: Spin Quads**) can be applied to adjacent four-point polygons (i.e., quads) that share an edge. It merges the two polygons together and then splits them using a different set of opposing points, each time selected in a clockwise direction. If you apply **Spin Quads** three times, you'll be right back where you started.



**Spin Quads** can be quite handy for fixing areas on a **SubPatch** object where a seam just doesn't look right, as in the example below.

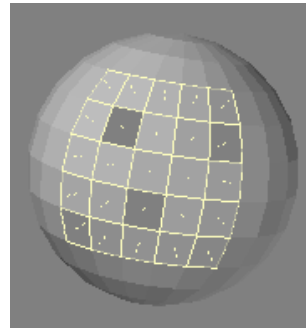


## Aligning Polygons

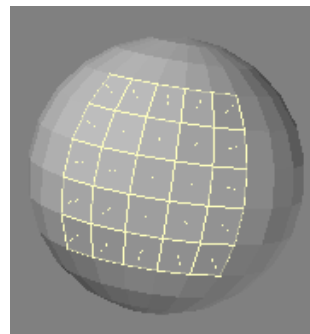
**Align Polygons** (**Detail > Polygons: Align**) attempts to automatically make all polygons face the correct direction. After using, you may still need to use the **Flip Polygons** command (**Detail > Polygons: Flip**) before you are done.

### To align polygons

**Step 1:** Select an area that includes the polygons needing to be flipped, as well as a large number of polygons facing the correct direction. Selecting polygons that share points and edges helps.



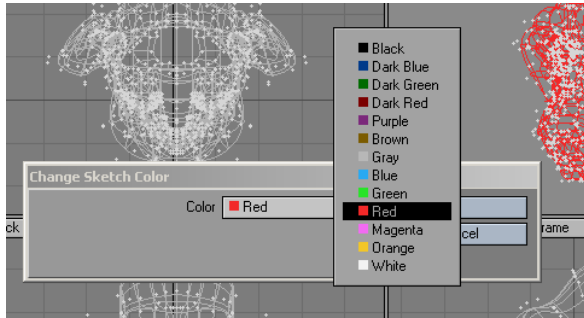
**Step 2:** Choose **Detail > Polygons: Align**. Verify that the polygons were flipped as desired.





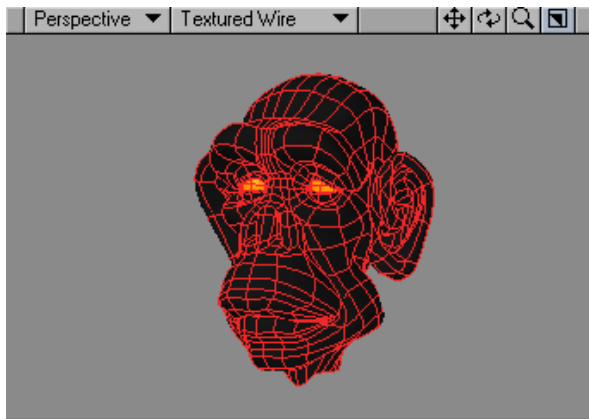
## Sketch Color

When using the various **Rendering Styles** in a viewport, you can change the polygon shading color by choosing **Detail > Polygons: Sketch Color**.

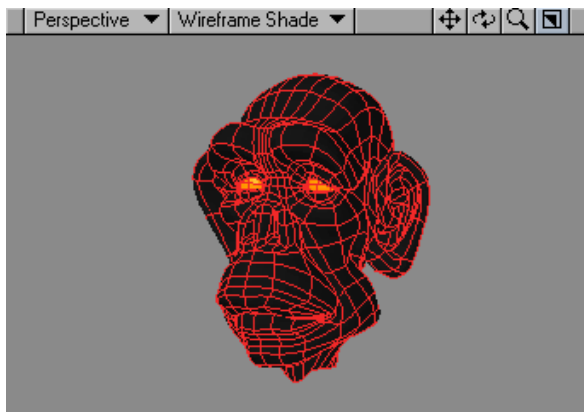


**Sketch Color** will affect the viewport in the following modes:

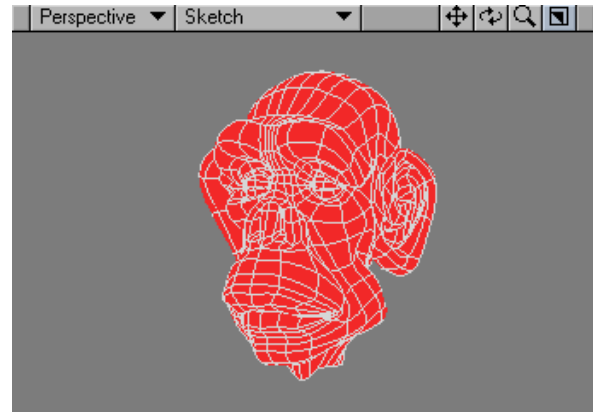
**Textured Wire** — The wireframes will be shown in the **Sketch Color** selected.



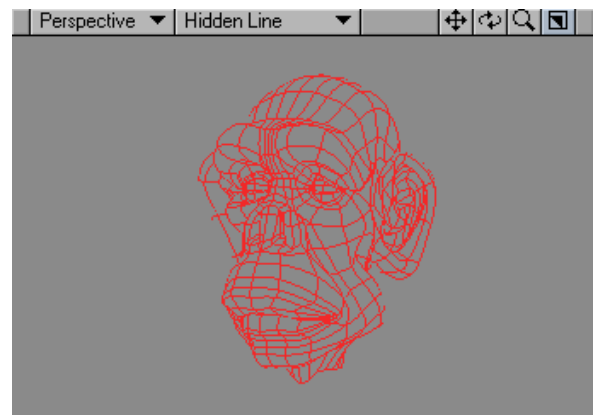
**Wireframe Shade** — The wireframes will be shown in the **Sketch Color** selected.



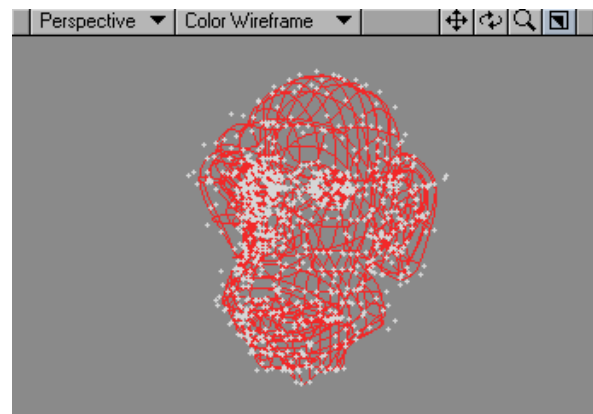
**Sketch** — The polygons will be shown in the **Sketch Color** selected.



**Hidden Line** — The wireframes will be shown in the **Sketch Color** selected.



**Color Wireframe** — The wireframes will be shown in the **Sketch Color** selected.

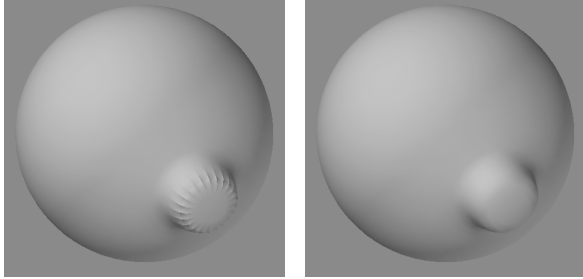


NOTE: The default **Sketch Color** can be changed in the **Display Options** panel.

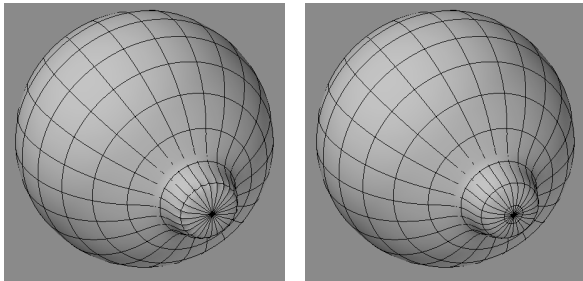


## Fix Poles

**Fix Poles (Detail> Polygons: More > Fix Poles)** is a simple tool which will take the poles you have selected (or all the poles that exist in your geometry if no points are selected) and will modify them to look better in subpatch mode by making cuts to form a tiny flat disc section at each pole, flattening and reducing the smoothing tension caused by the triangles all around it.



Left: Before, Right: After



Left: Before, Right: After

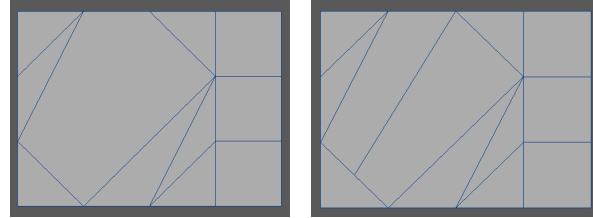
Note that it doesn't perform any miracles and some slight bumps and abnormalities may appear on the surface with curved poles. However, it certainly does make them look much better and more concealed most of the time, and will often obliterate such smoothing artifacts completely on poles which span across broad, flat regions. You can often apply the tool multiple times to the same pole to reduce the tension.



NOTE: Applying it twice on sphere poles looks best. Applying it too many times may cause the flatness of the discs created at the poles to become more pronounced.

## Fix 3-5

**Fix 3-5 (Detail> Polygons: More > Fix 3-5)** is a simple tool to turn each pair of polygons consisting of a triangle adjacent to a pentagon into quads which will generally work better with **Subpatch** mode. The tool works with your poly selection, and will repair any 3-5 polygons it detects in the selection. If no polygons are selected, the tool will analyze all the polygons in your object.

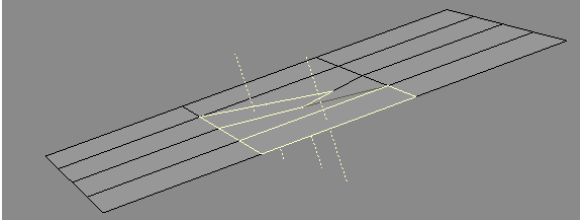


It is recommended that you always select the polygons you want to repair in a dense object, as the algorithm can take some time to search through all the polygons in a very dense model with high poly counts.



## Double Sided

Sometimes you'll actually want double-sided polygons, for example, when you need to go inside a hollow object. Most of the time, you can get away with simply using the **Double-sided** surface attribute on the **Basic Tab** of the **Surface Editor**. However, if you need to actually model the double-sided polygons, it is relatively simple. Just choose **Detail > Polygons: More > Double Sided**.



It is also possible to make double-sided polygons manually using the following procedure:

**Step 1:** Select the target polygons (even if it is the entire object).

**Step 2:** **Copy** the selection.

**Step 3:** **Paste** the selection.

**Step 4:** Choose **Detail > Polygons: Flip** to flip the (still) selected polygons.

**Step 5:** Choose **Detail > Points: Merge Points** to merge points.

## Unify Polygons Command

(default keyboard shortcut **Shift** I)

The **Unify Polygons** command (**Detail > Polygons: More > Unify Polygons**) will convert selected polygons into a single polygon. The polygons must share the same points, such as double-sided polygons. It creates single-sided polygons with regard to the direction of their surface normals. However, unusual object shapes can occasionally fool its complex algorithms, resulting in single-sided polygons facing the incorrect direction. If this happens, you may need to use the **Align Polygons** or **Flip Polygons** commands to correct the resulting direction of the normals.

## Unify Normals

(Detail>Polygons>Unify Normals) When you have normals on the same mesh that are pointing in opposite directions (such as a flipped polygon), Unify Normals will point them in the same direction. Unify Normals will flip all polygons in the same direction as the first polygon that is found or selected.





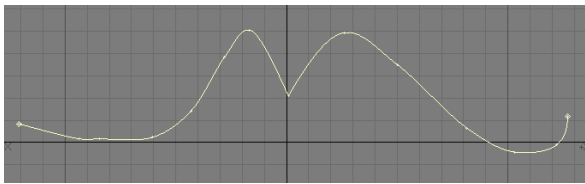
## Curves

### Smooth

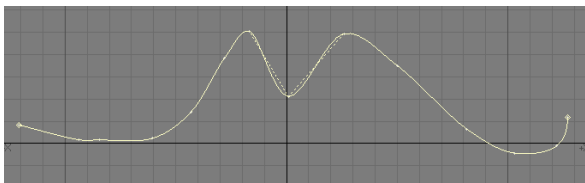
The Smooth command (**Detail > Curves: Smooth**) allows you to smooth two overlapping curves. When two curves share the same end point, because of point merging or some other reason, the curve may not flow smoothly through that point. This can be particularly troubling when designing a complex spline cage. You could kill the curve (k), reselect the points and create a new curve, but there is an easier way.

#### To smooth curves sharing a common end point:

Select the two curves (**Polygon Selection** mode).



Click **Detail > Curves: Smooth**. This smooths the joint as though the two curves were originally plotted as one continuous curve — even though they remain individual curves.



**NOTE:** Make sure that the point they share is in fact one point. (If not, merge the points into one.)

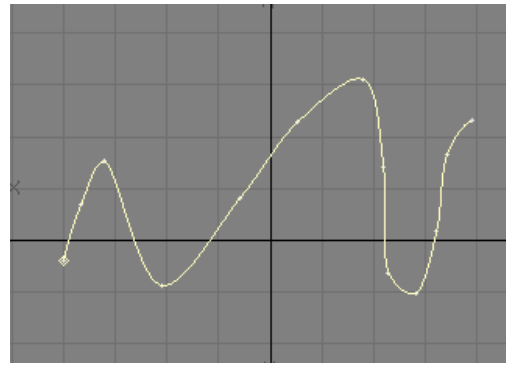
### Control Points

#### Using Control Points

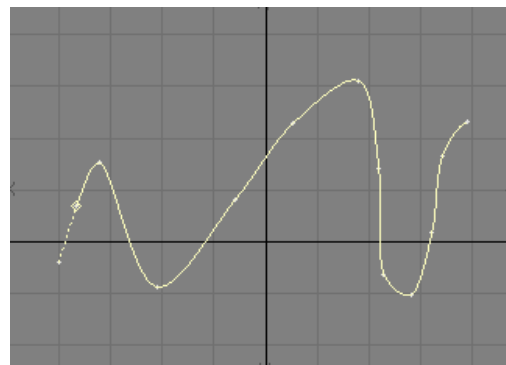
When using a curve as a Modeling tool, sometimes you need to adjust the curve to get it to “that perfect angle.” Because of the way points control an open curve’s shape, it can be difficult to get the desired result at the ends of a curve. Fortunately, Modeler lets you have extra “control” points at each end, which affect the curve shape, but don’t actually become part of the curve’s Modeling characteristics.

#### To activate control points on a curve:

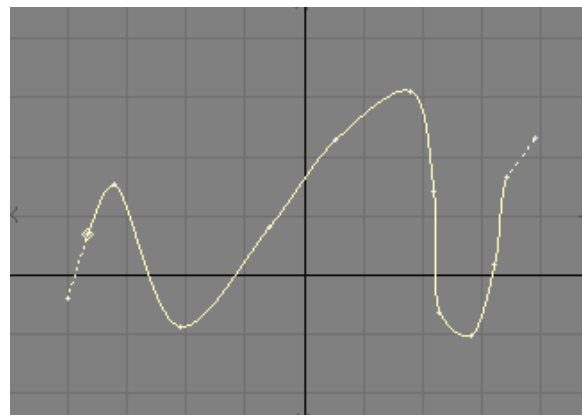
**Step 1:** If you have more than one curve, select your desired curve first using the **Polygon Selection** mode.



**Step 2:** Choose **Detail > Curves: Control Points > Begin Control Point** to detach the first point from the curve. It is your starting control point. Dragging this point will change the shape of the curve at the new starting point.



**Step 3:** Choose **Detail > Control Points > End Control Point** to detach the last point from the curve. It is your ending control point. Dragging it will change the shape of the curve at the new ending point.



The control points commands will also toggle the control point state off, if selected again.



**NOTE:** It is perfectly legal to have only a beginning or ending control point on a curve.



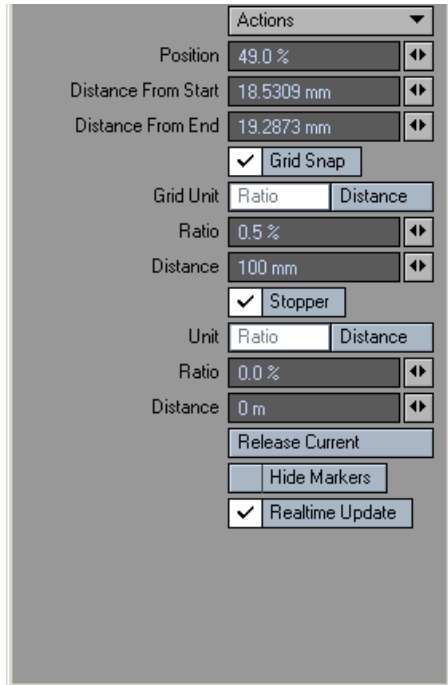
**HINT:** If you plan to use control points on a curve, you should add an extra point at the beginning and end of a curve for that purpose.



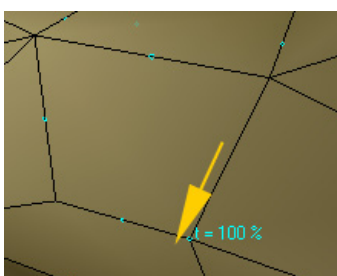
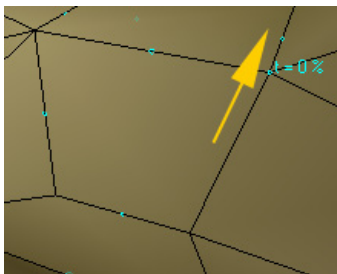
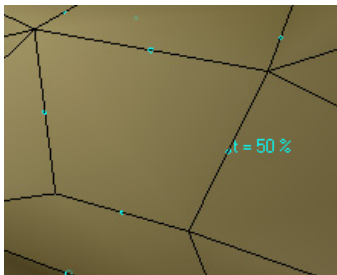
## Edges

### Add Edges

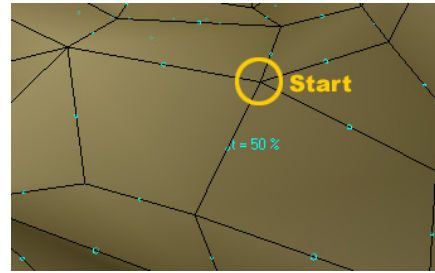
**Add Edges (Detail> Edges: Add Edges)** is part of the **Edge Tools** toolkit that gives you the ability to interactively place new edges on your geometry.



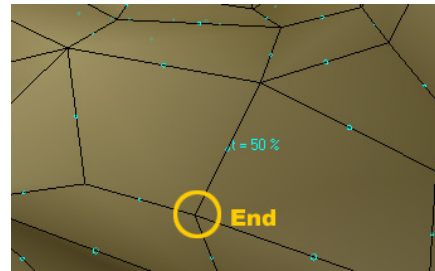
**Position** — Each edge is assigned a marker located at the center of the edge (50%). Use the mini-slider next to the input field or interactively move the position of the marker in the viewport with the **LMB**.



**Distance From Start** — This field will display the distance from the **Start** point of the selected edge to the marker on the edge.



**Distance From End** — This field will display the distance from the **End** point of the selected edge to the marker on the edge.



**Grid Snap** — Checking this will activate **Grid Snap**. Using **Grid Snap** will limit movement of the marker based on the **Ratio** or **Distance** found under **Grid Snap**.

**Grid Unit** — Choose between **Ratio** and **Distance** to control **Grid Snap**.

**Ratio** — **Grid Snap** is set in percentage from 0% to 100%. Limit the movement in x% increments.

**Distance** — **Grid Snap** is set in Units . Limit the movement in n unit increments.

**Stopper** — **Stopper** sets up limits on marker movement.

**Unit** — Choose between **Ratio** and **Distance** to control **Stopper** settings.

**Ratio** — **Marker Limits** are set in percentage from 0% to 100%. Limit the movement in n% increments.

**Distance** — **Marker Limits** are set in Units . Limit the movement in x unit increments.

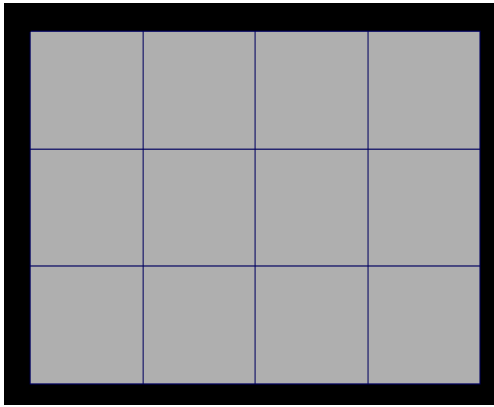
**Release Current** — This option will undo the last edge drawn.

**Hide Markers** — This option will hide the markers' visibility but will still allow edges to be created.

**Realtime Update** — When checked, this option will apply the edge to the geometry. When this option is off, it will not place the edge, but rather will place a preview of the edge. This can be very helpful with heavy geometry on slower machines.

### Steps to Add Edges

**Step 1:** Find geometry to which you would like to add edges.

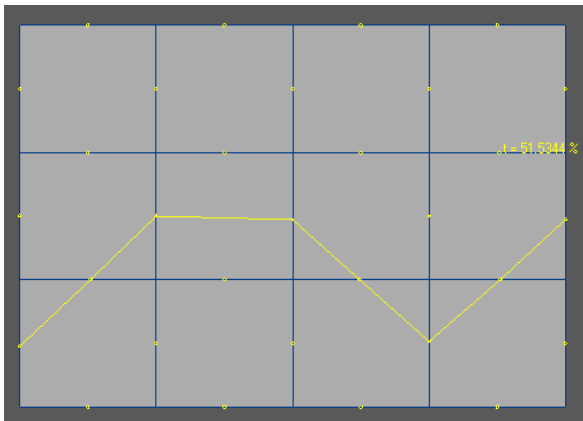
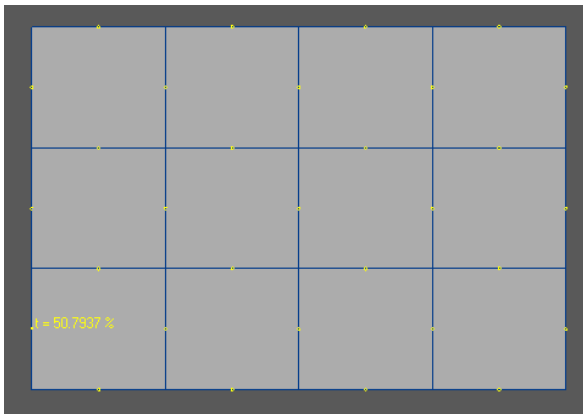


**Step 2:** Select **Add Edges** found under the **Detail Tab**.

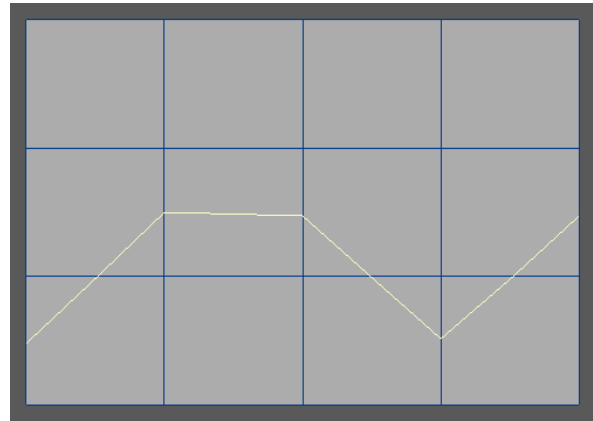


NOTE: Colored Markers should appear on all selected Edges unless **Hide Markers** is selected.

**Step 3:** Left click from marker to marker to draw edges.



**Step 4:** Press **Enter** on the keyboard to apply newly created edges.





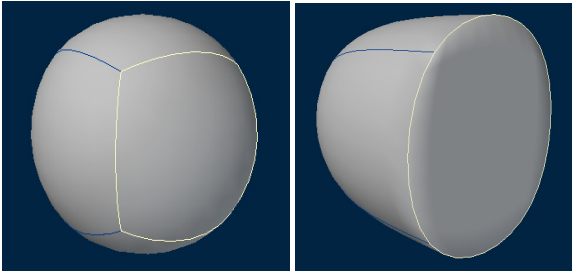
## Edge Weights

### Decrease/Increase Sharpness

Increases or decreases the sharpness of an edge. This allows for harder or softer corners between two polygons.

### Set Sharpness

Sets the sharpness value. For example, a value of 100% will create a sharp edge, while a value of 25% will result in a softer edge.



Value=0%

Value=100%



*Note: The effect of the edge weighting is only visible when in weight shade mode or when CC subdivisions are turned on for the given polygons.*

## Measurement

### Measure

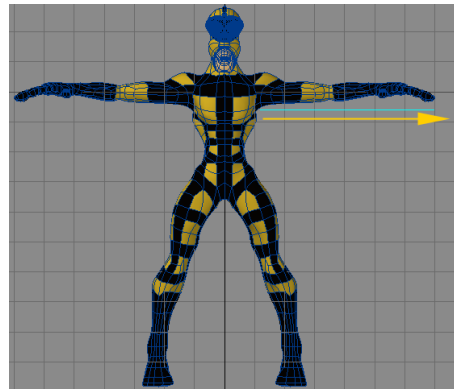
### Angle Tool

You will often need to estimate angles in the modeling process. Modeler provides what is best described as a virtual protractor. With it, you can measure the angle between any two points in any view.

#### To use the Angle Tool:

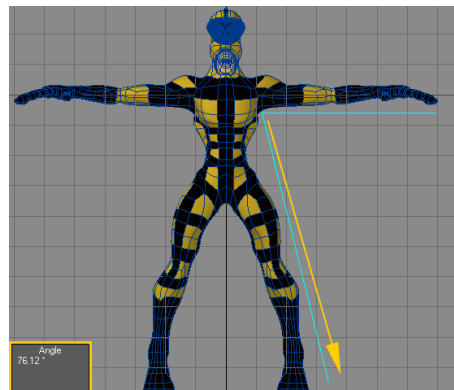
**Step 1:** Choose **Detail >Measure: Measure >Angle Tool**.

**Step 2:** Place your pointer at the vertex of the angle you wish to measure and drag the **LMB** out to form one side of the angle. A line will extend from the drag point.



**Step 3:** Release the mouse button. You can reposition the end of the line by dragging it; however, you cannot move the starting position at this point.

**Step 4:** Place your mouse pointer near the starting point (but not on top of it) and drag out the other side of the angle. The information display will now tell you the angle between the two lines.

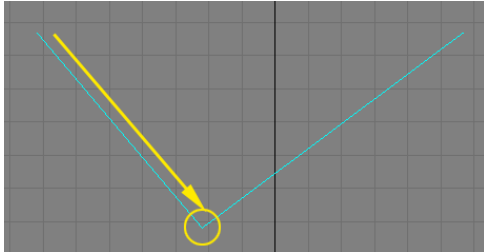




**Step 5:** Once you release the mouse button, you can reposition the vertex or either end by dragging them. To reset, click in a non-active part of the interface.



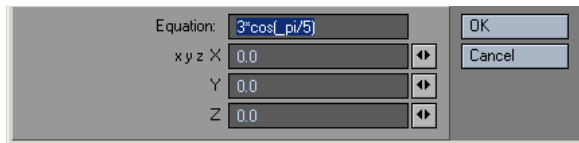
**NOTE:** If you place your mouse pointer on top of the starting point in step 4, the starting point will become the other end to the angle and you will drag out the first vertex instead of leaving it where it should be.



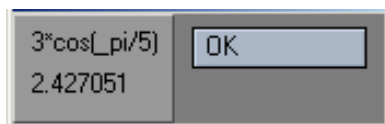
## Calculate

Calculate (**Detail > Measure: Measure > Calculate**) is a handy tool for quick calculation right in Modeler.

**Step 1:** Click **Calculate** from the **Detail Tab**. Enter the **Equation** and the **X,Y,Z** values.



**Step 2:** Press **Enter** on the keyboard to calculate.



## Measure Tool

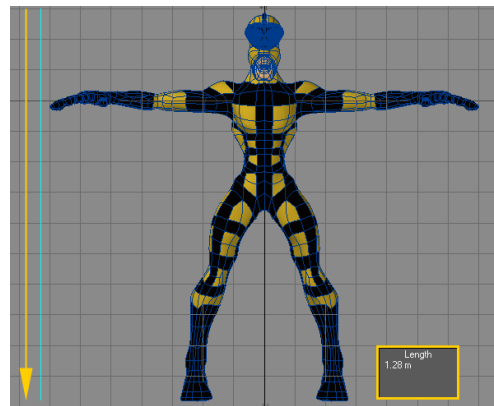
(default keyboard shortcut **Ctrl E**)

You will often need to estimate distances in the Modeling process. The grid size can yield reasonably accurate figures, but if the two points are not aligned somewhat vertically or horizontally, estimating can be more difficult.

Fortunately, Modeler provides what is best described as a virtual tape measure. With it, you can measure the distance between any two points in any view.

### To use the Measure Tool:

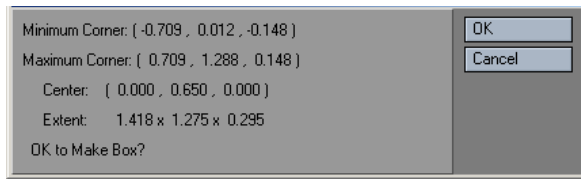
Choose **Detail > Measure: Measure > Measure Tool** and then drag between the two positions you want to measure. The information display in the bottom left corner will indicate the exact distance. You can reposition either end by dragging it. To reset, click in a non-active part of the interface.



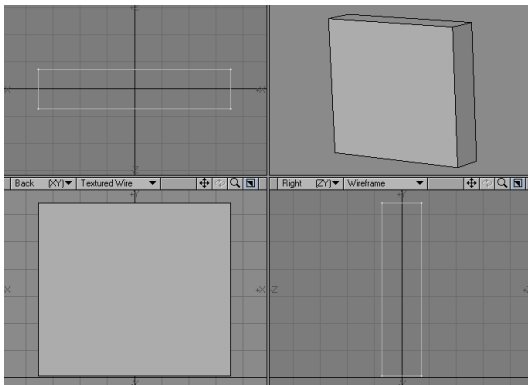
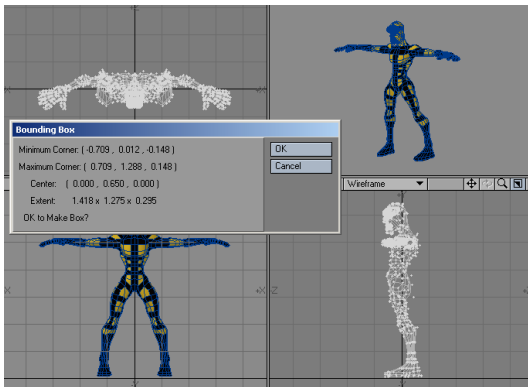


## Bounding Box Command

The **Bounding Box** command (**Detail > Measure: Measure > Bounding Box**) displays size and center position information for foreground layers.

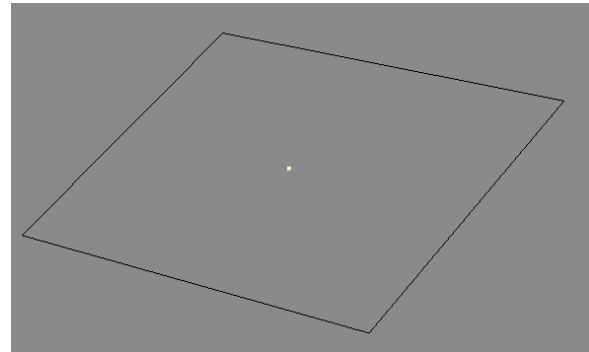


Clicking **OK** will replace the layer(s) with a corresponding bounding box. Use this command to create stand-ins for complex objects, which can be replaced for the final rendered animation.



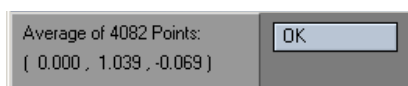
## Find Center

The **Find Center** command (**Detail > Measure: Measure > Find Center**) works very much like the **Point Center** command but instead of displaying a requester with the center information, it will place a point in the center of a single polygon.



## Point Center

The **Point Center** command (**Detail > Measure: Measure > Point Center**) will display a requester that gives you the **XYZ** coordinates of the center of the object in the foreground.



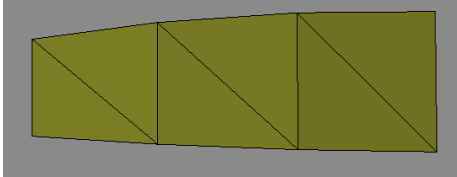


## Tri Strip ACTC

The **Tri Strip ACTC** command (**Detail> Tri Strip ACTC**) will count the number of **Tri-Strips** in an object and will create a **Vertex Color Map**. **ACTC** makes triangle strips and fans out of independent indexed triangles. These triangle strips and fans can improve graphics card **geometry** performance approaching 3x, or a 200% improvement.

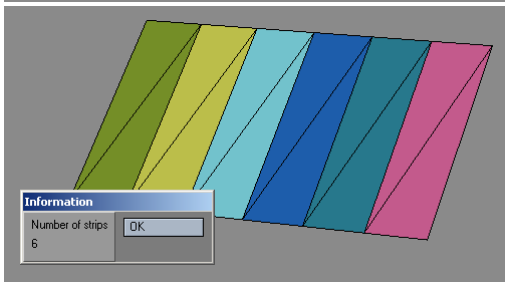
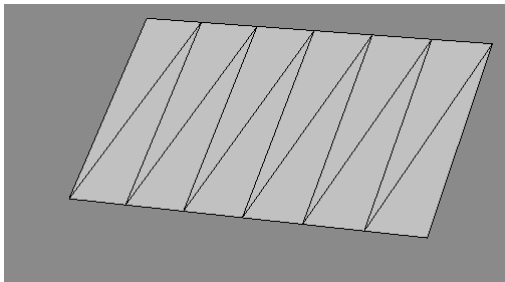
**ACTC** is an abbreviation for Applied Conjecture Triangle Consolidator, more information about this method can be found at: <http://plunk.org/~grantham/public/actc/index.html>.

### Tri-Strip Example:



Tri-strips are a common method to compress mesh data in a lossless way. They define an order of consecutive triangles, with each pair of adjacent triangles sharing a common edge.

Steps — The Strip count will appear in a requester that will pop up when the command is applied. Clicking **OK** will add the **Vertex Color Map**.



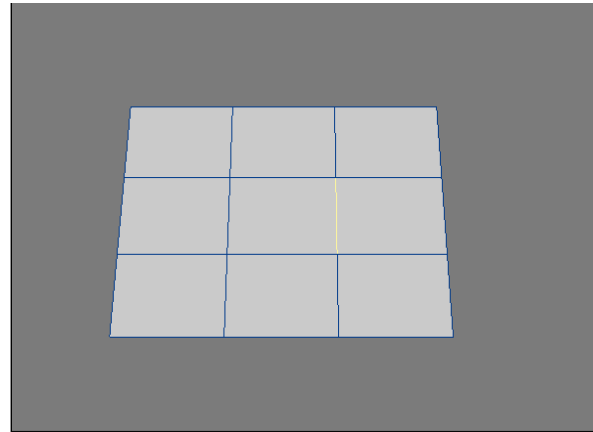
NOTE: Your object must consist of 3-point polygons in order to perform this operation.

## Absolute Measure

Calculates the distance between two points and will tell you exactly how far apart they are.

## Spin Edge

When an edge is selected, this tool will spin the edge and connect each end of the edge to the next adjacent point.



## New Selection Options

**Sel Entire Surface:** Selects all the polygons making up the surfaces of the polygons that are currently selected.

**Sel Entire Part:** Selects all the polygons making up the parts of the polygons that are currently selected.

**Select Path:** Selects a shortest path (there can be multiple) between two elements. The selected elements can be of any type (vertex, edge, polygon).

**Select Outline Points:** Selects the outer points of a selected set of polygons.

**Select Outline Edges:** Selects the outer edges of a selected set of polygons.

### Edge Selection

**Loop Expand:** Expands an edge selection along an edge loop.

**Loop Contract:** Shrinks an edge selection along an edge loop.





---

## Chapter 8: Map Tab

---



## Map Tab

### Introduction

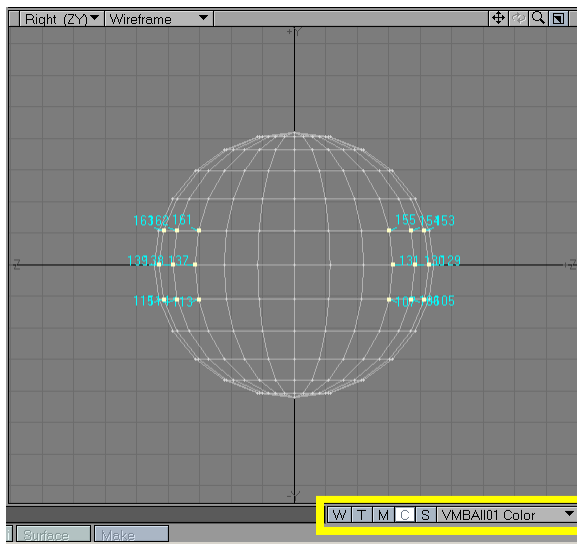
#### What are Vertex Maps?

As you might expect, every point in an object has independent position information. Points even have rotational information, although you normally don't rotate individual points. A *vertex map* (VMap) is additional data that a user can add onto each point in an object. Any point in an object can have a unique entry (value) for a particular VMap or no entry at all. The VMap data is all stored in the object file.

Basically, VMaps are to points, what surfaces are to polygons. Like surfaces, you name your VMaps, which allows you to access them to perform certain operations. Some VMap types are designed for special built-in functions. For example, UV Maps hold texture placement information. Endomorphs hold offset information for point position. Weight maps are used more generally, like for bone influence; however, the SubPatch weight map is used specifically for control-point tension.

If a point has no entry for a particular VMap, the VMap is not assigned to that point — this map will have *holes* in it. Note that an assigned value of 0 is not the same as not having the VMap assigned to that point. However, some functions that use VMaps consider these two states to be equal — usually by necessity — essentially filling the holes with zeros.

VMaps are normally created using the buttons in the lower-right corner of Modeler.

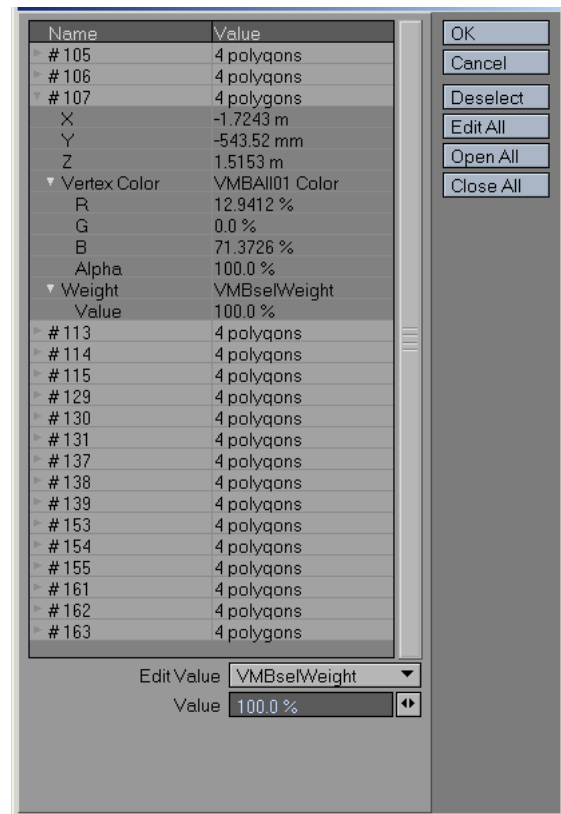


Once the map is created, you can set or edit the values in the map. Many LightWave commands, however, also have the ability to create VMaps.

VMaps are extra data and will increase your object's file size. Thus, depending on the situation, you may be able to minimise the file size by judiciously assigning VMaps to points.

Because you can use vertex maps (VMaps) throughout LightWave, you should have a solid understanding of how they work. Here are some important basic points:

- *Weight* maps have a general range of -100% to +100%.
- Use the Selection Info command (**Info** button on the bottom taskbar, or press I) to determine what VMaps and values are assigned to individual points.



- Internally, there is a single list of VMaps, which includes the maps from all the objects that are loaded. However, editing a map in one object will not affect other objects (using the same VMap name) because the data in the map is attached to the geometry of that object. There is no *cross-talk* between objects, except in the case of renaming. If you rename a VMap, all VMaps (for loaded objects) with the same name are renamed.

- Generally, the only VMaps listed in the main VMap pop-up menu are those that contain actual data in the current object. Other VMaps are filtered out. Not all VMap pop-up menus will do this filtering, however, particularly those on plugins.

New VMaps are an exception to the above standard. Initially, they will appear in pop-up menus even if they contain no data. However, if you deselect a new VMap before using it, it will disappear from current VMap listings. It might save you some typing time, so Modeler still places the name in the pop-up menu that appears beside the naming field when you are creating a new VMap.

- There can be one selected VMap of each general category (weight, UV texture, etc.), and they can be selected in different ways. The pop-up menu in the bottom-right corner of Modeler shows the current selection for the weight, UV texture, and morph categories, and can be used to change them. The VMap List window, discussed later, also shows the selected map in each category. Note that all new maps start out selected.



## What are UV Maps?

Sometimes, when mapping textures onto objects, you will find that the normal projection mapping just doesn't work. This usually happens when the object is organic, or irregular in shape. For those occasions, there's **UV Mapping**!

**UV Mapping** adds two extra coordinates to the points in your object; those on the U and V axis, running horizontally and vertically through a flat plane on which you can paint your texture. Since the coordinates are assigned to points, it's essentially as if that painted texture was fixed to the surface of the object, with pins where all the points are. No matter how irregular your object is, or how it moves or flexes, those pins stay in place, and the texture stays right where you put it. Technically, it's not as accurate as projection mapping, because the texture is really only exact at those points, and merely interpolated everywhere else, but artistically, it works.

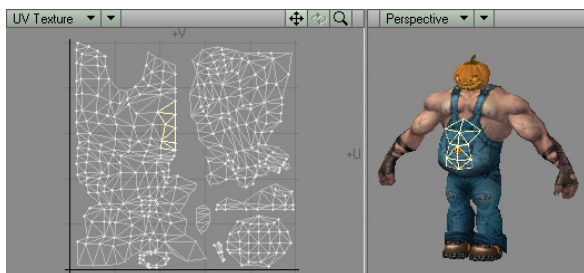
The trick is to make the map first. LightWave has many tools to help you do that.

### UV Texture Maps

Sometimes, standard image mapping tools (i.e., planar, cylindrical, and spherical mapping) may be somewhat limiting where the surface is irregular in shape. These techniques usually work well only where you can globally map the entire texture image using a linear interpolation along two axes. The object geometry essentially has no influence on how the texture is applied.

However, what if you could assign areas of a texture image to points on the surface, essentially *tacking* it down at key points? Well, you can with *UV mapping* in Modeler. Between the tacks, the image is stretched smoothly.

The U and V refer to texture map coordinates and are really not much different than the XYZ coordinates you are familiar with. In fact, UV mapping is the process of setting up a relationship between the two dimensions of an image, U and V, with the three dimensions of an object surface, XYZ.



UV Map on a Serious Sam Character. Note the selected polygons in both views.

Once this relationship is set up, changing any parameter (i.e., U, V, X, Y, or Z) will also relatively change the appearance of the texture mapping. With UV mapping, the object provides additional information for texture mapping, which can be different for any given point on the surface. The texture is more or less stuck to points on the surface using a relationship that you define.

## UVs and Projection

UVs have to come from somewhere. For existing polygonal models, the choices are limited to setting the UV coordinates for each point in the object manually, or applying some *projection*, which automatically generates the 2D texture coordinates from the given 3D point positions. In LightWave, you can create UVs by using projections, which also happen to be the same as the standard projections for texture mapping (i.e., planar, cylindrical, and spherical).

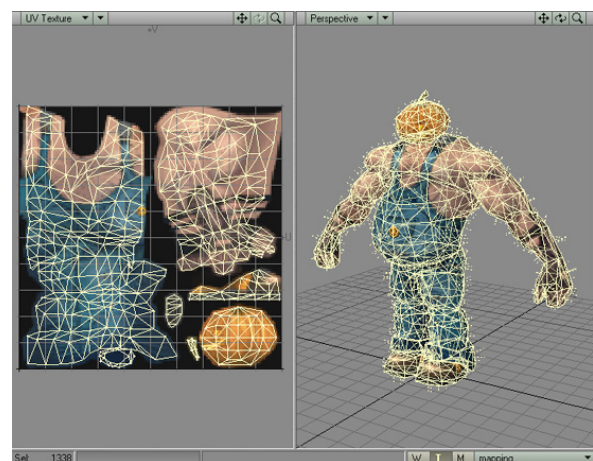
Usually, the projection for the UV map is not perfect for the model everywhere. The projected UV map must be tweaked — eyes and nostrils moved over the right parts of a face, or texture features matched to geometry features.



**NOTE:** Keep in mind that standard projection mapping is more accurate because it has some exact, continuous value over the entire surface.

UV mapping, on the other hand, is technically accurate only at small sample points. The surface is interpolated for the large areas in between the sample points. Adjusting the sample points so that the interpolated areas look right is difficult and the reason why UVs are more difficult to use.

For illustration purposes, let's say you had your texture image printed on a piece of very flexible rubber and wanted to fit it on a toy car made of wood. You could conform the rubber material to contours of the car by tacking it down with thumbtacks. That is more or less what UV mapping does. However, it is a little bit reversed: what you do is *tack* the UV points down onto the image.



### Discontinuous UVs

VMaps now support *discontinuous* values across polygon boundaries, which are useful for UV texture coordinates, gradient weights, and other VMap-controlled surfacing parameters. *Discontinuous UVs* are now automatically used when appropriate with Modeler's Modeling tools.



**WARNING:** Discontinuous UVs are now created in the normal course of (Modeler) business, thus a firm understanding of them is imperative.



## The old seam problem

Prior to LightWave 6.5, when you used any of the automatic UV creation options on continuous geometry (like a sphere or cylinder), you would often encounter weird *seam polygons* where most of the UV texture would wrap backwards along a seam in the geometry. The texture on this polygon would be mapped from something like 0.93 to 0.07 and the entire image would get squished backwards onto this polygon.

## Unweld Command

The **Unweld** command (**Detail > Points: Unweld**) creates multiple copies of the selected points so that none are shared by two polygons. Each polygon is given its own copy of the selected vertices, and VMap values for the polygon are made continuous over the new vertices. This tool is the key to being able to edit discontinuous UVs.

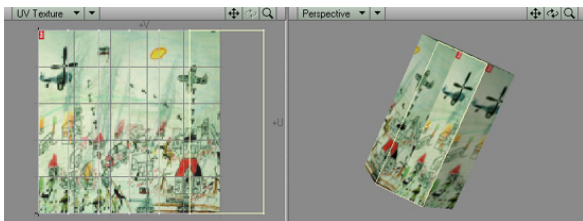
## Merge and Weld Commands

When points are merged with the Merge Points (**Construct > Reduce: MergePoints**) or Weld Points (**Detail > Points: Weld**) command, any introduced discrepancies between VMap values are resolved with discontinuous UVs.

Merge points will no longer merge points that are at different locations in any of their morphs. This prevents distortions of the topology or discontinuities in the morph VMaps. It also provides a method to force seams in the mesh.

## UVs Outside the Box

You may have noticed that UV coordinates can appear outside of the *UVbox*. That is, have U or V values less than 0 or greater than 1. The texture is essentially *tiled*, so these points are still attached to some part of the texture. It is sort of like the old Asteroids video game where the ship moves off the edge, but appears on the opposite side.



It may help you understand the result by thinking of two rotation keys in Layout of, say, 0 and 720 degrees. From the viewer's perspective, the orientations *at those keyframes* are the same; however, there is a (motion) *path* between the keys that LightWave interpolates. For UVs, there is a texture *path*. LightWave interpolates the texture between the UV coordinates.

Let's say you had U coordinates at .9 and 1.3. Now although 1.3 and .3 would be at the exact same position on the texture, there would be a big difference in the interpolation path of .9 to 1.3 and .9 to .3. The latter would be backwards.



NOTE: The Inline Help for LightWave includes a set of tutorials on creating and using UV Maps.

## What are Per-polygon Maps?

**Per-polygon UV Mapping**, or **Poly Mapping**, was introduced in LightWave 6.0b as a way to handle discontinuous **UVs**. (Without some kind of handling, they would cause the entire image map to appear backwards and squashed onto a single polygon at the map seam. Not a pretty sight.) They were superseded by the current **UV Mapping** tools found starting in LightWave 6.5, so unless you are working with a model that was mapped before that, there's really no reason to use them; the new tools are much more elegant and flexible.

Unlike **UV Maps**, **Poly Maps** cannot be edited with the normal **Modify** tools used in the rest of Modeler, because they are not continuous. Instead, they are locked snapshots of some **UV Mapping** of a polygon. So any operation on a vertex must also specify which polygon's view of that vertex is being edited.

You can select polygons that use per-poly UVs by choosing the **Select by Polymap** command (**View > Selection: Maps > Select by Polygon Map**).



## General

### Set Map Value

Set Map Value (**Map > General: Set Map Value**)



Clicking on this command opens a panel that allows you to directly specify a value for any of the **Vertex Maps** used in LightWave.

This value will be applied to all selected points. If no points are selected, this value will be applied to the entire map.

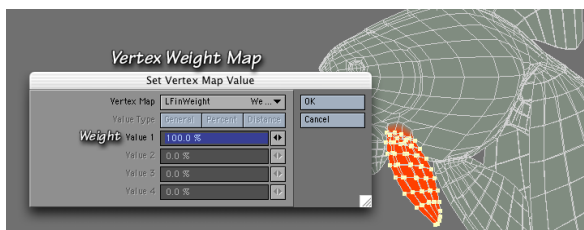
### Vertex Map

The **Vertex Map** pop up menu allows you to choose the map you want to work with. The map selected will be the only map affected, regardless of what map is actually showing in your viewports.



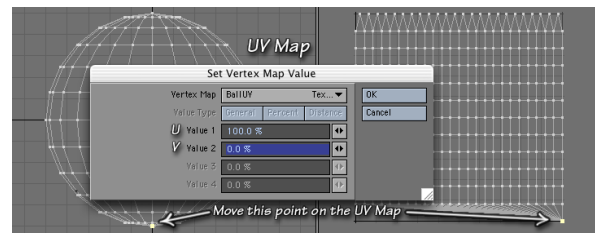
**NOTE:** It's entirely possible to assign values to maps you cannot see. In other words, if you don't see anything happening, check to make sure you are looking at the correct map.

Below that are various fields, which will be available or dimmed, depending on the map type you've chosen.



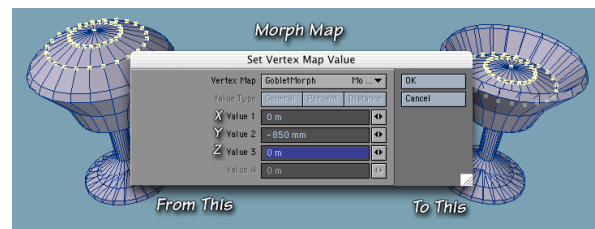
### Weight Map

If you select a **Weight Map**, only one value will be available. Whatever value you type into the field will be applied to all selected points in the map, replacing any values you might already have assigned.



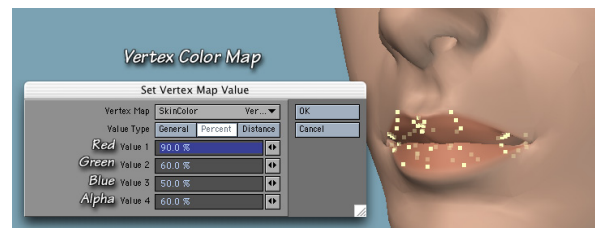
### Texture Map

If you select a **Texture Map**, two values will be available. The first is the **U** value, and the second is the **V** value. 0% for both values is the lower left corner, where the vertices meet on the **UV Map Texture** view. 100%, of course, is the upper right corner. All selected points will snap to the value chosen.



### Morph Map

If you select a **Morph Map**, three values will be available. They correspond to the three axes (X,Y, and Z.) All selected points will be moved along a delta described by the numbers you enter. In other words, if you enter .3 in the middle (Y) field, all selected points will be moved 300 mm upwards from their **Base** values.







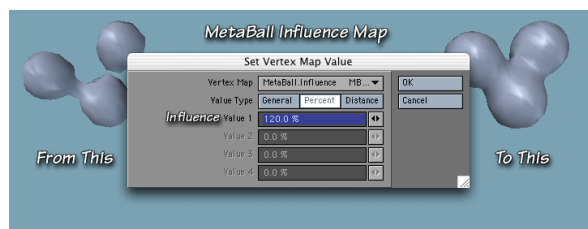
## Color Map

If you select a **Color Map**, four values and three ranges will be available. The values are for **Red**, **Green**, **Blue**, and **Alpha**. The **Ranges** are **General**, **Percent** and **Distance**. It's recommended that you don't use the **General** field. The **Percent** is percent, of course. If you use **Distance** then 1m = 100%.

Don't forget that if you don't set the **Alpha**, in the fourth field, you won't see the color on your map. (For full color, set it to 100%.) Using this tool with no points selected is a quick and easy way to make sure that all the points are in your **Color Map**. (Points with no assigned value are not the same as points with an assigned value of zero.)



**HINT:** Use this tool to assign all the points in your object to a **Vertex Color Map** if you are so inclined; but don't use it to assign the colors. For that, it is much better to select the points, and then use the **Point Color** tool. It will give you a regular LightWave color requester. (But it will only work on selected points.)



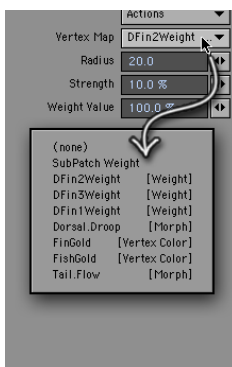
## Metaball

If you select a **Metaball Map**, one Value and three ranges will be available. The ranges are **General**, **Percent**, and **Distance**. They change the size of the **Metaballs**, and the distance at which they influence each other. Once again, any selected points, that is, **Metaballs**, will react to the numbers you enter in these fields.

This tool is not interactive; you won't see your changes until you close the panel. All the fields in all categories revert to zero when the panel is closed.

## Airbrush

The **Airbrush** tool allows you to "paint" directly on your **Vertex Maps**. How it works depends on which kind of map you're working with.

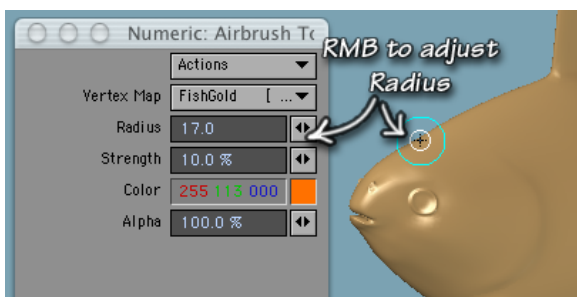


Choose the desired map from the **Vertex Map** drop down in the **Numeric** panel. All the maps available to the **Airbrush** will be listed here by name, with the type of map in brackets on the right.

Whichever map you pick will be shown in any viewports capable of showing it, assuming it's the same type of map chosen in the **Vmap Bar**. (ie. **Weight Maps** will be shown in viewports using the **Weight Map Render Mode**, **Color Maps** in **Smooth Shade**, **Texture**, and **Textured Wire**, etc.)

If you pick a different **Weight** or **Color Map** from the **Vmap Bar** at the bottom of the screen while the **Airbrush** is open, that map will replace the one that's showing, both in the viewports and the **Airbrush Numeric panel**, if it's the same type.

The "same type" caveat is important. If you have a map of a different type showing in the **Numeric panel** and the **Vmap Bar**, for instance, a **Weight Map** in the **Vmap Bar**, and a **Morph Map** in the **Numeric panel**, any airbrush strokes will be applied to the map in the panel, in this case the **Morph Map**, and not to the one in the **Vmap Bar**, even though that's the one you will be looking at. So, if the airbrush seems to have no effect, check to make sure you are working on the right map.





## Radius

The **Radius** determines the relative **Falloff** for the tool. You can adjust it interactively with the **RMB**, as you can with the **Falloff** values of most tools. In order to change anything, the radius must touch a point on your map; bear that in mind as you adjust it.



## Strength

The **Strength** determines the intensity of the effect. Small values are recommended; start with 10% and increase it if you need to.

This tool behaves like a real airbrush in some ways. If you move quickly, you'll get less coverage than you would with a slow stroke, and going back over areas will increase coverage. However, unlike its real-world namesake, the tool must be moving in order to do anything. Don't expect to get full coverage by holding it still over a point.

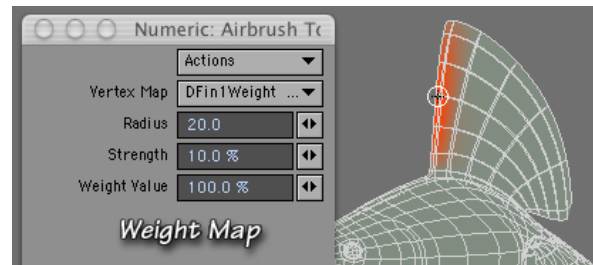
Also, remember that you are painting vertices, not polys. Running the **Airbrush** up and down the polys will have no effect unless the **Airbrush** radius touches a point. For best results, ignore the polys, and paint the points.



You can select points or polys to limit the effect to the selected vertices only; however, because of the nature of **Vertex Maps**, you'll find that the effect is blended with adjacent points or polys. Each point can only have one value; so as you can imagine, the more points you have, the more detailed your painting can be.



If you want a sharp edge between two polys, cut them, and paste them back in. Then either select the polys you wish to work with, or hide the others. In either case, since they are no longer sharing a point, there will be no blending. When you are finished, merge the points. They will keep their sharp edges.



## Weight Value

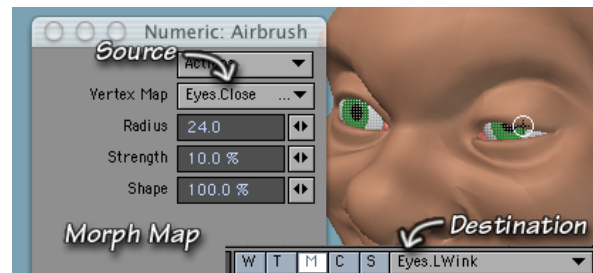
To use the **Airbrush** with a **Weight Map**, it's best to change one of the viewports to **Weight Shade**. (You don't have to, unless you want to see what you're doing; the tool will change the **Weight** value anyway. So be warned.) Pick the **Weight Map** you want to work with from the list; it will automatically be shown in the viewports.

Type the value you want to apply in the **Weight Value** field. This value will replace the current one. In other words, if you are painting with 10% weight, you don't have to worry about any of the vertices exceeding that value. You can also paint with 0%, to eliminate positive weight without worrying about negative weight.

If you hold down the **Shift** key, you'll paint with the inverse value. In other words, if you're painting with 30%, holding down the **Shift** key will enable you to paint with -30%. If you're using -50%, it will give you +50%.

If you hold down the **Ctrl** key, the weight will go towards zero, whether it's currently negative or positive. (In other words, it will go up if it's negative, or down if it's positive.)

You can push the values far beyond 100% and -100%, if you are so inclined.







## Airbrush and Morph Maps

To use the **Airbrush** with a **Morph Map**, select the morph you want to paint on from the **Vmap Bar** at the bottom of the screen. Select the morph you want to paint from in the **Vertex Map** menu in the **Airbrush Numeric** panel.

### Shape

Choose the percentage of the **Shape** that you want to apply from the **Shape** text field. You can change the current morph half way to another one, or change parts all the way, or just take 10% off the top. It's very flexible.

You can also paint a morph onto the **Base**; but if you do, it will change all the relative morphs in your object. You are given a warning if you attempt it.



## Airbrush and Color Maps

To use the **Airbrush** tool with a **Color Map**, choose the map you want to color from the drop down list. Make sure that it's being used on a surface, and is visible. (Check the **Advanced Tab** of the **Surface Editor** for use.) The **Airbrush** will paint on maps that are not showing, so be careful. If you don't see any effect, don't assume that your settings are too weak.

Choose a color from the normal **Color Requester**. Be aware, though, that LightWave will apply the color over the base color of your object using the **Multiply Blending Mode**. If you are familiar with Photoshop, it's very similar. Adding color will always make the base color darker, never lighter. (If that's a problem, simply start with a white base color, of course.) The colors you paint on the map itself will replace whatever colors are there, so it's easy to lighten them.

### Alpha

If you are using an **Alpha Channel** in your map, (enabled when you created it,) then you can also adjust the **Alpha** with the **Airbrush**. Do this by either changing the number in the **Alpha** text field, or by holding down the **Ctrl** key as you paint. You can increase the **Alpha** by using the **Alpha** text field; but it will also add color. (Which is just as well, really, since uncolored points with 100% **Alpha** are black.)

## Clear Map (default keyboard shortcut **\_**)

To clear a **Vertex Map**, select the map you want to clear from the **Vmap Bar**, and click on the command. All values will be cleared, no matter which kind of map you chose. (**Weight**, **Texture (UV)**, **Endomorphs**, **Color Vertex Maps**, or **Selection Sets**.)

To clear an entire map, make sure no points or polys are selected before you click.

This differs from the **Delete Map** command because it leaves the map. Only the information contained in the map is deleted, so you can immediately begin to put more information in, without having to create another map. If you don't add information, the map will be deleted when you leave it. (Empty maps don't persist in LightWave.)

To partially clear a map, select the points or polys that you would like to clear. Click the button, and the values for those points and polys will be gone.

This is extremely useful, as it allows you to correct your maps on a point-by-point basis.



**NOTE:** Don't use this function to clear a copy of a map to make symmetrical morphs, etc. Instead, just select the part you want to keep when you copy the map in the first place.

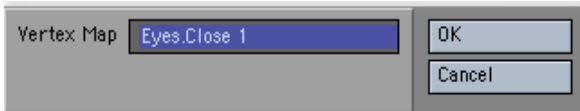


## Edit Maps

### Copy Vertex Map

Copy Vertex Map (**Map > General: Edit Maps > Copy Vertex Map**)

To copy a **Vertex Map**, select the map you want to copy from the **Vmap Bar**, and click on the command.



A dialog will appear that allows you to name the new **Vertex Map**. By default, it will have the name of the map you are copying, with a number appended to it. You can accept that name, or type in any name you like. (It is strongly suggested that you develop the habit of using meaningful names for your maps.)

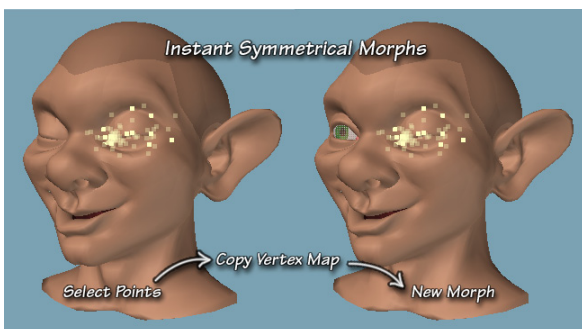
Click **OK**, and the copy is made and selected, ready for you to work on.

All values will be copied, no matter which kind of map you chose. (Weight, Texture (UV), **Endomorphs**, **Color Vertex Maps**, or Selection Sets.)

To copy an entire map, make sure that no points or polys are selected before you click on the command.

To copy a partial map, select the points or polys that you would like to copy. Click the button, and only the values for those points and polys will be copied to the new map.

This is extremely useful, because it allows you to make perfect pairs of **Endomorphs**, **Weight Maps**, etc. by making a full map using symmetry, and then copying half of it to a new map.



For instance, you might make a symmetrical morph of your character closing both eyes. Call it "Eyes.close" Then copy the left eye only, use this button, and call the new **Endomorph** "Eyes. IWink" Instant flexible facial expressions!



**NOTE:** Be aware that there's no list here to choose from; you must do that in the **Vmap Bar**.

### Delete Vertex Map

Delete Vertex Map (**Map > General: Edit Maps > Delete Vertex Map**)

To delete an entire **Vertex Map**, select the map you want to delete from the **Vmap Bar**, and click on the command. (**Weight**, **Texture (UV)**, **Endomorphs**, **Color Vertex Maps**, or **Selection Sets** may all be deleted with this command.)

The entire map will be deleted, whether any points and polys are selected or not. (If you wish to delete only some of the points or polys from your map, use the **Clear Map** command, instead.)

This is different from the **Clear Map** command because it doesn't simply empty your map. It completely deletes it, including its name.



**WARNING:** This command will delete all maps that use the name of the map chosen in the **Vmap Bar** from all open objects. For example, if you have a model with a morph you're calling "Eyes.Blink," and you decide that this particular character doesn't need that morph, and you might as well delete it, be careful. As soon as you click, all the "Eyes.Blink" morphs will be deleted from all the characters that are open at the time, whether they are visible or not.

Be aware that there's no list here to choose from; you must do that in the **Vmap Bar**.

### Rename Vertex Map

Rename Vertex Map (**Map > General: Edit Maps > Rename Vertex Map**)

To rename a **Vertex Map**, select the map you want to rename from the **Vmap Bar**, and click on the command. (**Weight**, **Texture (UV)**, **Endomorphs**, **Color Vertex Maps**, or **Selection Sets** may all be renamed with this command.)

Simply type the desired name into the text field and click **OK**.

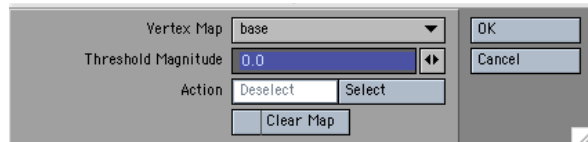
Be aware that there's no list here to choose from; you must do that in the **Vmap Bar**.



## More

### Cull Map

This command opens a dialog that allows you to select some or all of the points in a **Vertex Map**, and to automatically delete the ones that fall below a certain threshold.



To use it, choose the map you wish to cull from the drop down list. All types of **Vertex Maps** (**Weight**, **Texture (UV)**, **Endomorphs**, **Color Vertex Maps**, or **Selection Sets**) from all open objects will be listed here.

#### Deselect/Select

Use the **Deselect** or **Select** buttons if you want to deselect or select the points in the map.

#### Clear Map

Enable **Clear Map** if you wish points to be automatically cleared from the map. If you enable it, no points will be selected or deselected; they will simply be cleared if they fall below the threshold.

Then type the number you desire for the **Threshold** into the **Threshold Magnitude** field, and hit **Enter**. (It's important to do it this way if you are in the habit of using **Return** or **Enter** to move to the next field, because in this case it closes the dialog. If, on the other hand, you use the **Tab** key to move from field to field, go ahead and fill in the box in order, then click **OK**.)

Any points that fall below the threshold will be acted upon. (**Deselected**, **Selected**, or **Cleared**.)



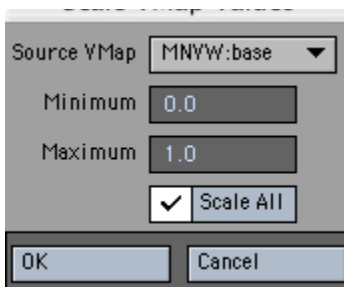
**NOTE:** The **Threshold** for most uses is between 2 and 0, although if you have pushed your weights, your numbers may be higher.



**HINT:** To select only high threshold points, select all points, and choose **Deselect** to drop those below the desired threshold, leaving only the higher ones selected.

## Normalize

This command allows you to scale the values of your **Vertex Maps**. It works on the entire map, whether any points have been selected or not.



#### Source VMap

Choose the map you wish to scale from the drop down menu. All kinds of **Vertex Maps** are listed here, with the type of map in a code to the left.

```
MNYW:base
WGHT:DFin1Weight
WGHT:DFin2Weight
WGHT:DFin3Weight
MORF:Dorsal.Droop
RGBA:FinGold
RGBA:FishGold
TXUV:Goldfish_UV
MBAL:MetaBall.Influence
MBAL:MetaBall.Radius
PICK:Midline Points
WGHT:PushWeight
MORF:Tail.Flow
RGBA:TryThis
RGBA:Vertex Color
```

**Weight Maps** are listed as WGHT.

**Texture Maps** are listed as TXUV.

**Morph Maps** are listed as MORF.

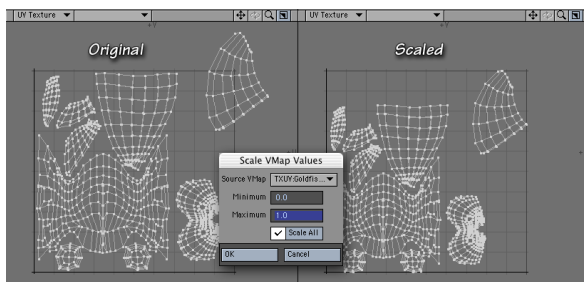
**Color Maps** are listed as RGBA or RGB (if they don't include an Alpha Channel).

**Selection Set Maps** are listed as PICK.

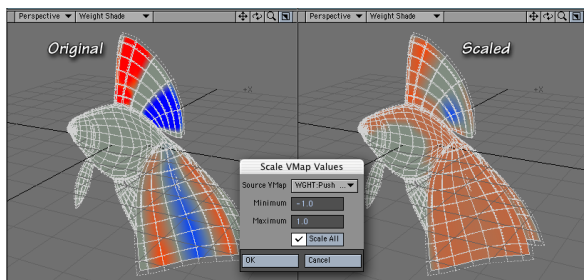
**Metaballs** are listed as MBAL.



**Minimum, Maximum** — Select a **Minimum** and **Maximum** value for the map. The entire range will be scaled, not clipped, to fit between these two values.



If you look at the result with this **UV Map**, you can see that the entire map was scaled so that none of the points were outside the range specified. (**Minimum** 0, **Maximum** 1.0).



You can see it again with this **Weight Map**, which had values that were pushed far beyond 100% to -100% on the fin, and normal weights within that range on the tail. Using a scaling range with a **Minimum** of -1.0 and a **Maximum** of 1.0 caused most of the negative weights to become positive, since the pushed positive was stronger than the negative. (Points that appear to remain at zero were not in the map, since no weight was assigned when the map was created. No weight is the same as a weight of zero, as you can see.)



**NOTE:** You'll have similar results with all **Vertex Maps**.

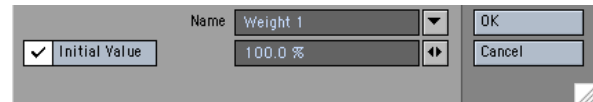
## Normal Baker

**Normal Baker** (**Map > General: More > Normalize Map**) will store the (normalized) point normal for each vertex as three numbers into a **VMAP** called PNT\_NORMALS. The values, starting with the first, represent the X, Y and Z directions of the vertex's normal.

## Weight

### New Weight Map

This command allows you to create a new **Weight Map**. It opens the **Create Weight Map** dialog, exactly like choosing **W** and **(new)** from the **Vmap Bar** at the bottom of the screen.



### Name

To name your map, either type a name into the text field, or choose an existing name from the drop down list to the right. The list contains all the names of all the **Weight Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

It's recommended that you choose a meaningful name, which will enable you to easily tell what the map is designed to do. This is especially true if you are working with other people; in fact, in those circumstances, there might be names that you are required to use, depending on the function of the map.

### Initial Value

Choose an **Initial Value** by clicking in the box to enable it, and typing the value you desire into the text field, or using the spinners.

Choosing an **Initial Value**, even if it's a value of zero, will place all the points into the map. (Unassigned points are not the same as points with a value of zero, even though they might look the same in the viewport **Weight Shade** render.) This may or may not be desirable because every bit of information takes up memory.



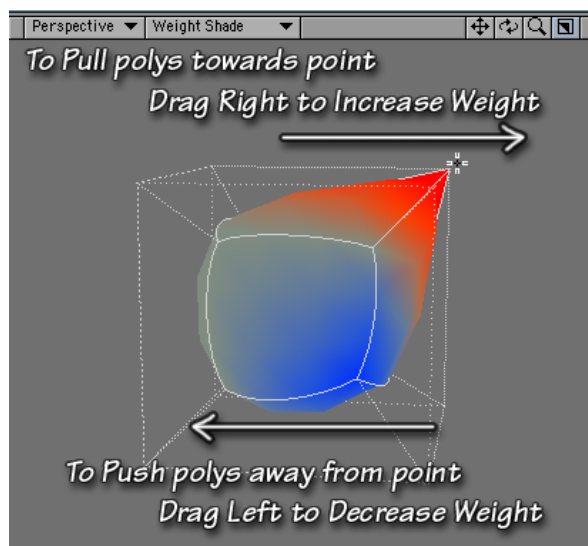
## Weights

This button allows you to edit existing **Weight Maps** by assigning weights to your points interactively.

It's best used with one of your viewports set to **Weight Render** mode, and with the **Numeric** panel open. (These things will allow you to see what you are doing.)

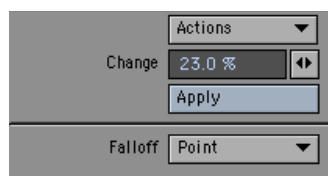


Choose the **Weight Map** you want to edit from the **Vmap Bar** at the bottom of the screen.



Place the tool on the point you wish to edit, and drag right to increase the weight, and left to decrease it. You can see the weight you are assigning in the **Numeric** panel, in the **Change** field. You may also be able to see it in the **Weight Shade** viewport, or in the shape of the object, depending on the **Weight Map** you are using.

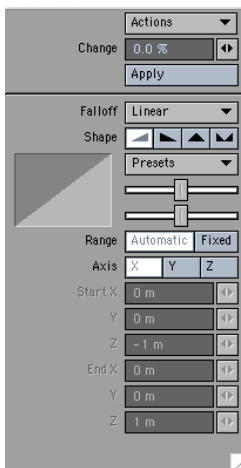
A weight of zero will have a grey-green color. Positive weight values will approach bright red. Negative values will approach bright blue.



## Change

The number in the **Change** field is a relative number, not an absolute. In other words, it tells you how much of a difference you've made in the weight, not what the weight is. This tool doesn't work particularly well for assigning specific weights to selected points. For that, you should use the **Set Map Value** tool. The strength of this tool lies in its interactivity.

There's nothing better to assign **Subpatch Weights** with, because you can see the results in real time as you work. Judicious use of the tool can save geometry, since you can sharpen edges with weights, instead of cuts.



## Falloff

Like many other tools in Modeler, this one allows you to adjust the **Falloff** range.

By default, the range is **Point**. In other words, the tool will only affect the point you are working with.

You can also assign other **Falloff** types:

**None** will assign the weight to all selected points/polys equally (or the entire model if none are selected.)

**Linear** and **Radial** allow you to interactively set a range with your **RMB**, or to use a number of presets.

**Polygon** applies the weight equally to all the points of all the polys which share the point under your tool.

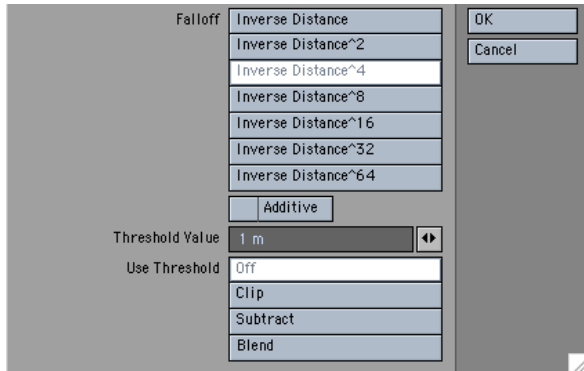
**Point Radial** allows you to set a falloff with the **RMB** that remains centered on the tool as you move it around. (Rather like the airbrush, but you drag to change the weight.)



## Bone Weight

This tool creates **Weight Maps** for bones, based on the choices you make within its dialog box. The map will have the same name as the bone that it's associated with. For example, if the bone is called Bone01, the map will be called Bone01 as well.

To use it, you must have both bone and geometry in foreground layers. Select the bone you wish to create a **Weight Map** for. (If no bones are selected, maps will be made for all foreground bones that don't have one.)



### Falloff

A dialog will open that allows you to choose the falloff algorithm for the bone. The various choices correspond to the **Falloffs** that you can use on bones when you aren't using **Weight Maps**. But with this tool you can start there, and then modify the weight to suit your own needs. (You can also see what the **Falloff** is doing in any viewport set to **Weight Shade**, where positive weights are red, zero is grey-green, and negative weights are blue.)

### Additive

If you enable **Additive**, the weight will be added to any existing value. So, if you have two bones with the same name, say both left and right arm bones called armbone.01, the effects from both would be shown on the same map.

### Threshold Value

The **Threshold Value** is a capsule-shaped region around the bone. You can type the value into the text box, or use the mini-slider.

### Use Threshold

The **Threshold Value** can be used in several ways.

**Off** will simply ignore it.

**Clip** will take any weight outside the threshold and set it to zero, which can have the effect of giving fairly sharp edges to the **Weight Map**.

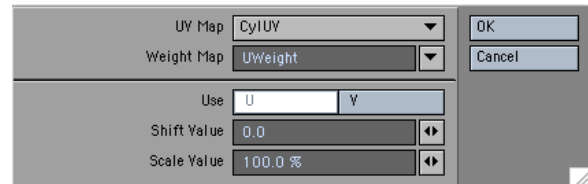
**Subtract** will subtract the weight at the threshold from all the weights in the map; so the weight inside the threshold progresses smoothly to zero, the weight at the threshold is zero, and weights beyond the threshold become negative.

**Blend** behaves like **Subtract** inside the threshold, but clips everything above the threshold, so all the weight beyond the threshold is zero.

## UV to Weight

This button allows you to create a **Weight Map** based on either the **U** or **V** value of a **UV Map**. (It does not allow you to use a greyscale map that you can color and apply the way you would apply a **Texture UV Map**. If that's what you're after, you're looking for **Textured Point**).

To use it, create a **UV Map** that has the points arranged along either the **U** or **V** axis, the way that you would like the weights assigned.



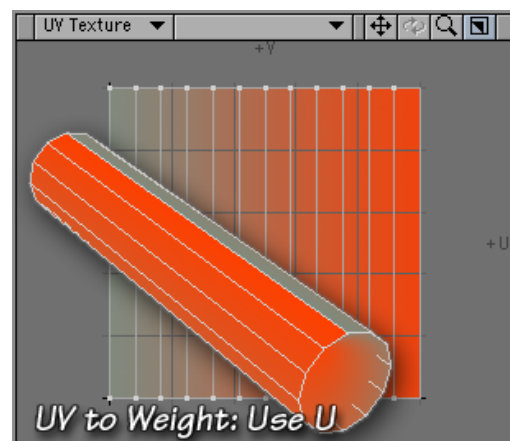
Click on the tool to open the associated dialog.

### UV Map

Choose the **UV Map** you have prepared from the **UV Map** drop down list, and type the name for the new **Weight Map** into the **Weight Map** text field (or choose it from the list, if you prefer to use an existing name. All **Weight Maps** used in this session will appear in the list, whether or not the object they are associated with remains open, and whether or not any points have been assigned to the map.)

### Use

Choose whether you want to use the **U** axis or the **V** axis to create the map.

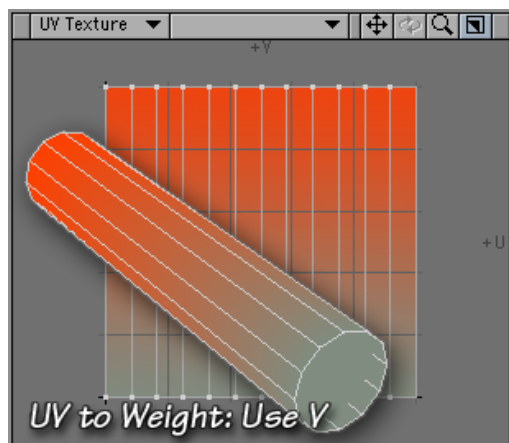


If you use the **U** axis, all the points on the left edge of the map will





receive a value of 0%, the points on the right edge will receive a value of 100%, and the points in between will have values in between, of course.



If you use the **V** axis, the points at the bottom are 0%, the points at the top are 100%, and those in-between are in between, of course.

### Shift Value

This option adds the number entered to the 0 to 1 range. For example, if you entered a **Shift Value** of -1, the range would become -1 to 0, giving you a map that had weights from -100% to 0%. Entering a **Shift Value** of 1 would give you a range of 100% to 200%, and so on.

### Scale Value

This option scales the values by the number entered. For example, if you entered a **Scale Value** of 200%, you would get a weight range from 0% to 200%. A **Scale Value** of -150% would give you a weight range of 0% to -150%, and so on.

These values work together. So, for example, if you wanted a range between -100% and 100%, you would enter a **Shift Value** of -1, and a **Scale Value** of 200%.

Neither **Shift** nor **Scale** are either clipped or constrained.

## PatchWeight2Weight

This command allows you to transfer the information in a **SubPatch Weight Map** to a new map.



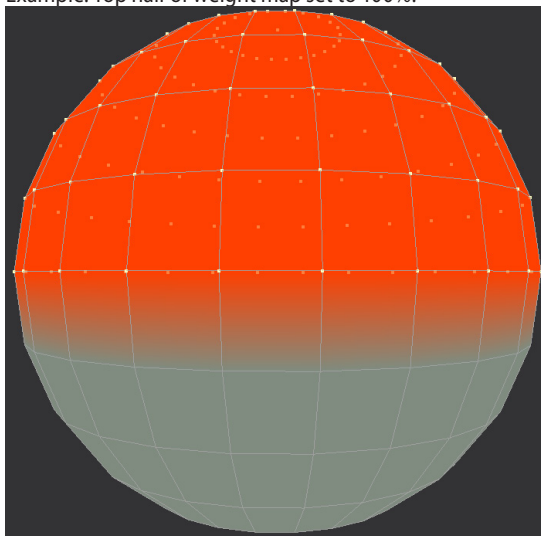
To use it, click to open the dialog box. Choose the **Subpatch Weight Map** you wish to use from the drop down list (**base**, in most cases,) then type the name of the new (or existing) **Weight Map** into the **Weight Map** field. The new map will be created, and chosen in the **Vmap Bar**.

If you used an existing map, the values in that map will be replaced with the values in the **SubPatch Weight Map**.

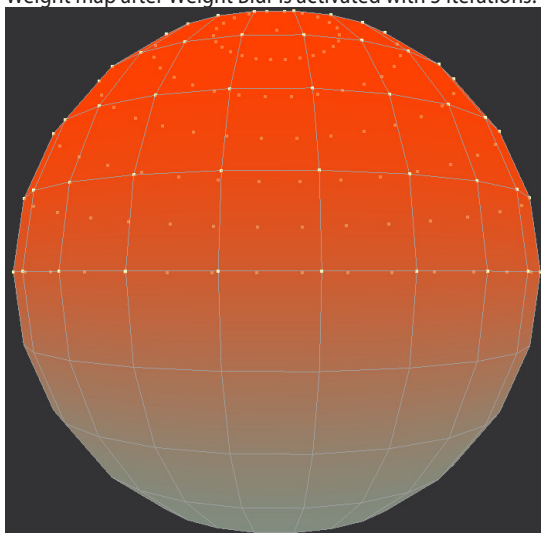
## Weight Blur

Weight Blur takes the current weight map and creates a gradient based on the upper and lower values.

Example: Top half of weight map set to 100%.



Weight map after Weight Blur is activated with 5 iterations.



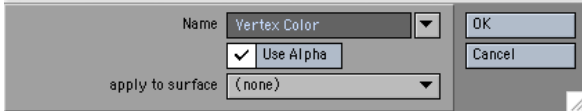




## Color

### New Color Map

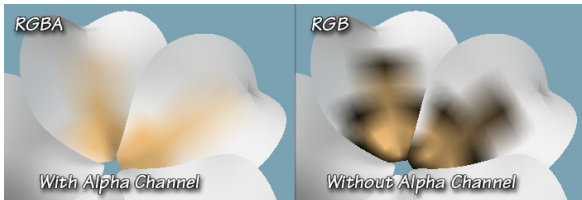
This button allows you to create a new **Color Map**. It opens the **Create Vertex Color Map** dialog, exactly like choosing **C** and **(new)** from the **Vmap Bar** at the bottom of the screen.



Choose a new name, whether or not to add an **Alpha Channel**, and which surface you want the map applied to.

### Name

To name your map, either type a name into the text field, or choose an existing name from the drop down list to the right. The list contains all the names of all the **Color Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.



### Use Alpha

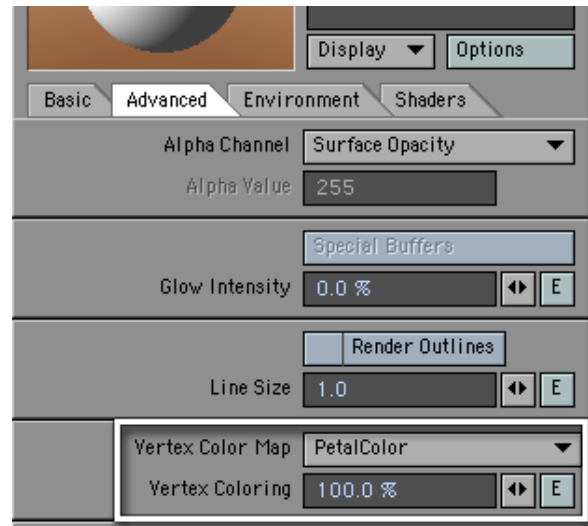
It's recommended that you use an **Alpha Channel** in your texture, unless you intend to completely color the object. If you don't, you are likely to find that you have black artifacts around the edges when you use the **Airbrush**. This happens when a polygon has some points that are in the **Color Map**, and some that aren't. (Points that are unassigned are not the same as points that are assigned a value of zero.) The **Alpha Channel** eliminates this, and allows better feathering, so the color you're applying blends smoothly into the base color of the object.

In general, **Alpha Channels** allow you more flexibility, and are considered superior; but they do take up memory, and aren't necessary if there aren't any unassigned vertices.

### Apply to surface

Choose which surface you would like the new **Color Map** applied to from the drop down list. All the surfaces in the current object will be there.

You don't have to choose a surface; but if you don't, the new map won't be visible on anything. You will be able to assign values to it, and even color on it; but you won't see it until it's applied to a surface. So be careful. You don't want to waste time, or become frustrated, because you think the airbrush isn't working, when the real problem is that you are painting on an invisible map!

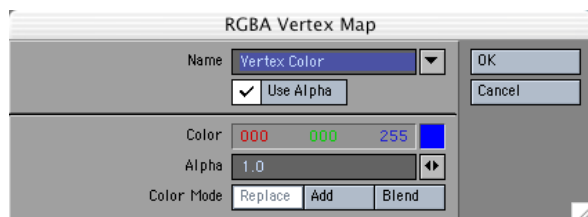


If you don't choose a surface, you can always choose one later by opening the **Surface Editor**, and choosing a map from the **Vertex Color Map** list in the **Advanced Tab**. If you do, it will automatically be added here, at 100% strength. You can increase or decrease the strength in the **Surface Editor** at any time.



## Point Color

This button allows you to assign a color to any selected points. To use it, select the points you would like to add the color to. (If no points are selected, this tool won't do anything.)



Click the button to open the **RGBA Vertex Map** dialog.

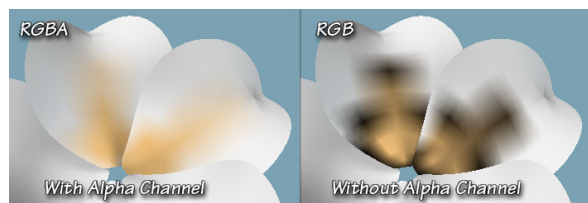
### Name

Type the name of the **Vertex Color Map** you would like to work on into the **Name** field, or choose it from the drop down list to the right. The list contains all the names of all the **Color Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

You can start a new map either by typing the new name into the field, or by choosing (**new**) from the list, and then typing a new name into the field. However, be aware that this map won't be assigned to any surface, so you won't be able to see the results of your choices until you assign it to one. (You'll have to open the **Surface Editor**, and assign the map to a visible surface by selecting the surface, and then choosing this map from the **Vertex Color Map** list in the **Advanced Tab**). So be careful. If you don't see any change, don't assume that the tool didn't work. Check to make sure that the map has been assigned to a surface that's visible in one of your viewpoints.

### Use Alpha

Decide whether or not to use an **Alpha Map** with the selection.



If you are making a new map, it's recommended that you do, unless you intend to completely color the object. Although the color will look fine, and blend nicely when using this tool, if you ever use the **Airbrush** on this map, you'll have black artifacts around the edges where the color doesn't feather properly.

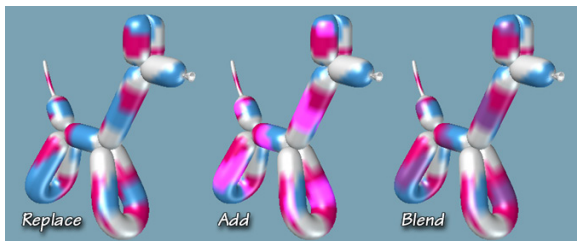
If you are using an already existing, you must choose to use the **Alpha** if it has one, or not to use it if it doesn't. The choice will not be disabled; but if you disable **Use Alpha** on an **RGBA Map**, or enable it on an **RGB Map**, **Point Color** will simply make a new map with the same name, and follow your choices. Since that map hasn't been assigned to any surface, you won't be able to see any results. (But, you can see them if you go to the **Surface Editor Advanced Tab**, find the other map with the same name, and assign it to the surface).

### Color

Pick a color to put on all the points, using the normal **LightWave Color Requester**. Drag the numbers to choose an **RGB** color, or right-click to toggle to **HSV** and then drag, or click on the color swatch to open the **Color Picker** you have chosen to use.

### Color Mode

In the **Color Mode** field, you can choose how this color will be applied to the points.



### Replace

**Replace** removes any color currently there, and replaces it with this one.

### Add

Adds the **RGB** value of the current color to the **RGB** value of the color already there, if any. This may result in **RGB** values higher than 255, of course, which gives you ultra-bright colors.

### Blend

Adds the **RGB** values of both new and existing colors and averages them by dividing the result by two. The value won't exceed 255, unless the existing color did.

### Alpha

Finally, if you enabled **Use Alpha**, specify the amount of alpha you want to use.



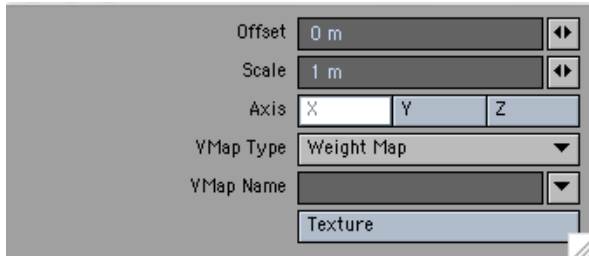
**WARNING:** Unlike most of the other **Alpha** fields, this one isn't a percentage. That means that 1 is equivalent to 100%. So, if you want "normal" coverage, enter 1 in this field. (If you enter 100, you'll really get 10000%, and your color will be far darker than you expect.) If you enter 0, which is the default, you won't be able to see any color at all, because it will be totally masked. (You can see if it's really there, though, by checking the **Point Information (I)**. You can also change the alpha there, so you can see the color.)

(It's important to enter the **Alpha** last if you are in the habit of using **Return** or **Enter** to move to the next field, because in this case it closes the dialog. If, on the other hand, you use the **Tab** key to move from field to field, go ahead and fill in the box in order, then click **OK**.)



## Textured Point

This button allows you to assign the luminosity values of a normal **LightWave Texture** (image, procedural, or gradient) to a **Weight** or **Morph Map**, or the **Color** values to a **Color Map**. Bear in mind, though, that the texture is assigned to points, not polys (hence the name) so, unless you have a very heavy mesh, you won't get the detail you'd get by using the textures on a surface. Other than that, though, it works in a very similar way.



### Offset

**Offset** will move a 3D texture through the surface.

### Scale

**Scale** changes the scale on all three axes simultaneously.

### Axis

**Axis** allows you to change the **Projection Axis** for the texture.

### Vmap Type

Choose the type of map you want to apply the texture to from the **VMap Type** drop down menu. The available choices are **Weight Map**, **Morph Target**, or **Vertex Color**.

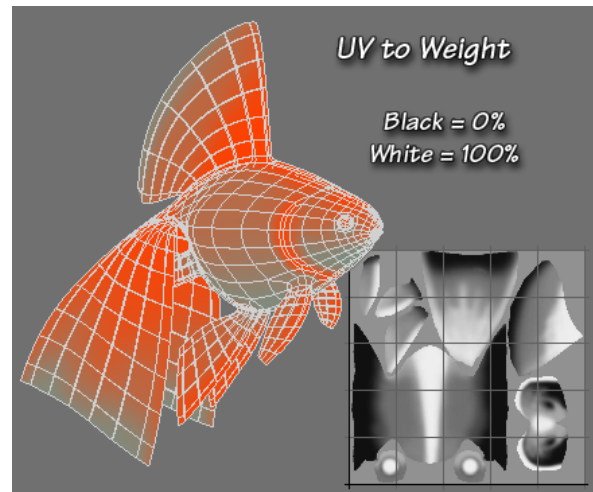
### VMap Name

Type the name of the **Vertex Map** you would like to work on into the **Name** field, or choose it from the drop down list to the right. The list contains all the names of all the maps of that type used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

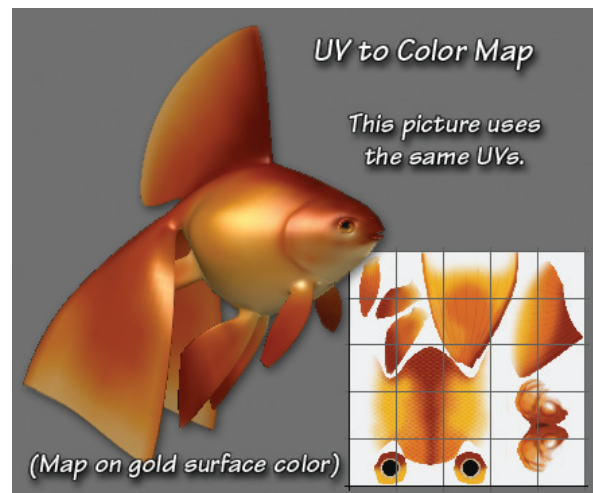
You can also type a new name into the field to start a new map. In the case of a **Color Map**, you can choose **(new)** and then type one in; but be aware that, as always, you won't be able to see it until it's been assigned to a Surface. (To do that, you'll need to open the **Surface Editor**, go to the **Advanced Tab**, and choose the map from the **Vertex Color Map** drop down.)

**Weight Maps**, **Morph Maps**, or existing **Color Maps** that have been assigned to a visible surface, will become selected in the **Vmap Bar**, and will be visible in any viewports capable of showing them. (ie. **Weight Maps** will be shown in viewports using the **Weight Map Render** mode, **Color Maps** in **Smooth Shade**, **Texture**, and **Textured Wire**, etc.)

Click on the **Texture** button to open the familiar **LightWave Texture** panel. This works the same as it does everywhere else. You'll find that the **Gradient** parameter choices are limited to **Previous Layer**, **X**, **Y**, **Z**, **Distance to Center**, and **Weight Map**; but other than that it's what you are used to.



As you work, you'll be able to see the texture affecting the map in real time. Colors will change, weights will go from red to green, and morphs will distort as you watch.



Although you can change the orientation and scaling from within the **Texture** dialog, as you normally would, you can also change them in the **Textured Point** dialog (labeled **Apply Texture**)



## Vertex Paint

### Overview

**Vertex Paint** is a Modeler plugin that lets you interactively “paint” a **Vertex Color Map** on objects and **Weight Maps** on **Skelegons**.

A **Vertex Color Map** is a LightWave feature that lets you assign colors to an object’s vertices (i.e., points). Vertex color maps are a basic LightWave feature so you can render their effect without using surface shader plugins. Although similar to Modeler’s **Airbrush** tool, **Vertex Paint** has its own independent interface for painting.

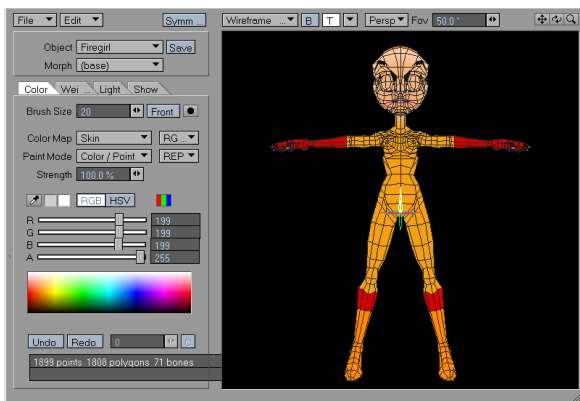
LightWave **Skelegons** let you define bones in Modeler. **Weight Maps** can provide additional deformation information. **Vertex Paint** provides a variety of features for you to set weights for **Skelegons**. You can see how the weight will deform the object in real-time while painting.

The **Vertex Loader** plugin can be used to load the bone information modified in Layout so that you can edit weights using **Vertex Paint**.

### How to launch Vertex Paint

Vertex Paint consists of two plugins: Vertex Paint and Vertex Loader.

If you want to edit a Vertex Color Map or Weight Maps (for Skelegons):

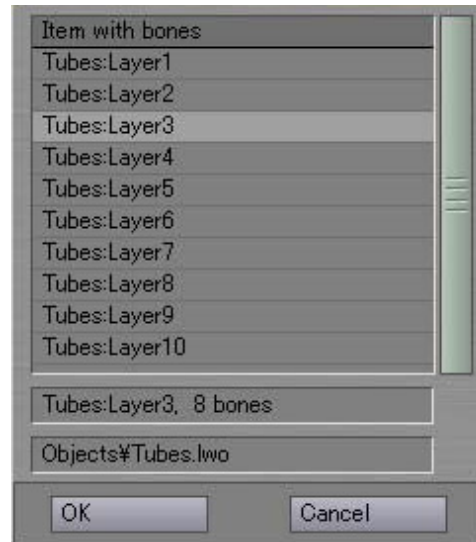


First, load an object you want to edit into Modeler. Then, choose **Map > Color: Vertex Paint**. For **Skelegon Weight Maps**, your object must already have **Skelegons**. The **Skelegon** names set by certain tools, such as **Skelegon Tree**, are used for each Skelegon’s **Weight Map** name.

### If you want to edit weight maps using Layout bones:

Use **Vertex Loader** to load objects with bones that were created from scratch in Layout, or bones that were converted from **Skelegons**, in order to edit the **Weight Maps** in **Vertex Paint**.

To use **Vertex Loader**, choose **File > Import > Vertex Loader**. First the **Load Scene** requester opens; select a scene file that has object(s) with bones. After you select a scene file, the dialog box shown below appears. There you select an object that you want to edit and click **OK**.



The selected object is automatically loaded into Modeler and displayed in **Vertex Paint** with its bones. If the object is already loaded, only the bones are loaded from the scene file. Weight maps set on the **Bone Properties** panel in Layout are used for each corresponding bone.

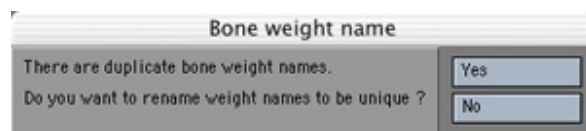


**NOTE:** Before launching **Vertex Loader**, make sure Layout’s **Content Directory** setting matches Modeler’s.

### If multiple skelegons share the same weight map name:

If a single weight map is shared by multiple **Skelegons** (or bones), **Vertex Paint** displays the dialog box below. Generally, one weight map should correspond to only one bone in **Vertex Paint**.

If you click **Yes**, **Vertex Paint** automatically uniquely renames the map so there is no longer a duplicate. If you click **No**, **Vertex Paint** makes an exception and allows a “shared” weight map.

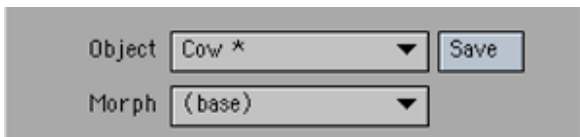






## Saving the Data

Changes made in **Vertex Paint** are **not** automatically updated in the Modeler object. To reflect the **Vertex Paint** changes, click the **Save** button or choose **Save To Modeler** from the **Edit** menu. (An asterisk (\*) next to the object name means the object has additional information not yet updated.)



The following is what **Vertex Paint** updates:

**Vertex Color** (RGB VMap, RGBA VMap)

**Weight** (WGHT VMap)

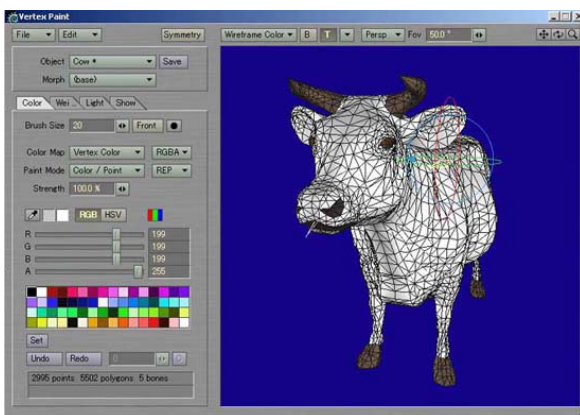
**Morph** (MORF VMap, SPOT VMap)

**Bone name** (BONE, PART PTag)

**Bone Weight name** (BNWT PTag)

## Vertex Paint: User Interface

Choose **Map > Color: Vertex Paint** to launch Vertex Paint. The interface appears below.



### Changing the size of the main panel

To change the size of the main panel, drag the mark at the bottom-right corner with **LMB**. The display will be jumbled while dragging, but once you release the mouse button, it will display properly.

## Manipulating the 3D window

The large window on the right side of the panel is a 3D view where you can paint the vertex color of the loaded object or edit the weight value of bones. When you first run Vertex Paint, the 3D view displays the object you are going to edit. The 3D view is changed in much the same way the Perspective view is in Modeler.



### Rotating the view

There are two ways to rotate the 3D view. One is to drag your **LMB** on the rotate button at the upper right of the panel. The other is to drag your **LMB** while pressing **Alt** key. Both ways rotate object in the direction you drag.



### Moving the view

There are two ways to move the 3D view. One is to drag your **LMB** on the pan button at the upper right of the panel. The other is to drag your **LMB** while pressing **Shift** key. Both ways move object in the direction you drag.



### Scaling the view

There are two ways to rotate the 3D view. One is to drag your **LMB** on the zoom button at the upper right of the panel. The other is to drag your **LMB** while pressing **Alt** and **Ctrl** keys. Moving your mouse cursor to the right magnifies the object, and moving to the left reduces the object.

## Perspective and Orthogonal view

Vertex Paint supports both the perspective projection and the orthogonal projection to display objects in 3D space. To display the object in perspective projection, select **Persp** from the 3D view info pop-up menu. To display the object in orthogonal projection, select **Ortho** from the 3D view info pop-up menu. You might find that you can paint vertices easier with the **Ortho** mode.

## Changing the camera zoom factor

You can change the camera's zoom factor when you are in the **Persp** mode. Dragging the mini-slider next to **Fov** at the top of the panel left or right, zoom factor changes. Increasing the value, the camera lens will be a telephoto lens, and decreasing the value, it will be more wide-angle.

## Changing the view

Click the down-arrow popup at the top of the panel to change the 3D view display.

- Front View** ... displays the object from the front side.
- Back View** ... displays the object from the back side.
- Right View** ... displays the object from the right side.
- Left View** ... displays the object from the left side.
- Top View** ... displays the object from the top side.
- Bottom View** ... displays the object from the bottom side.
- Fit Model** ... displays the object at the center of the display.
- Fit Bone** ... displays the selected bone at the center of the display.

## Switching the display mode

Vertex Paint can display objects using various display modes depending upon your needs. To change, use the pop-up menu at the upper left of the 3D view.

**Points** — Displays the points that make up the object. The vertex color is set for the point color. Light source is not used.

**Wireframe** — Displays the objects in wireframe. The vertex color is set for each vertex of the lines, if the color is different between the start and end point of the line, it displays a gradient color. Light source is not used.



**Frontface** — Displays only polygons with normals facing the camera in wireframe. Light source is not used.

**Vertex Color** — Displays the polygons in color, using the vertex color map currently selected. Light source is not used.

**Wireframe Color** — Displays the polygons in color using the vertex color map currently selected, and also displays wireframe over it. Light source is not used.

**Lighting Shade** — Displays the objects using the light source set on the **Light** menu. The object's surface information is used for the polygons' color attributes. No vertex color shown.

**Weight Value** — Displays the weight value assigned to the currently selected bone in red gradient. If no bones exist, it displays in grey.

**Weight Ratio** — Displays the weight ratio of the vertices for the selected bone. Stronger color means more weight. Each bone has its own unique color. color. If no bones exist, the object is displayed in grey.

## **B** **B Button**

Clicking the **B** button on, switches to the "Blending display mode". This is a blending display using the vertex color alpha in the **Vertex Color** and the **Wireframe Color** modes. This also enables the translucent display of the texture, using its alpha plane.

## **T** **T Button**

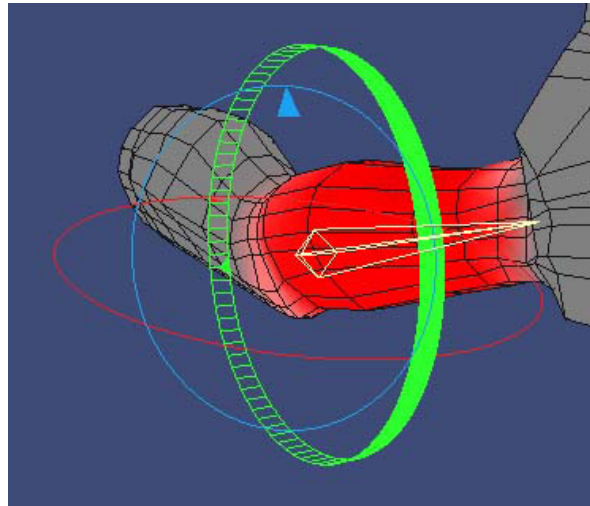
Clicking the **T** button on, switches to the "Texture display mode". This is a texture mapping display visible in the **Vertex Color**, **Wireframe Color** and **Lighting Shade** mode.

## **Rotating a Bone**

When the rotate button next to the bone **VMap** name on the Weight menu tab is *showing*, you can rotate the bones and see how the object deforms. Each ring means: Red=Heading (Y axis), Green=Pitch (X axis), and Blue=Bank (Z axis).



NOTE: Clicking the button next to the bone VMap name cycles it through its settings (rotate, scale, move).



To rotate the bone, drag on the 3D window with **LMB** while pressing **Ctrl** key. The bone rotates as your mouse goes.

To rotate along only one axis, hold **Ctrl** and select the ring of the axis you want to rotate with **LMB**, then drag it. Selected ring is shown as a wide-band ring.

To reset the rotated bone at the default (Rest Position), with the Weight Tab selected, choose **Reset the Bone** from the **Edit** pop-up menu. To reset all the bones, choose **Reset All Bones**.

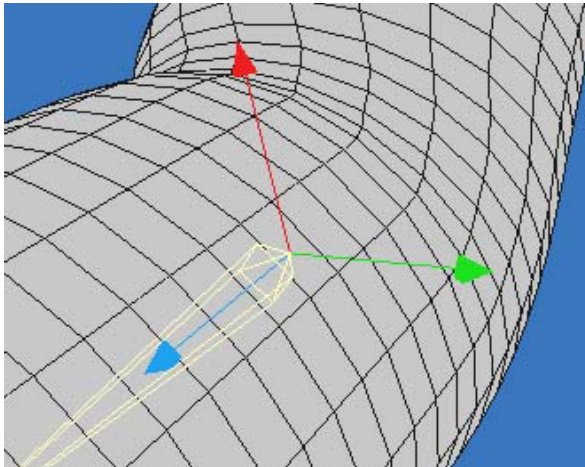


## Moving a Bone

When the move button next to the bone name on the Weight menu tab is *showing*, you can move the bones and see the deformation. A coordinate axis as shown below is displayed for the current bone. Each axis represents: Red=X axis, Green=Y axis, and Blue=Z axis.



NOTE: Clicking the button next to the bone VMap name cycles it through its settings (rotate, scale, move).



To move the bone, hold **Ctrl** and drag on the 3D window with **LMB**.

To move along only one axis, hold **Ctrl** and select the axis you want to move and then drag it. The selected axis is shown as a bold line.

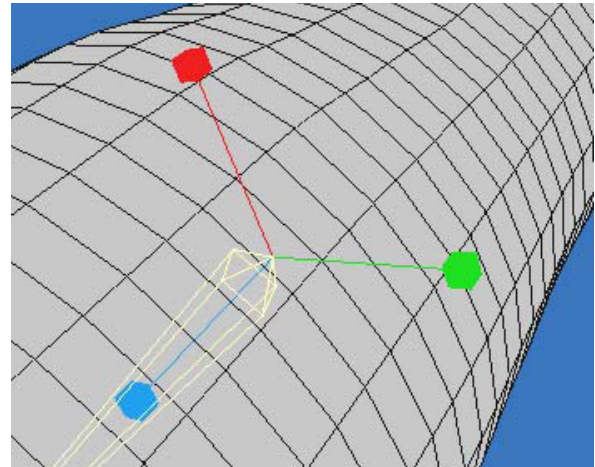


## Scaling a Bone

When the zoom button next to the bone name on the Weight menu tab is showing, you can scale bones up or down. A coordinate axis as shown below is displayed for the current bone. Each axis represents: Red=X axis, Green=Y axis, and Blue=Z axis.



NOTE: Clicking the button next to the bone VMap name cycles it through its settings (rotate, scale, move).



To scale the bone up/down, hold **Ctrl** and drag on the 3D window with **LMB**. Dragging your mouse up/right scales up the bone, and dragging your mouse to down/left scales down the bone.

To scale along only one axis, hold **Ctrl** and select the axis you want to scale with **LMB** then drag it. The selected axis is shown as a bold line.

## Changing the Mode

Vertex Paint has four working (editing) modes. You can change those modes by clicking on the menu tabs at the center-left of the window. Note that the **Edit** pop-up menu will show the different menus depending on what mode you are in.

- **Color:** Vertex color edit mode
- **Weight:** Bone weight edit mode
- **Light:** Light setting mode
- **Show:** Polygons Show/Hide setting mode

## Displaying a Morph Target

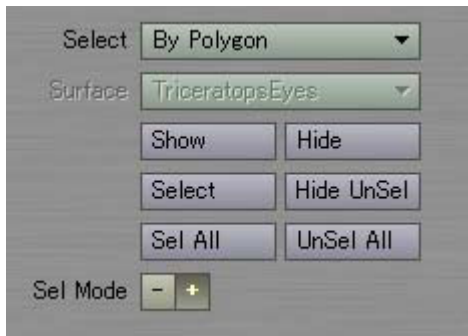
If the object contains EndoMorph targets, you can choose the displayed target on the **Morph** pop-up menu.





## Vertex Paint: Showing / Hiding Polygons

### Selecting / Deselecting Polygons



### Select / Deselect Polygons Using Your Mouse

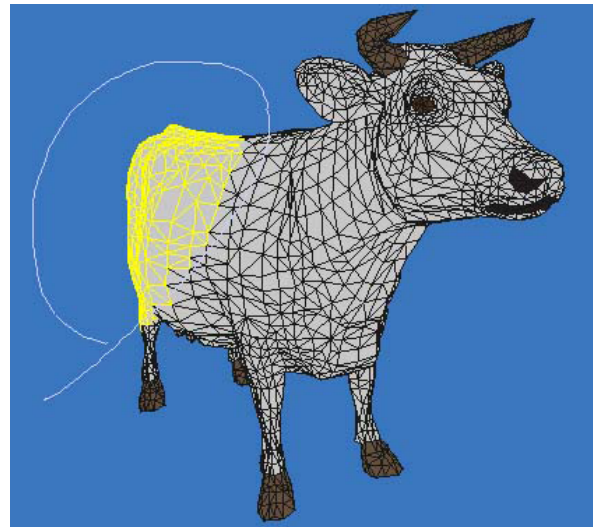
When you are using the + **Sel Mode**, you can *select* polygons by dragging your **LMB** over them in the 3D view.

When you are using the - **Sel Mode**, you can *deselect* polygons by dragging your **LMB** over them in the 3D view. Dragging with your **RMB** deselects the polygon regardless of what selection mode you are in.

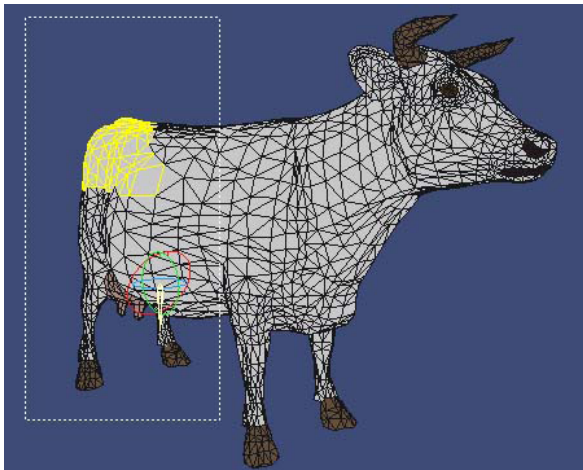
Hold the **Ctrl** and **Shift** keys as you drag your **LMB** to select/unselect using a bounding box.

Hold the **Ctrl** and **Shift** keys as you drag your **RMB** to select/unselect using a lasso.

You can toggle the **Sel Mode** by pressing the **SPACE** key.



Lasso selection



Bounding box selection



## Show / Hide Polygons By Polygon

Vertex Paint can show/hide polygons either by polygon or by surface. Select **By Polygon** from the **Select** pop-up menu to manipulate by polygon. There are also some selection and hiding functions available in the **Edit** menu.



Show	shows all polygons
Select	selects all polygons
Hide	hides selected polygons
Hide UnSel	hide unselected polygons
Sel All	selects shown polygons
UnSel All	unselects shown polygons

## Default Mouse Operations

The following table is a list of the mouse operations. You can customise the operation on the Preference panel (**File > Preferences**).

Operation	Command	Action
LMB ALT-SHIFT	View Move	Move the view
LMB ALT	View Rotate	Rotate the view
LMB CTRL-ALT	View Zoom	Scale the view
LMB	View Tool Move	Move the view (using the icon)
LMB	View Tool Rotate	Rotate the view (using the icon)
LMB	View Tool Zoom	Scale the view (using the icon)
LMB	Color Paint	Paint the vertex color (in Color mode)
RMB	Color Radius	Show size of the color paint brush (in Color mode)
LMB CTRL-SHIFT	Color PickColor	Select color from the 3D view (in Color mode)
LMB	Color Palette	Select color from the color palette (in Color mode)
LMB	Weight Paint	Paint bone weight (in Weight mode)
LMB CTRL	Bone Navigate	Rotate/Move/Scale the bone
LMB SHIFT	Bone Select	Select the bone
RMB	Weight Radius	Show size of the weight paint brush (in Weight mode)
LMB CTRL-SHIFT	Weight Rectangle Select	Rectangle selection of the point (in Weight / Info mode)
RMB CTRL-SHIFT	Weight Lasso Select	Lasso selection of the point (in Weight / Info mode)
LMB	Light Move	Move the light source (in Light mode)
LMB	Show Select	Select the polygon (in Show mode)
RMB	Show UnSel	Deselect the polygon (in Show mode)
LMB CTRL-SHIFT	Show Rect Select	Rectangle selection of the polygon (in Show mode)
RMB CTRL-SHIFT	Show Lasso Select	Lasso selection of the polygon (in Show mode)



## Keyboard Shortcuts

Key	Action
a	Display object at the center of the window
A	Display current bone at the center of the window
,	Zoom out
.	Zoom in
F	Display object in Front view
T	Display object in Top view
S	Display object in Right side view
F1	Points Display mode
F2	Wireframe Display mode
F3	Front Face Display mode
F4	Vertex Color Display mode
F5	Wireframe Color Display mode
F6	Lighting Shade Display mode
F7	Weight Value Display mode
F8	Weight Ratio Display mode
u	Undo
Down	Select next bone
Up	Select previous bone
c	Copy current bone weight
v	Paste to current bone weight
V	Paste to current bone weight symmetrically in X axis
x	Clear current bone weight
+	Scale up current bone weight
-	Scale down current bone weight
r	Reset current bone rotation
R	Reset all bone rotation
*	Step forward
/	Step backward
t	Bone move mode
y	Bone rotation mode
h	Bone scale mode
-	Hide current polygons
=	Hide unselected polygons
	Invert Show/Hide polygons
SPACE	Switch polygon select/deselect mode



## Show / Hide Polygons By Surface

Select **By Surface** from the **Select** pop-up menu to manipulate by surface. You'll also need to select a surface from the **Surface** pop-up menu.



Show	shows polygons of selected surface
Select	selects polygons of selected surface
Hide	hides polygons of selected surface
Hide UnSel	hide unselected polygons
Sel All	selects shown polygons
UnSel All	unselects shown polygons

## Vertex Paint: Light Setting

### Setting the Light Source

Setting light sources on the Light tab determines the number of the lights, color properties, position, and other properties used in the “**Lighting Shade**” display mode. At the same time, it is used for the baking shading color with the **Burn Lighting Colors** command in the color mode. You can have up to eight different lights

### Choosing Light

To choose light, select one from the **Name** pop-up menu or click the light icon in the 3D view with your **LMB**, if it's visible.

Clicking on the checkbox next to the **Name** pop-up menu activates the light and affects the calculations of the light source on the scene. Click the checkbox off to deactivate the light source. Inactive light sources do not illuminate the scene.

## Light Type

In Vertex Paint, you can use distant lights, point lights, and spotlights. You can also set the ambient color for the lights, which affects the entire scene. Below is a table of the parameters available for each light. These are used for OpenGL's light settings. For more information on the parameters, refer to “OpenGL Programming Guide”, published by Addison Wesley (ISBN: 0-321-17348-1).

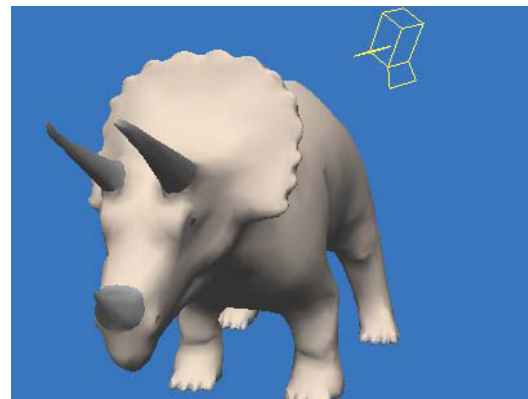
	Distant Light	Point Light	Spotlight
Diffuse	4	4	4
Ambient	4	4	4
Specular	4	4	4
Position	4	4	4
Spotlight Position	7	7	4
Spotlight Angle	7	7	4
Constant Attenuation	7	4	4
Linear Attenuation	7	4	4
Quadratic Attenuation	7	4	4

## Positioning the Light

To move a light, click on the **Position** field on the Light tab and enter the coordinates in the dialog that appears. You can also just grab the light in the 3D view with your **LMB** and drag it to wherever you like.

### Distant Light

Generally, a distant light is a light source which you only specify the light direction. In Vertex Paint, the direction of the distant light is measured as a vector from the light **Position** to the Origin. Moving the light position changes the vector direction.





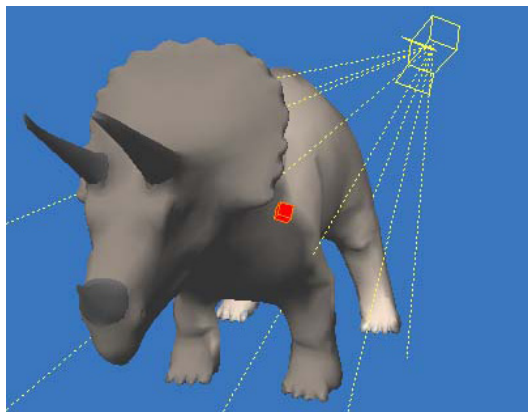
## Point Light

A Point light is a light source that has no direction and illuminates omnidirectionally (in 360 degrees). In Vertex Paint, the position of the point light is specified as the light **Position**.



## Spotlight

A Spotlight is a light source that has both position and direction information. In Vertex Paint, the position of the light is specified as the light **Position** and a coordinate where the spotlight “points” is defined by the **Spotlight Position**. The **Spotlight Position** is indicated by a red box so you can drag it in the 3D view. Also, the dotted lines coming out of the light represent the spotlight cone angle which can be changed by adjusting the **Spotlight Angle** setting.



## Vertex Paint: Vertex Color Paint

### Create Vertex Color Map

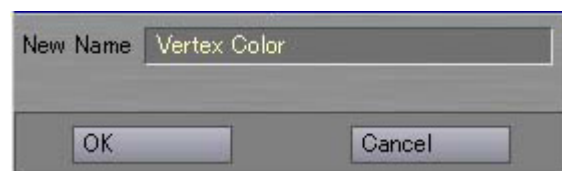
To use **Vertex Paint**, you must define at least one **vertex color map** *before* you begin painting. Note that if an object mesh has more than one vertex color map, Vertex Paint uses the first map as the current map.



To create a new Color map, select the Color tab, then choose **Edit > Create Vertex Color Map**. Enter a unique map name and click **OK**.

### Rename Vertex Color Map

To rename a vertex color map, select the Color tab and the map, then choose **Edit > Rename Vertex Color Map**.



Enter the new name and click **OK**. Note that you still need to update the object by clicking the **Save** button next to the **Object** pop-up menu.

### Delete Vertex Color Map

To delete a vertex color map, select the Color tab and the map, then choose **Edit > Delete Vertex Color Map**. The change is reflected when you click the **Save** button next to the **Object** pop-up menu.

## Vertex Color Map Type

There are two types of vertex color maps. **RGBA** means that the map includes the alpha information and **RGB** means that the map does not include the alpha information. In Vertex Paint, all the vertex color maps are treated as if they include the alpha. You can change the color map type by selecting the pop-up menu next to the **Color Map** option on the Color tab. The change is reflected when you click the **Save** button next to the **Object** pop-up menu.

## Component vs. Blending Display





The **Vertex Color** and **Wireframe Color** display modes can blend the display of vertex color (RGB) and Vertex alpha data. Clicking the **B** button on changes display mode to “alpha-blended.” The **Vertex Alpha** option on the Preference panel determines how to blend the alpha.

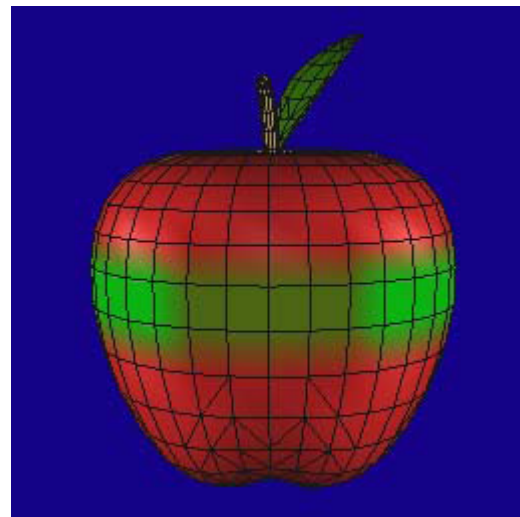
If you set the **Vertex Alpha** option to **Surface Color**, information on the vertex color’s alpha is displayed as a blended percentage value of the vertex color and surface color — the same way LightWave handles vertex color alpha.

$$\text{Color.Red} = \text{Vertex.Red} * \text{Vertex.Alpha} + \text{Surface.Red} * (1.0 - \text{Vertex.Alpha})$$

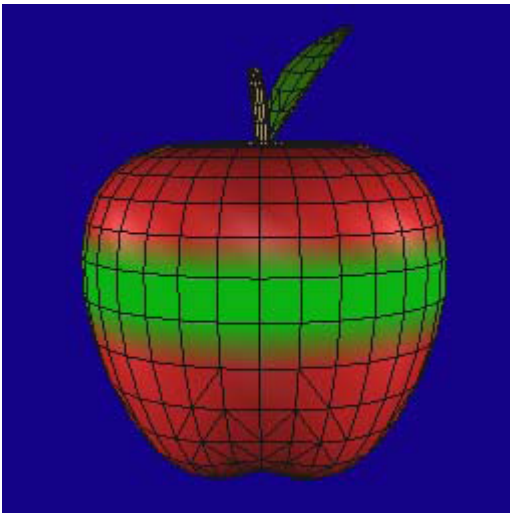
$$\text{Color.Green} = \text{Vertex.Green} * \text{Vertex.Alpha} + \text{Surface.Green} * (1.0 - \text{Vertex.Alpha})$$

$$\text{Color.Blue} = \text{Vertex.Blue} * \text{Vertex.Alpha} + \text{Surface.Blue} * (1.0 - \text{Vertex.Alpha})$$

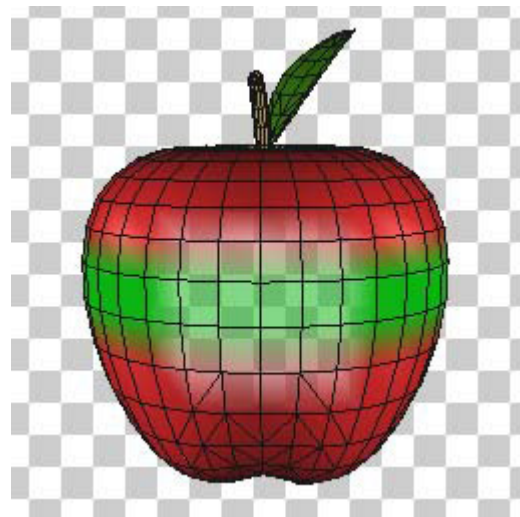
If you set the **Vertex Alpha** option to **Opacity**, information on the vertex color’s alpha is displayed as an opacity value. If the alpha value is 1.0, the vertex color would be completely opaque. If the alpha value is 0.0, then the vertex color would be completely transparent.



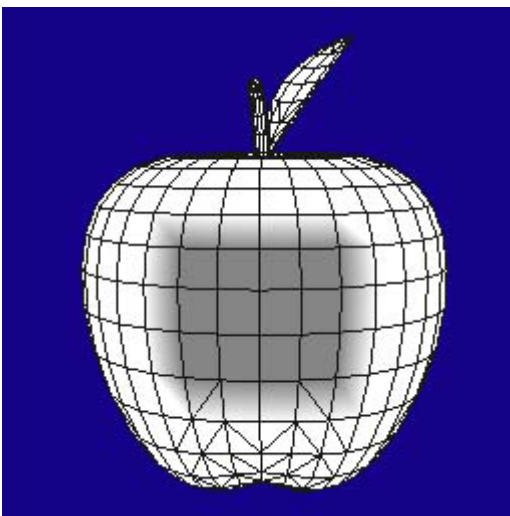
A blended display mode choosing **Surface Color** for the **Vertex Alpha** option. Both vertex color and surface color is displayed.



A component display mode using RGB painting. Only the vertex color is displayed.



A blended display mode choosing **Opacity** for the **Vertex Alpha** option. Vertices with an alpha value less than 1.0 are semi-transparent.



A component display mode using Alpha painting mode. Only the Alpha luminosity is displayed.



## Vertex Color Painting

Vertex Paint has three painting modes to paint the vertex color using a brush.

**Color / Paint:** Paint per point

**Color / Index:** Paint per vertex index of polygon

**Color / Polygon:** Paint per polygon

To begin painting, select a vertex color map on the **Color Map** pop-up menu or create a new vertex color map. Dragging your **LMB** on the 3D view paints points or polygons.

## Selecting Operation Mode

There are three operating modes that determine how paint is applied to existing vertex colors.

**ADD** — adds the value of the paint color to existing color.

**SUB** — subtracts the value of the paint color from existing color.

**REP** — replaces the value of existing color with the value of the paint color.

**ERA** — sets the value of existing color to 0.

## Selecting Component

You have two components for painting vertex color: RGB and Alpha.

To switch, click on the color component icon.



RGB



Alpha

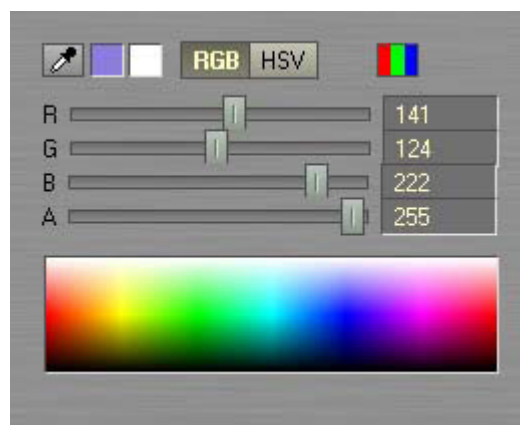
## Selecting Paint Color

There are two way to choose the paint color: the color slider bar and color palette.

The color slider bar has **RGB** and **HSV** modes. You can switch modes depending on your needs. The current RGB color is shown on the swatch to the right of the eye dropper icon. The current Alpha is shown on the swatch to the right of the color swatch. Although Vertex Paint internally maintains color using floating values (i.e., 0.0 to 1.0), numerical color values are rounded to the nearest integral value from 0 to 255.



You can choose a color directly by dragging your **LMB** over the palette, a color is selected when you release your mouse button. To choose an Alpha value, use the slider instead of the palette. You can customise the palette, which is discussed later.



You can also pick any color displayed on the 3D view. Hold the **Ctrl** and **Shift** keys, then click on the desired color in the 3D view. Another way to pick a color on the 3D view is to activate the eyedropper icon and drag on the 3D view with your **LMB**.

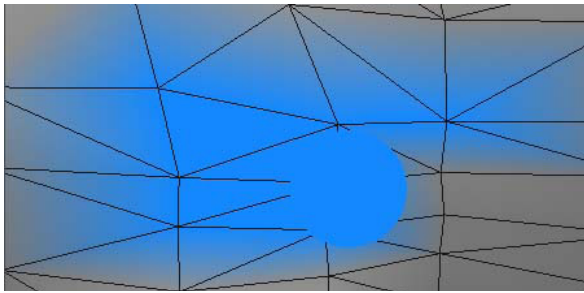


You can use the default LightWave color picker to pick a color as well. Clicking on the color swatch with your **LMB** will launch it.





## Painting Vertex Colors on Points



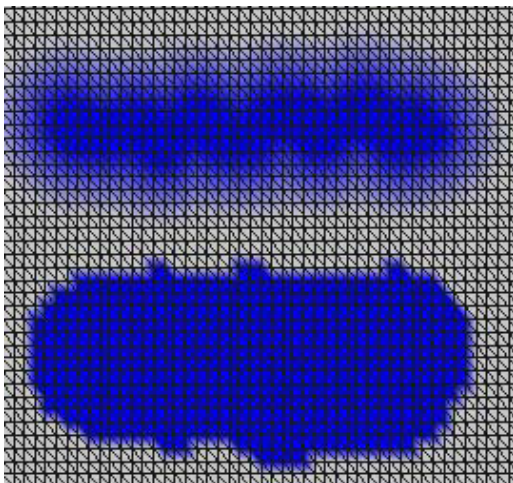
If you choose **Color / Point** for the Paint mode, you can paint vertex color on points, using three parameters: **Brush Size**, **Front**, and **Strength**.

**Brush Size** — Determines the size of your paint brush. You can visually see the brush size when you drag and paint on the 3D view with your **LMB**. To adjust your brush size, enter a value in the **Brush Size** input field. You can also adjust the size by dragging your **RMB** in the 3D view. Release your mouse button to accept the new size.

**Front** — Use this option when you do not want to paint points on the opposite side.

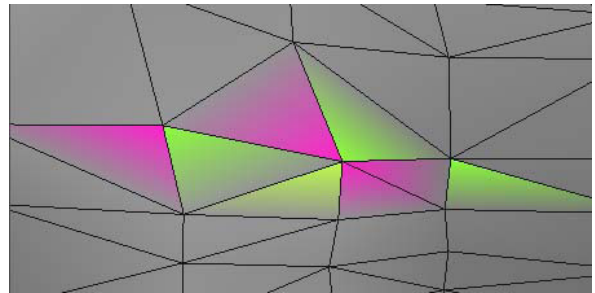
**Brush Shape** — You can paint with a round or square brush by clicking on the icon to the right of the **Brush Size** slider.

**Strength** — Strength allows you to paint as if you were using a soft brush. If **Strength** is set to 100%, all painted vertices get the same value. If **Strength** is set to 0%, the effect of the brush is 100% at its very center, but 0% at its edge. Areas of the brush between the center and edge use a proportionate value.



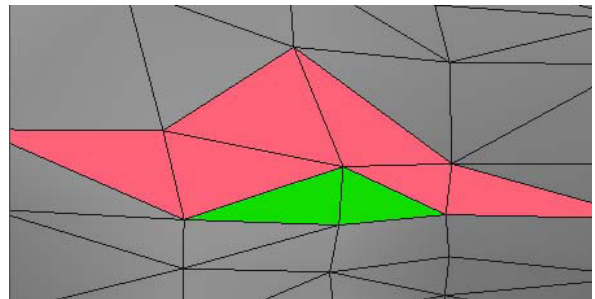
Top: Strength=25%, Bottom:Strength=100%

## Painting Polygon Vertex Indices



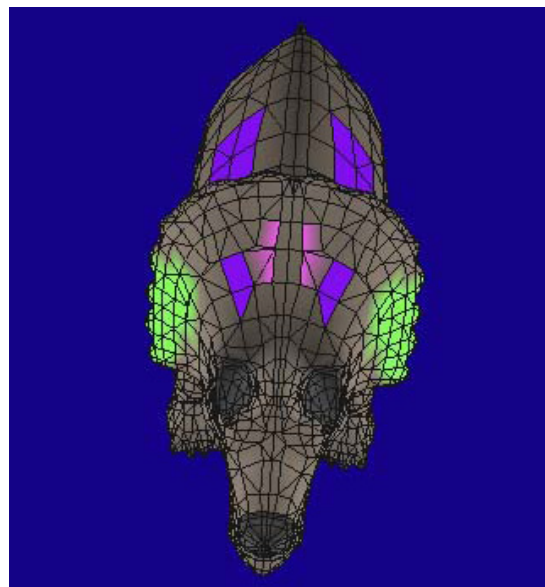
If you choose **Color / Index** for the Paint mode, you can paint the “vertex index” of polygons. The vertex index is a unique reference for each polygon vertex. Although a point may be shared by polygons, the vertex index is different. Just click inside the polygon nearest the vertex you want to affect. **Brush Size**, **Front**, and **Strength** settings are ignored.

## Painting Vertex Color per Polygon



If you choose **Color / Polygon** as the Paint mode, you can paint individual polygons a solid color. **Brush Size**, **Front** and **Strength** settings are ignored. This essentially sets all of the polygon's vertex indices to the same value.

## Painting Symmetrically along the X Axis



Clicking the symmetry button at the top of the interface lets you to paint symmetrically along the X Axis. If you paint geometry on one side (left or right), all the polygons and points symmetrical along the X axis is painted at the same time. This option can be used with any paint mode.

Symmetry



## Customising the Color Palette

You can customise your color palette for Vertex Paint by choosing **Edit > Customize Color Palette**. There are four palettes available: 16, 64, 256, and Gradient, the default. Activating the **Clear Colors** option erases all existing palette information, resetting the colors to black (0,0,0).



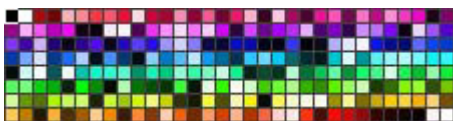
Customised palette information is stored in the preference file. It maintains the information until another palette is loaded or the palette is reset.



Palette Size = 16



Palette Size = 64



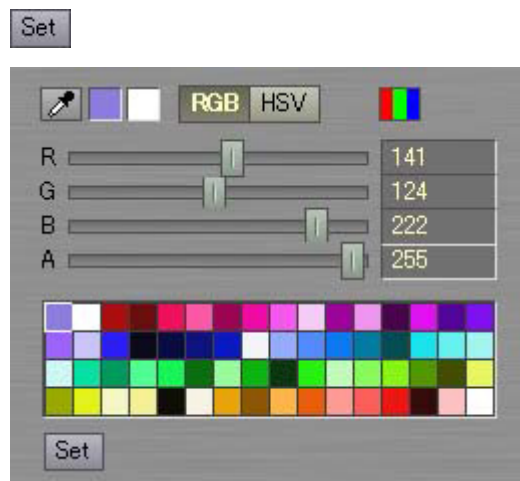
Palette Size = 256



Gradient

To add a color to a cel in a (non-gradient) palette, first select a

cel with your mouse. Then, adjust the current RGB color values. Finally, click the set button below the palette. The RGB color will be added to the selected palette cel.



## Loading a Color Palette

To load a color palette, with the Color tab selected, choose **Edit > Load Color Palette File**. Then select a previously saved palette file. The following four formats are supported: 8-bit BMP, 8-bit TARGA, Microsoft Palette® (.pal), and ACT color table (.act).

## Saving a Color Palette

To save a color palette, with the Color tab selected, choose **Edit > Save Color Palette File**. The file is saved in Microsoft Palette® format (.pal).



## Baking OpenGL Shading

With Vertex Paint, you can “bake” OpenGL color information (including light source calculation) to a vertex color map. You can see the shading status to be baked using the **Lighting Shade** display mode. Adjust the light strength, location and other settings on the Light tab before baking.

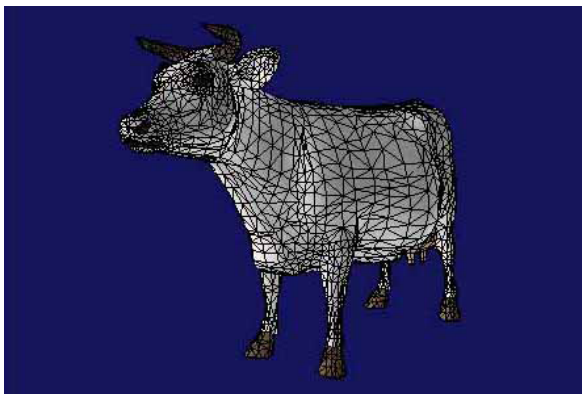
To bake the colors, first create a vertex color map, then with the Color tab selected, choose **Edit > Burn Shading Colors**.



NOTE: You have to set your view so that the entire object is visible in the 3D view. If not, the part outside the view will be omitted from the color calculations, or an error will occur.



The Lighting Shade display mode. The light source is calculated.



The Wireframe Color display mode, displaying vertex color that was baked by the **Burn Shading Colors** command.

## Reset to Surface Color

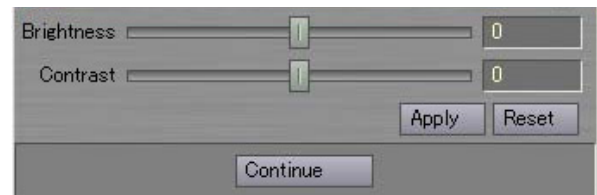
Executing “**Reset To Surface Colors**” from the “Edit” menu resets vertex color to surface color.

## Adjusting Vertex

Vertex Paint has several tools for adjusting vertex colors baked by **Burn Shading Colors** or the **Shader Baker** layout plugin.

## Adjusting Brightness and Contrast

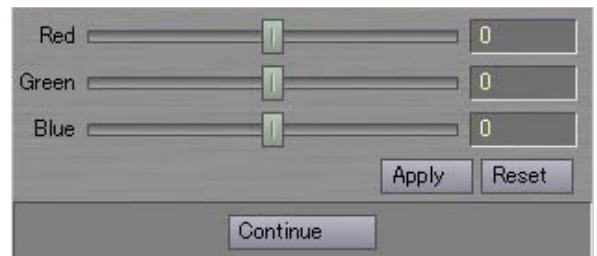
With the Color tab selected, choosing **Edit > Brightness/Contrast** displays the following dialog box, where you can adjust brightness and contrast.



When you move the sliders, the change in brightness or contrast is previewed on the display in realtime. However, the change is not saved until you click the **Apply** button.

## Adjusting Balance

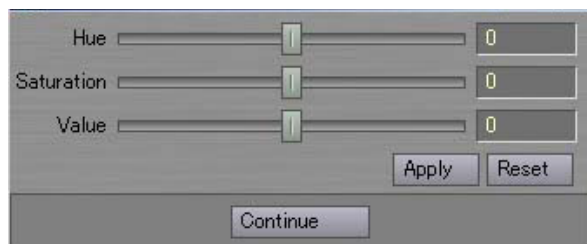
With the Color tab selected, choosing **Edit > Color Balance** displays the following dialog box, where you can adjust each RGB component separately.



## Adjusting Hue, Saturation and Value



With the Color tab selected, choosing **Edit > Hue/Saturation** displays the following dialog box, where you can adjust hue, saturation and value.



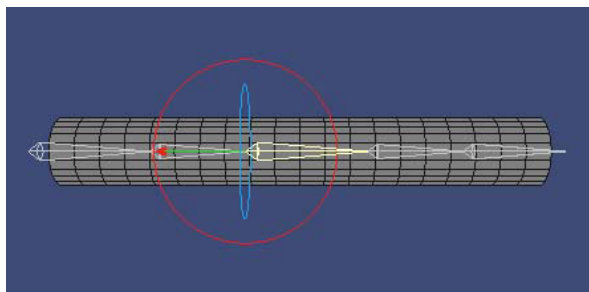
## Vertex Paint: Bone Weight Paint

### Choosing Bones

There are three ways to select the current bone, which becomes highlighted in yellow:

1. With the Weight tab selected, select the bone name from the **Bone** pop-up menu.
2. Press the **Up** or **Down cursor key** to choose the next or previous bone until you get the desired one.
3. Hold the **Shift** key and click the desired bone with your **LMB** directly in the 3D view.

It may be difficult to choose a bone in the 3D view when it is inside the mesh. In such cases, disable **Hide Unselected Bones** on the Preferences panel. This shows the unselected bones outside the object so they are easily selected.

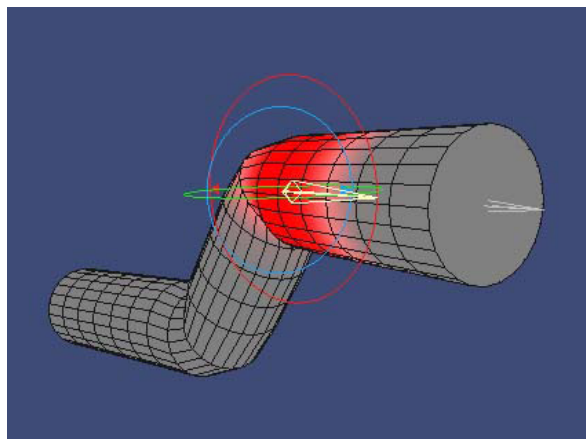


Disabling **"Hide Unselected Bones"** shows the bones

Also, you can work with bones easier when they are centered in the 3D view. To do this, just press the **A** key or select **Fit Bone** from the pop-up menu (small button with down arrowhead) along the top of the main interface.

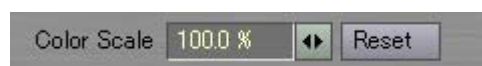
### Displaying Weight

There are two display modes for bone weight: Weight Value and Weight Ratio.

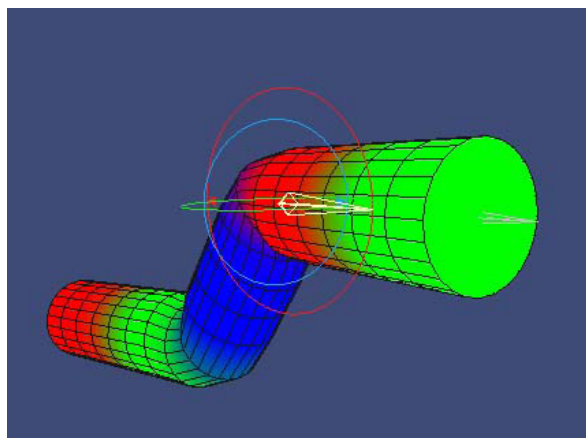


Weight Value display mode

**Weight Value** displays the value of the weight map that belongs to the current bone using a red gradient. Areas displayed in grey have no weight, or are just slightly weighted. This mode is useful when you are painting the weight. You can adjust the density of color using the **Color Scale** option on the Paint tab.

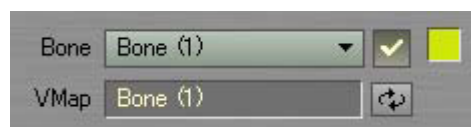


You can also set the default value for **Color Scale** with the **Weight Color Scale** setting on the Preference panel.



Weight Ratio display mode

**Weight Ratio** is a mode that displays the ratio of bones' influence on each point. Red, blue and green are preset for each bone; however, you can reset the colors using the color icon next to the **Bone** name pop-up menu.



"Bone Color" icon





## Renaming Bones

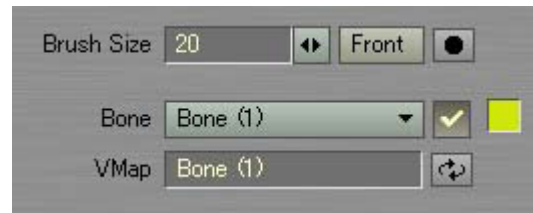
To rename bones, choose **Edit > Rename Bone** with the Weight tab selected. A bone tree, like the one shown below appears. Select the bone you want to rename with your LMB and enter a name in the **New Bone Name** field. You can also rename the weight map name (**New VMap Name** field) at the same time.

The bones are displayed in a hierarchical order. Bone visible in the 3D view will be updated in real time. Also, the bone selected here becomes the current bone.

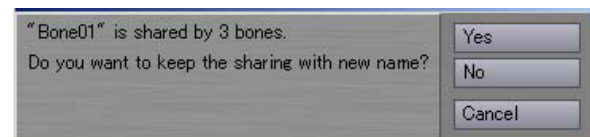


## Renaming Weight Maps

To rename a weight map, enter new name in **VMap** field. Don't forget to save your changes.



You might see the following dialog box when you try to rename a VMap. This box appears when one weight VMap is already used (shared) by one or more other bones. If you click **Yes**, all of the VMap names used for the bones that refer to this VMap will be renamed. If you click **No**, only the VMap which the current bone refers to will be renamed - if no such VMap already exists, a new one will be created.



This panel shows that three bones share one name "Bone01." So if you click **Yes**, all three bones that referred to VMap "Bone01" will refer to newly renamed VMaps.

## Copying Weight

You can copy a value of the weight map that belongs to the current bone to the Vertex Paint clipboard buffer. Press the **C** key or choose **Edit > Copy**. No message will be shown.

## Pasting Weight

You can paste the copied weight value from the clipboard buffer to another bone's weight map. Press the **V** key or choose **Edit > Paste**.

## Pasting Weight Symmetrically in X Axis

You can paste the copied weight value from the clipboard buffer to another bone's weight map along the X axis symmetrically. Press the **V** key or choose **Edit > Paste-X**. The point located symmetrical along the X axis is found and weight map value is set.

You can specify the amount of "calculation error" when finding the symmetrical point with the **Mirror Epsilon** setting on the Preference panel.



## Pasting Weight Symmetrically Along an Arbitrary Axis

You can paste the copied weight value from the clipboard buffer to another bone's weight map symmetrically along an arbitrary axis. Choose **Edit > Paste-Axis** to display the panel shown below. Select the desired **Axis** and offset **Position**. Then click **OK**.



### Axis

Specifies the axis which the weight is to be pasted symmetrically. If you choose **X**, all of the points that meet the condition " $X' = -X$ " is looked up.

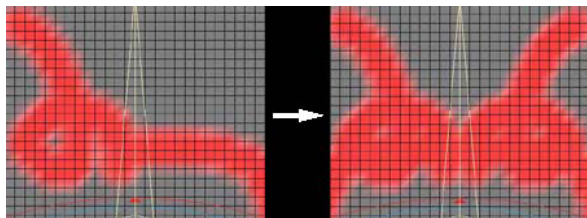
### Position

Specifies the origin position. If you selected the **X Axis** and then set **Position** to 1m, all of the points that meet the condition " $X' = -X + 2m$ " is looked up.

### Operation

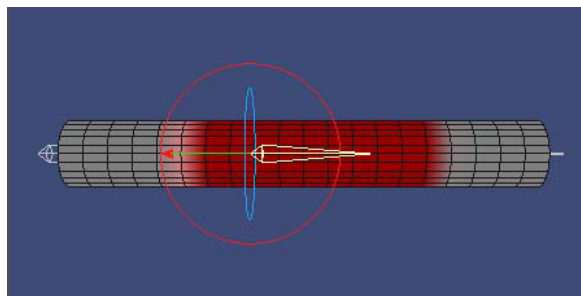
Specifies how the weight will be pasted to the points. **REP** replaces the whole original weight value with the new value. **OVR** overwrites part of the original value only if the new value is bigger than the original ones.

Using the **OVR** option allows you to paste the map symmetrically onto the same weight VMap. To do this, copy the weight and then paste it to the same weight VMap using the **OVR** option, you can create a weight map symmetrical to itself.

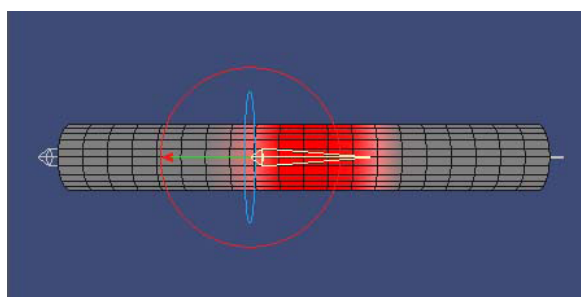


## Normalising Weights

Sometimes you may get excessive weight values from using the **BoneWeight** Modeler command or just from painting in Vertex Paint. Weight values go beyond the normal range 0.0 to 1.0. Bone deformation can deform without "normalising" the weight because it is performed relatively, that is, using the percentage of the weight. However, in some cases (such as adjusting the weight by painting), you may want to normalise the weight to give you better control. Choose **Edit > Normalize** to normalise the weight values between 0.0 and 1.0, according to their percentages.



A weight map that is not normalised



A weight map that is normalised

## Trimming Weight

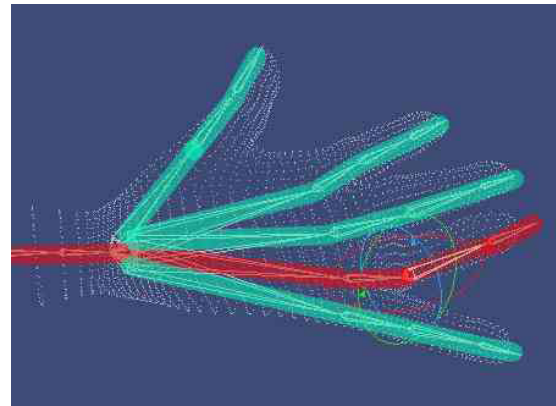


Bones inside “branching-type” geometry, like fingers, can be problematic when the weight map is created using a falloff calculation without any correction. Bone deformation will likely affect areas you wish to be independent. This is because the falloff calculation is performed referring to only the distance between a bone and a point.

The **Edit > Trim Branch** function solves the problem by allowing you to trim the weight for the current weight map. It first calculates the “primary bone” for each point (i.e., the bone closest to the point). If the bone is not a descendant of the current bone, the weight map value that the point holds is cleared. “A descendant” is a “bone route path” generated from one parent (see Fig.3).



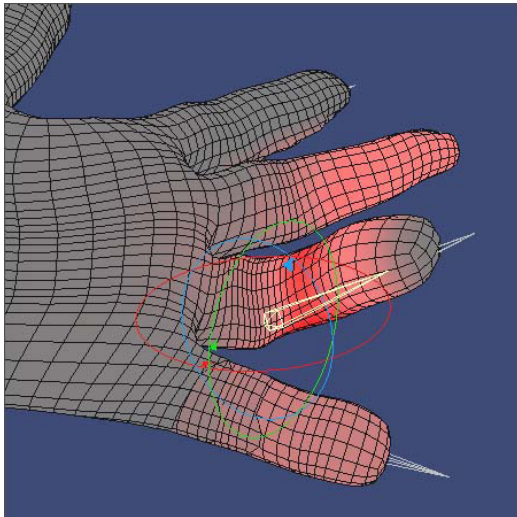
**NOTE:** If you use this command where the bone route path branches, like at the base of a finger, it may not deform smoothly. As such, try to use this command where there is not bone branching. Also, trimming may not work well depending on the bone location. If you encounter this problem, try using the **ERA** or **SUB** painting operations to adjust the weight value.



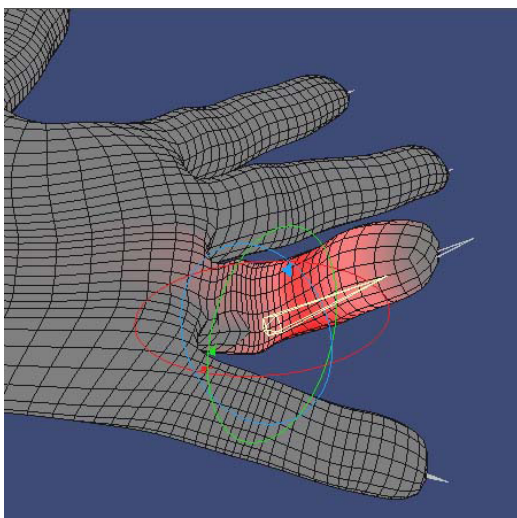
Bones on the red line represent a descendant of the current bone, and bones on the blue line represent the other descendants.

### Painting Weight

Select the Paint subtab on the Weight tab to “paint weight.” There



Before using **Trim Branch**. If you bend the finger using the highlighted bone, neighbouring fingers will also bend because they have bone weight.

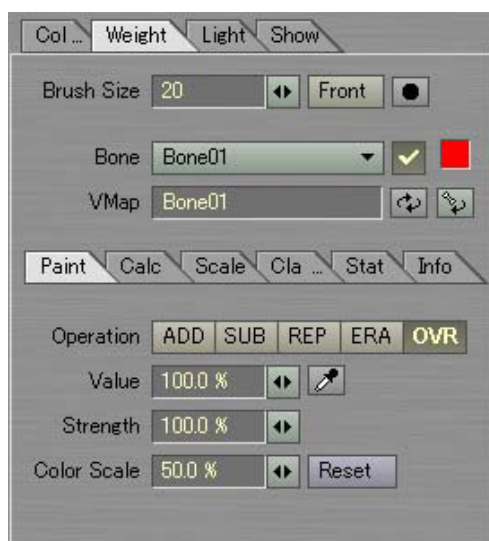


After using **Trim Branch**. Unnecessary weight for the points on neighbouring fingers have been erased. You can now bend the finger independently.





are some adjustable parameters that will determine how the effect is applied. Dragging in the 3D view with your LMB paints the points under your mouse pointer.



The **Brush Size** and the **Front** settings are shared with the the vertex color painting operations.

## Operation

Specify how the point weight is applied:

**ADD** — This mode adds the **Value** to the painted target weight. Note that the operation is only applied once while you press the mouse button, start dragging, and then release the button.

**SUB** — This mode subtracts the **Value** from the painted target weight. Note that the operation is only applied once while you press the mouse button, start dragging, and then release the button.

**REP** — This mode replaces the painted target weight with the **Value**.

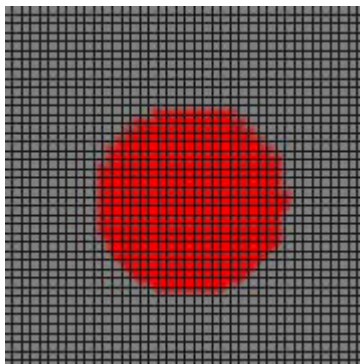
**ERA** — This mode sets the painted target weight to 0.0.

**OVR** — This mode replaces the weight only if the **Value** setting is larger than the painted target weight value.

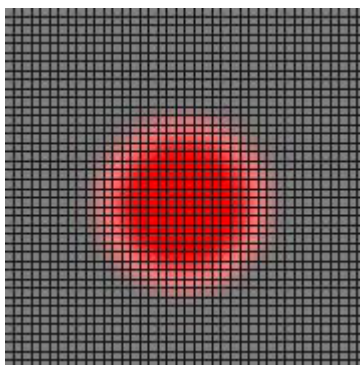
**Value** — This specifies the weight value that you are going to paint. A value of 100% is internally converted to 1.0 as a weight value. When you activate the picker button, you can refer to the weight value in the 3D view. Dragging your **LMB** refers to a weight value of the point that is closest to your mouse pointer.

## Strength

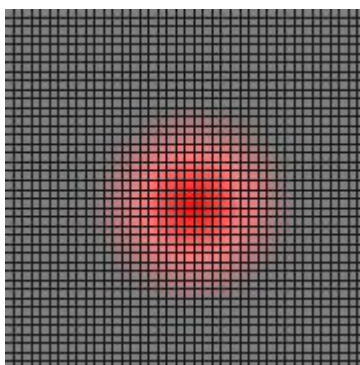
**Strength** sets the point from the center of the brush where the paint falls off. If **Strength** is 100%, points inside the brush are all painted with a the full **Value**. If **Strength** is 0%, the center of the brush is painted with 100% of strength, and the value decreases relatively from from the center to the outer edge of the brush. If **Strength** is 50%, points within half the radius receive the full **Value**, then it falls off toward the edge of the brush.



**Strength** = 100%. All the point inside the brush is painted with the same value.



**Strength** = 50%. Points inside half the radius are painted with 100% of the Value, then decreases relative to the distance from the center.



**Strength** = 0%. The paint strength decreases relative to the distance from the center of the brush.

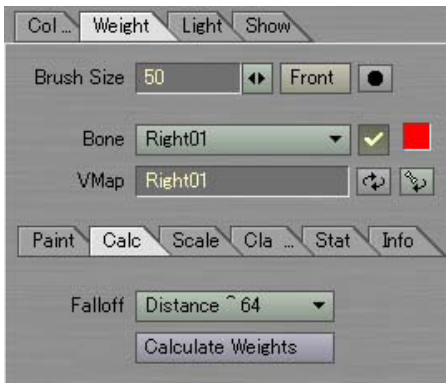


## Color Scale

This adjusts the strength of red gradient color in the **Weight Value** display mode. This is only a display setting, so changing this value does not affect the actual weight value.

## Calculating Weight

To calculate weight using a falloff, select the Calc subtab on the Weight tab. Basically, this function works the same as the Modeler plugin Bone Weight. Note that any weight calculations will be normalised.

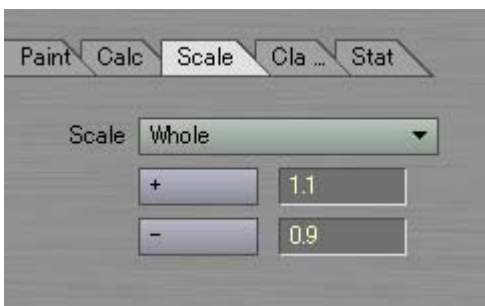


**Falloff** specifies how the weight value is calculated. **Linear** sets the bone weight closest to the point to 1.0. Other weight is set to 0.0. **Distance** uses the inverse of the distance from point to bone as weight value. Other options (e.g., Distance ^ 32) use the distance raised to a specific power. Higher powers cause a quicker falloff.

**Calculate Weights** executes the calculation based on the settings you have chosen.

## Scaling Weight

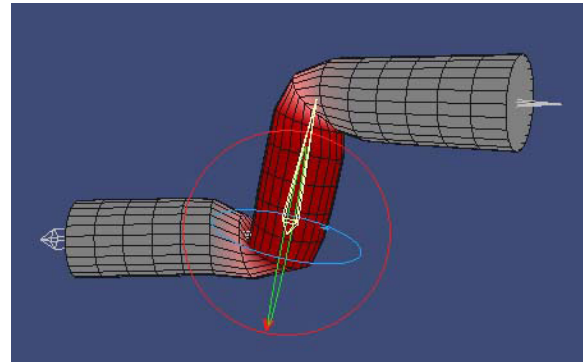
To scale weight values, select the Scale subtab on the Weight tab.



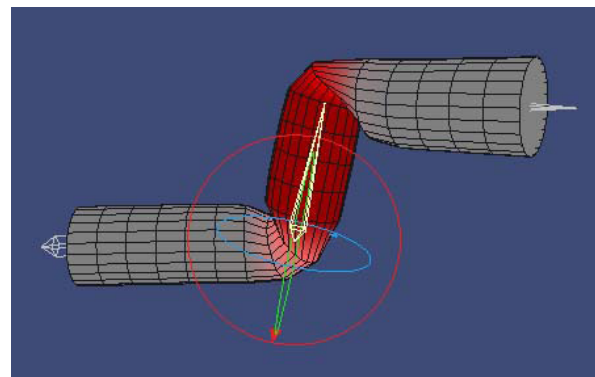
The **Scale** setting specifies how the weight values are scaled.

<b>Whole</b>	Scales all weight value uniformly.
<b>At Start Joint</b>	Scales values using the bone's starting point (the fat end) as the center.
<b>At End Joint</b>	Scales values using the bone's ending point (the sharp end) as the center.

Click the **+** button to scale up the weight values using the value set in the input field next to the button. You can also scale it up by pressing the **“+”** key. Click the **-** button to scale down the weight value using the value set in the input field next to the button. You can also scale it down by pressing the **“-”** key.



A weight map scaled up from the bone's starting point



A weight map scaled up from the bone's ending point



## Clamping Weight

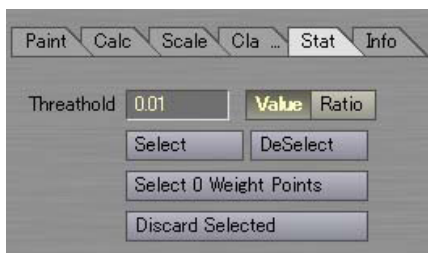
To clamp the weight, select the Clamp subtab on the Weight tab. When you deform single-mesh objects using bones, the more bones you have, the slower the calculation speed. By discarding the weight value for bones that have little influence, the calculation speed will be improved. This can be one of the most important factors in single-mesh realtime animations.



The **N of Matrix** setting corresponds to the number of bones that are sorted by their strength of influence. Let's say you have a mesh with twenty bones. If you set **N of Matrix** to 5 and then press **Clamp**, up to the top five strongest weight values of the bones (for each bone) are retained. The other values (that is, values of the bones with weak influence) are set to 0.0.

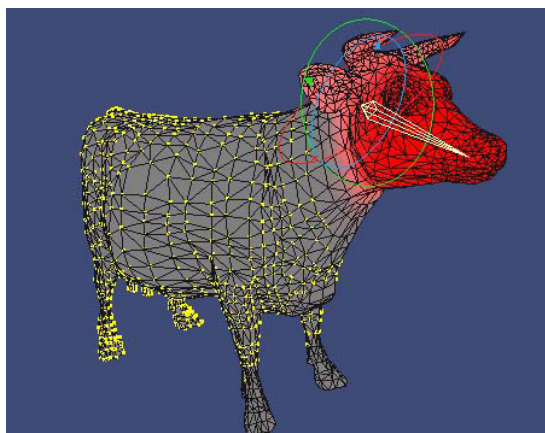
## Viewing Weight Statistics

Use the Weight Statistics tab to select weighted points based on a value. These can be removed from the map to increase efficiency. To view weight statistics, select the Stat subtab on the Weight tab.



The **Threshold** setting specifies the threshold for selecting points. The buttons to the right of the input field represent the units. **Value** means the actual weight value. **Ratio** means a normalised range from 0.0 to 1.0.

Clicking the **Select** button will highlight in yellow the points which have a weight value less than the **Threshold**. To deselect, click the **DeSelect** button.

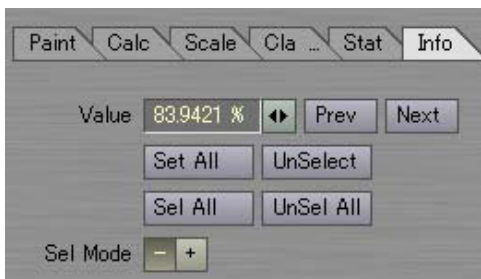


The **Discard Selected** button removes the map from the selected points.

The **Select 0 Weight Points** button detects zero-weighted points.

## Setting Weight Value

To set the weight value for individual points, select the Info subtab on the Weight tab.



## Selecting/Deselecting Points

By default, no points are selected. There are several ways to select points to edit.

### To select all points

Click the **Sel All** button to select all points. Press the **UnSel All** button to deselect all points.

### To use your mouse pointer

Drag your **LMB** in the 3D view to select points.

### To select using a bounding box

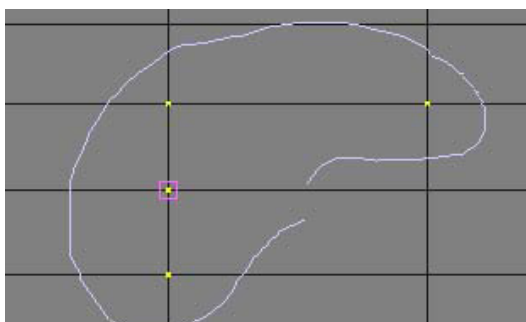
Hold the **Ctrl** and **Shift** keys and drag a bounding box with your **LMB** in the 3D view.

Points included in this bounding box are selected when you release your mouse button.



### To select using lasso

Hold the **Ctrl** and **Shift** keys and drag your **RMB** in the 3D view. Points within the lassoed area are selected when you release your mouse button.



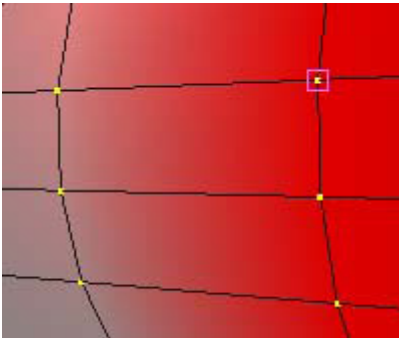
You can use the **Sel Mode** setting to switch between Select (+) and Deselect (-) modes. You can also switch modes by hitting the Space key.

Clicking the **UnSelect** button deselects the current editable point (pink square).

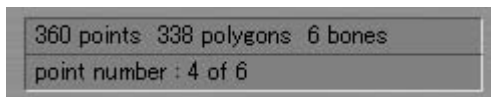
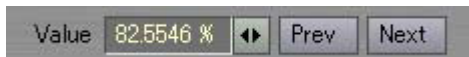


## The Current Editable Point

Selected points are shown as a yellow point, and the current editable point is surrounded by a small pink square.



The value of the current editable point is displayed in the **Value** field. You can change the weight value by dragging the slider or entering a value directly. You can cycling through all of the selected points, to change the current editable point, by clicking the **Prev** and **Next** buttons. The number of points selected and the (reference) number for the current editable point is displayed in the information display field at the bottom.



## Setting Weight Value

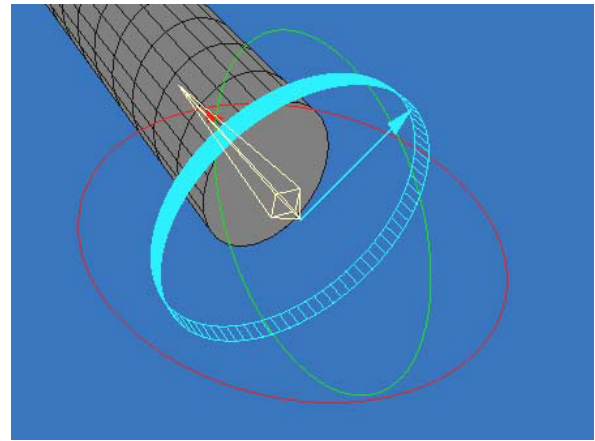
To specify the weight value for all selected points, enter a number in the **Value** field and click the **Set All** button.

## Editing the Bank Rotation Axis of Skelegons

Skelegons generated in Modeler have vector information for the *bank rotation* axis that refers to the upward direction. This determines the bank rotation angle of the bone when converted in Layout. To adjust the bank rotation axis in Vertex Paint, activate the Bank Rotation Axis Edit option. (Note: Vertex Paint will reset all skelegons to their rest position before allowing you to edit the bank rotation.)



Hold the Ctrl key and drag your **LMB** in the 3D view. A thick blue ring represents the bank rotation. The arrow represents the "upward direction" of the skelegon.



This editing option is only available for Modeler's skelegons. You cannot use this option on the bones loaded by Vertex Loader.

If you want the bank rotation axis to align to a specified axis precisely, press the **N** key. The bank rotation axis will correspond to the axis you chose.



## Vertex Paint: Creating a Morph Target

To create a new morph target, choose (**new**) from the **Morph** pop-up menu on the main panel.

A dialog will appear where you indicate the **Name** and **Type** of morph target.



If you enter a new name when creating a morph target, geometry with bone deformation (from rotation, move, and scale) in the 3D view is created. Deformed geometry created using the preview animation function can also be used.

## Vertex Paint: Misc





## Import/Export Vertex Paint 1.0 file

The **Import...** and **Export...** items on the **File** menu can be used to import/export the Vertex Paint 1.0 format of the vertex color file (.vcl). (Vertex Paint 1.0 came with LW 5.6)

## Loading a Vertex Color file (.vcl) into Vertex Paint

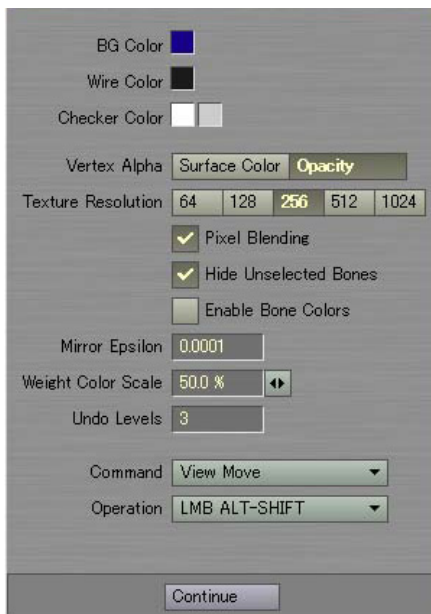
You must first load the LightWave 5.6 format LWO file into Modeler before loading the vertex color file (.vcl). Then start Vertex Paint. Create a new vertex color map by choosing **Edit > Create Vertex Color Map** and then choose **File > Import...** Select a .vcl file associated with the loaded lwo file on the file dialog.

## Saving the Current Vertex Color Map to a .vcl File

Select the vertex color map you want to save and choose **File > Export....** Enter a name in the file dialog and click **Save**.

## Preference Panel

You can define some default setting on the Preferences panel (**File > Preferences**). Note that some changes will not take effect until the next time you launch Vertex Paint.



## BG Color

**BG Color** specifies the background color. Clicking on the color box with your **LMB** displays the standard LightWave color picker. The change is applied when you close the Preferences panel. (Note: If your display does not update, try clicking in the 3D window.)

## Wire Color

**Wire Color** specifies the wireframe line color. Clicking on the color box with your **LMB** displays the standard LightWave color picker. This setting affects the Wireframe Color, Weight Value, and Weight Ratio display modes. The change is applied when you close the Preferences panel. (Note: If your display does not update, try clicking in the 3D window.)

## Checker Color

The two **Checker Color** boxes specifies the colors used for the checker pattern that makes up the background in the blending display mode. Clicking either of the color boxes displays the standard LightWave color picker.

## Vertex Alpha

**Vertex Alpha** determines how the vertex color's alpha information is displayed in the Vertex Color and Wireframe Color display modes. If you select **Surface Color**, information on the vertex color's alpha is treated as a value of blended percentage of vertex color and surface color. If you select **Opacity**, information on the vertex color's alpha is displayed as a value of opacity.

## Texture Resolution

**Texture Resolution** determines the texture resolution displayed in the Vertex Color, Wireframe Color and Lighting Shade display modes.

## Pixel Blending

If **Pixel Blending** is active, both texture color and vertex color (or surface color) are displayed together. If **Pixel Blending** is inactive, texture color overrides the vertex color and Vertex Paint only displays texture color.

## Hide Unselected Bones

**Hide Unselected Bones** specifies the display mode for unselected bones. If this option is active, unselected bones hide behind the objects. If this option is inactive, unselected bones are shown in front of the object so that you can grab and control them more easily.

## Enable Bone Colors

If **Enable Bone Colors** is active, bone geometry is displayed using the Weight Ratio color.

## Mirror Epsilon

**Mirror Epsilon** determines the accuracy used when Vertex Paint searches the axis-symmetry point for the **Symmetry** mode, and the **Paste-X** and **Paste-Axis** options. The smaller value you set, the more precise the symmetry point are searched.

## Weight Color Scale

**Weight Color Scale** determines the default value of the color scale used for the Weight Value display mode.

## Undo Levels

**Undo Levels** specifies the maximum number of undos in the Color and the Weight display modes. Higher values require more memory.



## Command / Operation

Select desired setting from those pop-up menu when you want to customise the default mouse operations. Assign each **Operation** setting to each **Command** setting.

## Frequently Asked Questions

### Q. Does Vertex Paint work with SubPatch objects?

**A.** Yes. A SubPatch mesh is displayed as a polygon mesh. Vertex Paint can edit the required VMAP information. The original SubPatch information is maintained after it is modified.

### Q. I cannot paint vertex colors.

**A.** One or more vertex color maps are required to paint vertex colors. To create one, with the Color tab selected, choose **Edit > Create Vertex Color Map**.

### Q. An error “Unable to burn render colors.” occurs when I try to bake vertex colors.

**A.** The Burn Shading Colors command uses OpenGL functions to acquire the color that is currently displayed. Therefore, Vertex Paint will not work properly if there is anything covering the 3D view or the object is outside the 3D view. Put the object inside the view and try again.

### Q. I created a vertex color map but the vertex color is not displayed in Modeler.

**A.** You have to assign the vertex color map to a surface. To do this, specify the map for the **Vertex Color Map** setting on the Surface Editor's Advanced tab.

### Q. Does Vertex Paint work with Layout in real time via the HUB?

**A.** Clicking Vertex Paint's **Save** button overwrites modified weight value and/or vertex color in Modeler. The changes should be reflected on the object in Layout after saving. (Note: Conversely, Vertex Paint does *not* acquire surface properties changed in Layout.)

### Q. Can I utilise information such as vertex color for game development?

**A.** You can access information, such as vertex color, by using the LightWave 8.0 plugin SDK. Other information, such as UV map and bone weight, can be acquired from both Modeler and Layout.

You also can download *Object Library 2* from DStorm to convert vertex color information stored in a LWO file. This could be used for something like in-house tool development.

## Texture

### What is UV Mapping?

Sometimes, when mapping textures onto objects, you will find that the normal projection mapping just doesn't work. This usually happens when the object is organic, or irregular in shape. For those occasions, there's **UV Mapping**!

**UV Mapping** adds two extra coordinates to the points in your object; those on the U and V axis, running horizontally and vertically through a flat plane on which you can paint your texture. Since the coordinates are assigned to points, it's essentially as if that painted texture was fixed to the surface of the object, with pins where all the points are. No matter how irregular your object is, or how it moves or flexes, those pins stay in place, and the texture stays right where you put it. Technically, it's not as accurate as projection mapping, because the texture is really only exact at those points, and merely interpolated everywhere else. But it's close enough!

The trick is to make the map first. LightWave has many tools to help you do that.

### What is Per-polygon Mapping?

**Per-polygon UV Mapping**, or **Poly Mapping**, was introduced in LightWave 6.0b as a way to handle discontinuous **UVs**. (Without some kind of handling, they would cause the entire image map to appear backwards and squashed onto a single polygon at the map seam. Not a pretty sight.) They were superseded by the current **UV Mapping** tools in LightWave 6.5, so unless you are working with a model that was mapped before that, there's really no reason to use them; the new tools are much more elegant and flexible.

Unlike **UV Maps**, **Poly Maps** cannot be edited with the normal **Modify** tools used in the rest of Modeler, because they are not continuous. Instead, they are locked snapshots of some **UV Mapping** of a polygon. So any operation on a vertex must also specify which polygon's view of that vertex is being edited.

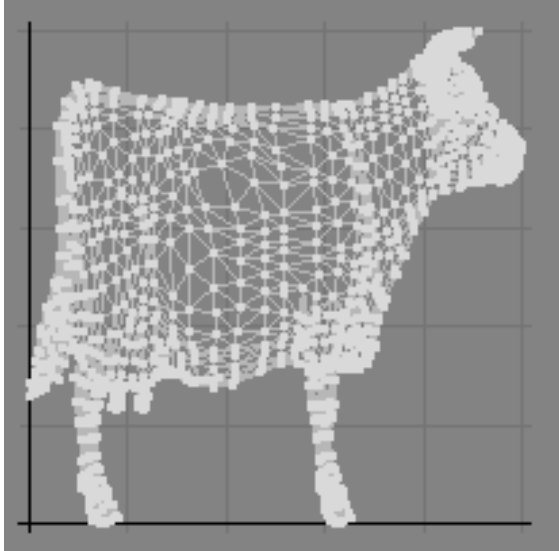
You can select polygons that use per-poly UVs by choosing the **Select by Polymap** command (**View > Selection: Maps > Select by Polygon Map**).



## Map Types

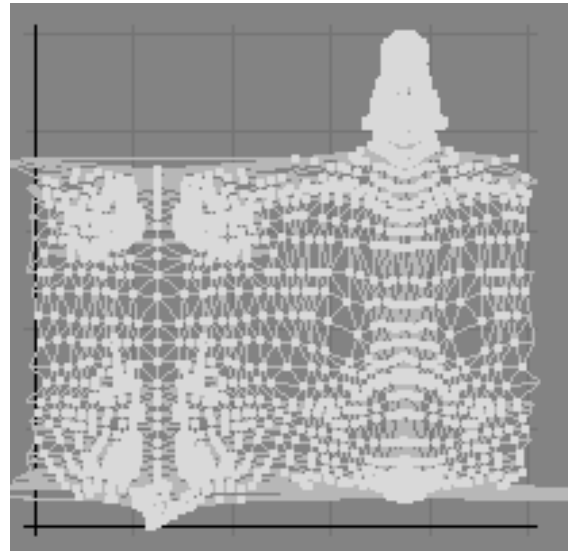
UV maps can be of several different types explained here:

Then choose the type of projection you wish to use from the **Map Type** list. They correspond to the standard projections you may already be familiar with.



### Planar

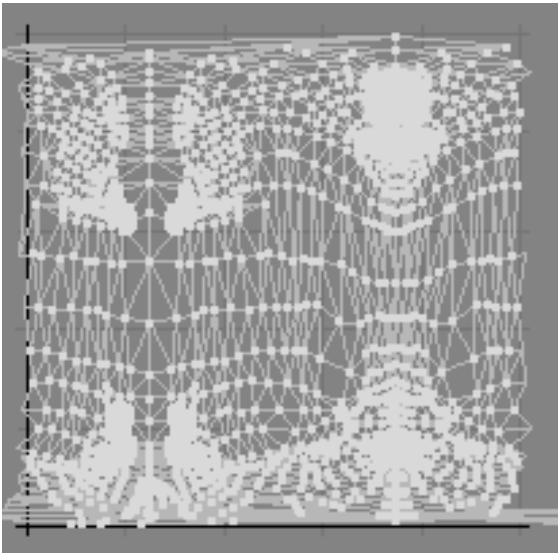
The first, **Planar**, simply looks at the object from one of the three axes (X, Y, or Z.) What you'll get looks just like what you see in one of the **Orthogonal** viewports, but resized to fill the square of the **UV** space. If you choose X, you'll get what you see in the **Right** or **Left** viewport. If you choose Y, you get what you would see in the **Top** viewport. If you choose Z, you'll get what you see in the **Front** or **Back** viewport.



### Cylindrical

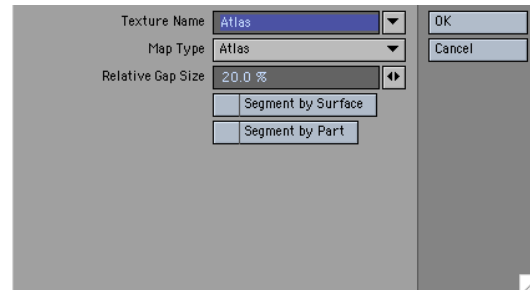
The second, **Cylindrical**, gives you the same effect as you would get from a camera that rotated around the object, (sort of like having your teeth x-rayed) using the axis you chose as the center of the rotation. So, if you choose X, the camera will rotate around a pole going from left to right, and the top of the map will be the right side of the object. If you choose Y, the camera will rotate around an upright pole, and the top of the map will be the top of the object. If you choose Z, the camera will rotate around a pole going from close to far, and the top of the map will be the back of the object.





### Spherical

The third, **Spherical**, behaves as if all the points are fastened to the inside of a sphere, and then the sphere is laid out flat. The poles of the sphere are on the axis you choose. So, if you pick X, the poles will be at the right and left, and the top of the map will be a few polys on the right side of your object. If you choose Y, the poles will be at the top and bottom, and the top of the map will be a few polys at the top of your object. If you choose Z, the poles will be close and far, and the top of the map will be a few polys on the back of your object.



### Atlas

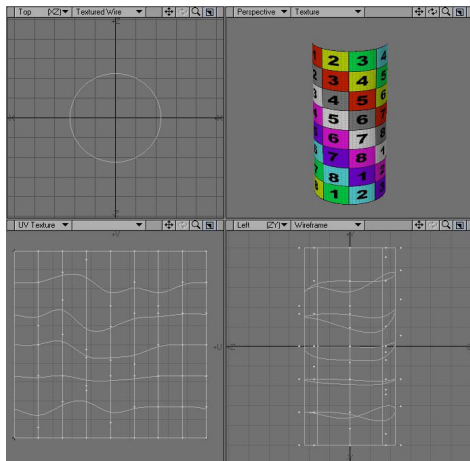
The fourth **Mapping** method, **Atlas**, will call a slightly different dialog, because it has different tools. **Atlas** spreads the UVs out, so that none of the polygons overlap, and there is a minimum of distortion. This is good, if there's no shading from poly to poly. It's not so good if you need to shade smoothly over a number of polys, because it can be very difficult to paint on. You'll have to decide according to your needs.

### Relative Gap Size

But, because the polys become separated in **Atlas** mapping, instead of choosing an axis (which would be meaningless), you can choose how much space should be left between groups of polys using **Relative Gap Size**.

### Segment by Surface, Segment by Part

This option allows you to choose whether the polys should be separated by **Surface** or **Part**. (**Segment by Surface**, **Segment by Part**.) It's recommended that you enable these, if you have a number of polys in a surface or part group that will be getting the same color, because it can make it easier to paint.

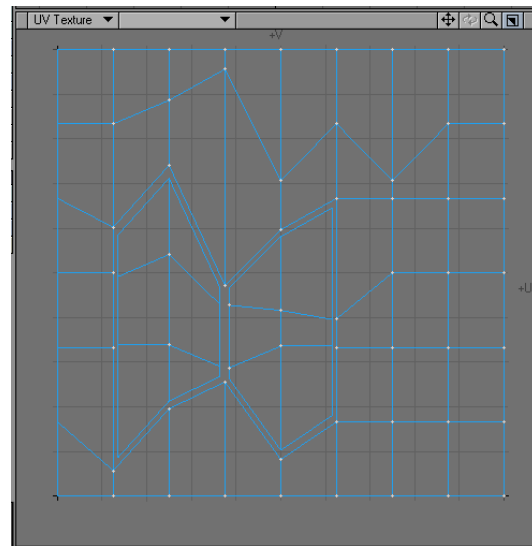


However, there can be problem areas on a UV map that require special treatment. UV maps can have edges that are open in a map but closed on the mesh. A single edge on a mesh can be two edges on a map, or a discontinuous edge. These edges are usually along the UV unwrap seams. This usually depends on the UV unwrapping method and the UV editing done, which means that each UV map is unique to every mesh. Discontinuous edges usually surround a so-called UV island. A single UV map can contain multiple UV islands. These UV islands can be rotated and positioned freely in the UV space (by you, or using unwrapping methods like Atlas). A single island or even a single polygon can also be part of multiple UV maps (with different positions and orientations in each UV map).

Usually Subpatch-interpolation behaves differently on the outer edge of a mesh than inside a mesh. Since discontinuous edges are outer edges relative to a UV map or UV map island, but are inside edges relative to the mesh, it is not always possible for LightWave to automatically perform a correct Subpatch-interpolation over discontinuous edges or over edges that are shared by multiple UV maps. For these types of situations, the following interpolation types have been offered to get the least distortion out of a UV map.

### Interpolation Type

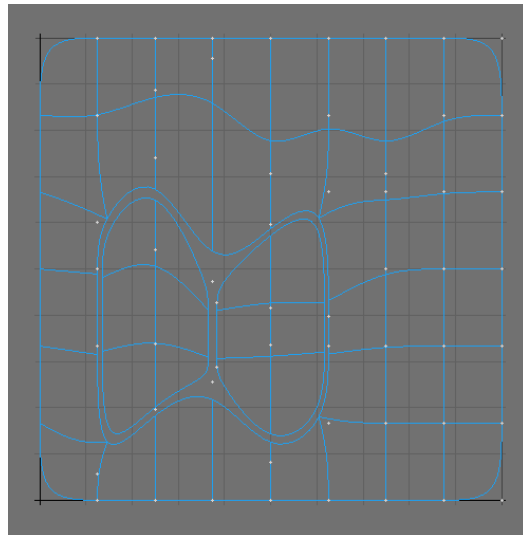
**Linear:** The UV map is interpolated linearly, most similar to the original mapping option and does not use any subpatching at all.



A UV map with two UV islands enclosed by a 3rd large UV island. Linear Interpolation Mode

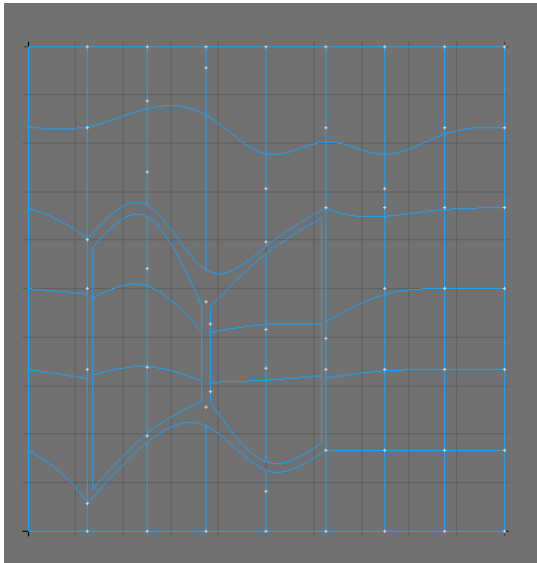
The edges on the very left and right of this UV map are discontinuous, as are the edges along the two small islands in the centre. In this case the biggest problem is that some vertices are shared by all three UV islands. This makes many of the interpolation- methods fail. The situation would be even more severe, if each UV island was in its own UV map.

**Subpatch:** All Edges are interpolated along the subpatch lines, as if the UV map was a subpatch mesh. (Note that the discontinuous edges do not line up.)



Subpatch Interpolation Mode

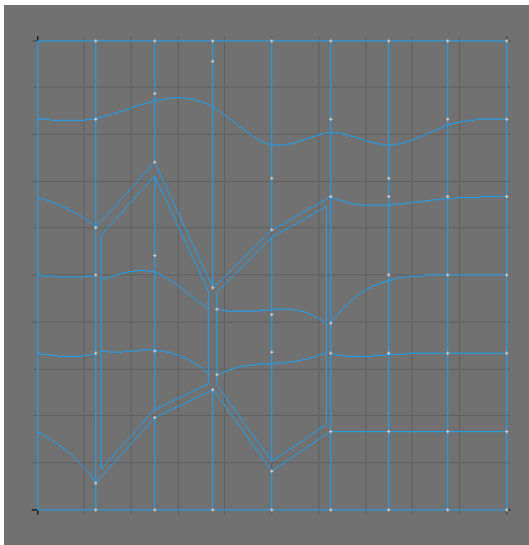
**Linear Corners:** Corner-Points that are along the outer edge of the UV map, but not the edge of the mesh, are interpolated linearly. Notice that it looks better than subpatch, but the discontinuous edges still do not meet up.



Corners Interpolation Mode

Linear

**Linear Edges:** Edges that are along the outer edge of the UV map, but not the edge of the mesh, are interpolated linearly. Discontinuous Edges now line up, but due to lack of cubic interpolation, some distortion may still occur.



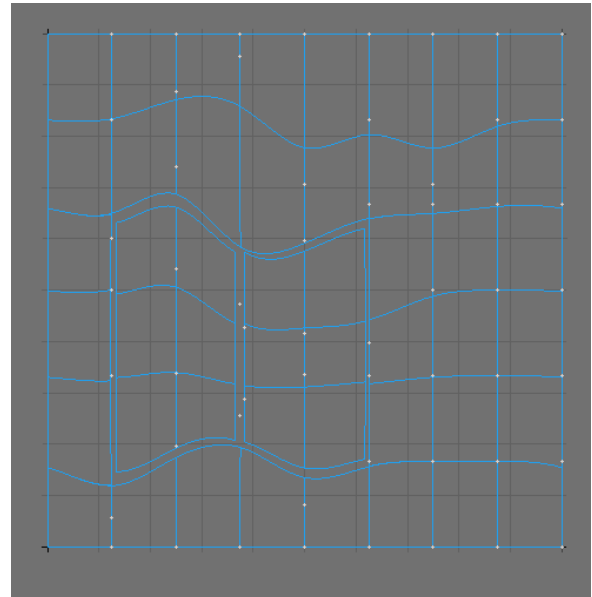
Interpolation Mode

Linear Edges

In this special case only the “Linear Edges” method would allow for well aligned discontinuous UV edges. However you would have to allow for some distortion along these edges (similar to the way it used to be before LightWave 8.2).

**Across discontinuous edges:** Interpolation occurs along discontinuous edges. This method usually provides less distortion along discontinuous edges. A discontinuous edge can be considered a point that shares two separate polygons on a mesh,

yet falls on the edge of the UV map so the two polygons are on opposite edges of the UV map. This option lines up the points so the two edges meet up. Please note that this option still does not work across multiple UV maps.



Across Discontinuous Edges Interpolation Mode



**Warning:** If heavy scaling is used or parts are rotated, Across Discontinuous Edges can produce errors. However, this solution is recommended first, since most of the time, the least distortion occurs at discontinuous edges.

The Subpatch Interpolations can be further edited in the Assign UV Coordinates window and the Vertex Maps Panel.

## Axis

Choose the axis you wish to align your UVs to: X, Y or Z.

## Settings

In addition to choosing the axes, you can also decide whether to leave the **Center** and **Size** settings on **Automatic**, or to adjust them **Manually**. If you click on **Manual**, the **Center** and **Size** fields will become active. For most purposes, it's recommended to simply leave them on **Automatic**.

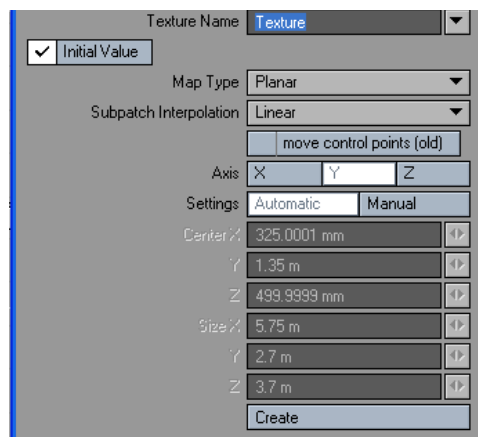
Once you have a **UV Map** and are ready to paint it, don't make a screenshot, which will be of poor quality at best, and will need to be manipulated before you can use it. Instead, go to **File > Export > Export Encapsulated Postscript**, choose **Texture UV** from the **View** field, and make an .eps file, so you can have clean, high-resolution maps.



## New UV Map

This button allows you to make a new UV map.

### How do I use this tool?



When you click the button, it opens a dialog that looks exactly like the one you get when you choose **(new)** from the **T** button in the **Vmap Bar**.

### Texture Name

First, it allows you to name your new **UV Map**. Type the name you desire into the text field, or choose a name from the drop down list. The list contains all the names of all the **UV Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

### Initial Value

You may also start by assigning your map an **Initial Value**. If you do, several other fields become available. If you don't, they remain dimmed, of course. You may or may not want to do this, depending on how you are planning to build your map, and the shape of your object.

If you do, **Map Type** will offer you four different kinds of maps to choose from. These correspond to the standard projections you may already be familiar with.

## Make UVs

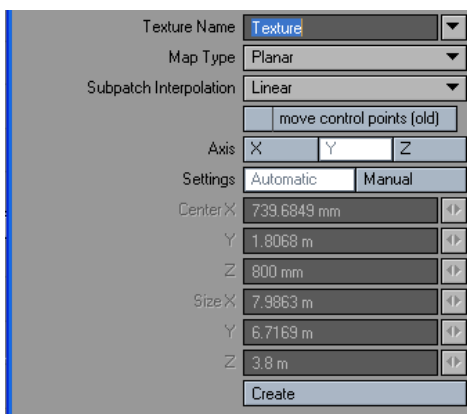
This button allows you to assign selected polygons to a **UV Map**. If no polys are selected, then all visible polys will be mapped.

### How do I use this tool?

This button is normally used when you've decided to make your own **Atlas Map** by hand. (An **Atlas Map** has polygons separated and spread out for painting.) Say, for instance, that you want to map a can of beans. The can itself would be best mapped as a cylinder, since it is one. The top and bottom, on the other hand, are really flat planes, and should be mapped using the **Planar** method.



There's no built-in way to do that, and that will give you a map that's simple to paint. But you can do it yourself very easily by selecting all the can polys, and using this button to make a **Cylindrical Map**. Then unweld the points on the seam, slide the "can" down, select the polys for the top and bottom, cut them and paste them back, and use the button again to get UVs with **Planar Mapping**. Select them separately (select connected) and slide them apart. Hit "**M**" to merge all the points you unwelded, and your can is perfectly mapped, and ready to go; and this is all possible thanks to the **Make UVs** button. You can guess from the image above which one is easier to paint!



When you click the button, it opens a dialog that looks just like the **New UV Map** dialog, except that it doesn't have an **Initial Value** button.

### Texture Name

Choose the name of the **UV Map** you would like these polys assigned to from the drop down menu. The list contains all the names of all the **UV Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them. If you would like to start a new map here, you can. Just type the name for the new map into the text field.

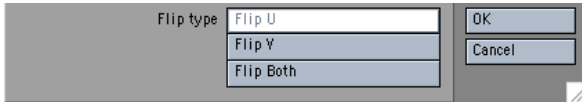


## Flip UVs

This button allows you to flip the UVs of selected polys, so that the polys which are showing a mirror image of the texture can be fixed, or simply so that they fit more conveniently onto the **UV Map**.

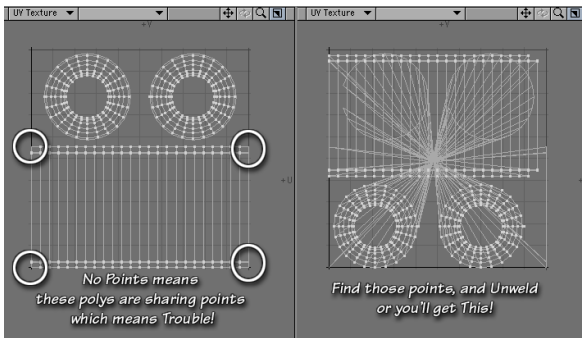
If no polys are selected, then all visible polys in the current map will be flipped.

### How do I use this tool?



When you click the button, it will open a dialog which allows you to choose to flip the polys across the U (horizontal) axis, across the V (vertical) axis, or both. (Flipping both, of course, won't fix mirror images in your render; but it might put the polys just where you want them.)

Simply click to choose a direction, and then click **OK**. The polys will flip.



In order to do this successfully, you must be sure that you have unwelded all the seams on your model. If you don't, any polys that aren't showing a point will remain "stuck" to the map in that place, and will become severely stretched. (In the picture shown, you can see that there are points missing in the corners marked. There are also points missing all around the circumference of both circles. Whenever two polys share a point, only one will have it on the map, and it's a problem if it's on the edge)

To fix it, undo, then select the points that are being shared (On the model, they'll be on the poly next to the one without its point. If you select that poly on the map, it will light up on the model too, and you'll be able to find which points to unweld easily.)

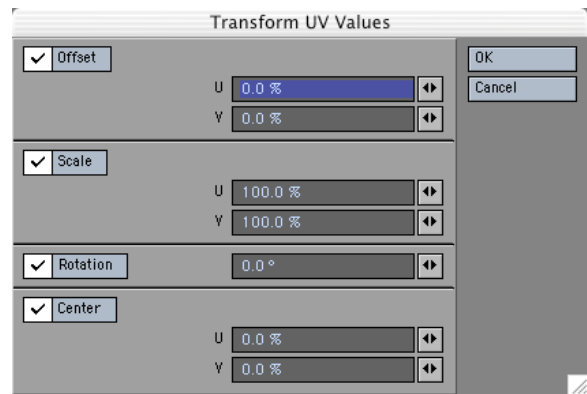
**Go to Detail > Points > Unweld**, and every poly will be given its own points.

When all the edge polys are showing points, go ahead and flip.

## Transform UV

This button allows you to adjust the **UV** values of selected polygons in several ways. If no polys are selected, it will act on all the visible polys in the current map.

### How do I use this tool?



When you click the button, it will open a dialog which allows you to choose several ways to manipulate the **UV** values of selected points on your map. (If no points are selected, it will act on all of them.)

### Offset

This option will move the points by the specified percentage of the **UV** space, in either the **U** (horizontal) or **V** (vertical) direction. Positive values will move them to the right, or up. Negative values will move to the left, or down.

### Scale

This option will scale the points by the specified percentage, in either the **U** (horizontal) or **V** (vertical) direction. 100%, of course, is the current size. Larger numbers will increase the size, and smaller will decrease it. Unless you specify a different center, they will shrink or grow from 0,0, which is at the bottom left corner, where the black axis lines cross.

### Rotation

This option will rotate the points around the center point. Positive values will move them clockwise, negative values will move them counter-clockwise. Unless you specify a different center, they will rotate around 0,0 or the bottom left corner, where the black axis lines cross.

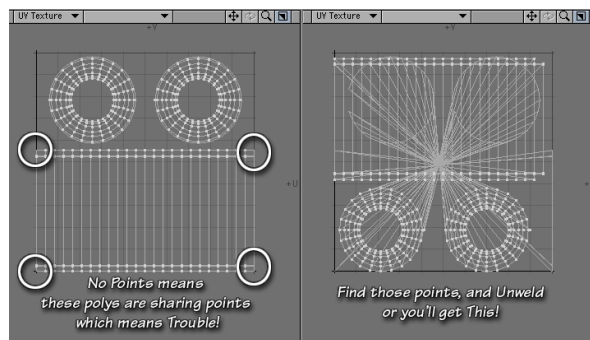
### Center

This option allows you to use a different center for the **Scaling** and **Rotation**, specified as a percentage of the **UV** space in either the **U** (horizontal) or **V** (vertical) direction. Positive values move the center to the right or up. Negative values move it to the left or down.

Of course, you can also use the normal **Modify** tools (**Move**, **Drag**, **Snap**, **Scale**, **Stretch**, **Rotate**, and so on) on **UV Maps**, with much the same results you would expect if you used them in a normal Modeler window. As long as you don't use the Numeric



Panel, nothing you do in the UV Window will affect the geometry of your model at all, since **UV** space is separate from X,Y,Z space.



Be aware, though, that in order to do any of this successfully, you must be sure that you have unwelded all the seams on your model. If you don't, any polys that aren't showing a point will remain "stuck" to the map in that place, and will become severely stretched. (In the picture shown, you can see that there are points missing in the corners marked. There are also points missing all around the circumference of both circles. Whenever two polys share a point, only one will have it on the map, and it's a problem if it's on the edge)

This image shows the use of the **Flip** tool; but the same principle applies.

To fix it, undo, then select the points that are being shared (On the model, they'll be on the poly next to the one without its point. If you select that poly on the map, it will light up on the model too, and you'll be able to find which points to unweld easily.)

**Go to Detail > Points > Unweld**, and every poly will be given its own points.

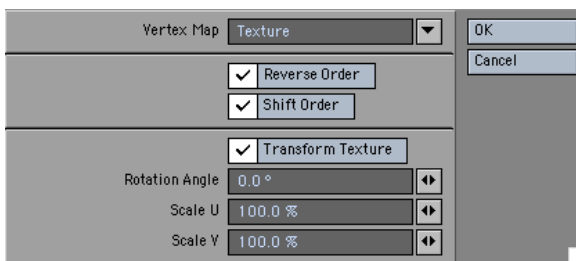
When all the edge polys are showing points, go ahead and manipulate them.

## Poly Map

### Adjust Polygon Map

This button will allow you to adjust your old **Per-Polygon Maps**, or to treat new **UV Maps** as if they were **Poly Maps**, and adjust them.

#### How do I use this tool?



Select the polygons that use the **Poly Map**, and click on the command to open a dialog, which lets you adjust them in several ways. It will also let you treat normal **UV Maps** as if they were **Poly Maps**, although it's better to use the newer tools. To do that, select the polygons, and enter the map's name in the **Vertex Map** field. (If you are editing real **Poly Maps**, leave that field blank.) This tool will not work on unselected polys.

**Reverse Order** and **Shift Order** adjust the connection between texture points and the polygon's vertices. Checking **Transform Texture** will enable the **Rotation Angle**, **Scale U** and **Scale V** fields. They all allow you to perform 2D transformations of the **UV** coordinates. The power of these operations is limited in comparison to the tools available for editing **VMaps**.

If you want to use those tools on your **Poly Maps**, simply use the **Unweld** command (**Detail > Points > Unweld**). This will put points on all the polys, and remove **Per-polygon UV Mapping**. At that point, you can use all the **Modify** tools, as you normally would. If your models are simple enough, you can **Unweld** selected points; if not, you can effectively unweld the entire model. Just be sure not to move anything in X,Y,Z space, and the **Merge** command (**M**) will put it all back together again when you're finished mapping.





## Clear Polygon Map

This button will clear the per-polygon UVs from their polygons. Just select them, and click. All the textures will simply vanish.

On the other hand, if you want to translate the old mapping into the new discontinuous **UV Maps**, then you'll want to use the **Poly Map to UVs** button (**Map > Texture: Poly Map to UVs**), and do that instead.

### How do I use this tool?

Just select the polygons you wish to clear, and click **No more maps**.

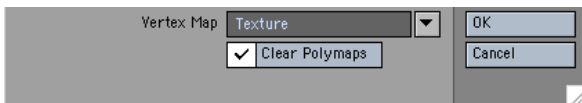
## Lock UVs to Polygon

This button is used to assign the old per-polygon maps to the polygons. Unless you intend to use the older style maps, you really don't need to use this.

## Poly Map to UVs

If you want to convert a **Per-polygon Map** to the more elegant and flexible **Vertex UV Map**, just select the polys you want to convert, and click this button.

### How do I use this tool?



### Vertex Map

Choose a map name. Either type one in, or choose one from the list.

If you have a viewport set to **UV Texture**, you'll see the **Poly Map** polygons appear there. If you look, though, you'll notice that there aren't any points on them; so although you can now see them, you still can't manipulate them.

To fix that, just use the **Unweld** command (**Detail > Points > Unweld**). This will put points on all the polys, allowing you to use all the **Modify** tools, as you normally would. If your models are simple enough, you can unweld selected points; if not, you can effectively unweld the entire model. Just be sure not to move anything in **X,Y,Z** space, and the **Merge** command (**M**) will put it all back together again when you're finished mapping.

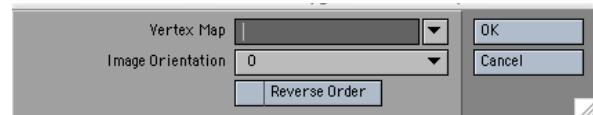
## New Quad Polygon Map

New Quad Polygon Map (**Map > Texture: Poly Map > New Quad Polygon Map**)

This button allows you to make a new **Per-polygon Texture Map** where each selected quad (four-sided poly) covers the entire map square.

### How do I use this tool?

Select the quads you want to map. (It does nothing with unselected polys, and nothing with polys that aren't quads.)



### Vertex Map

Click the tool, and a dialog will open. If you want to make a discontinuous (normal) **UV Map**, then enter the name in the **Vertex Map** field, or choose an existing map name from the drop down list. The list contains all the names of all the **UV Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

If you want to make a **Per-polygon Map**, leave the **Vertex Map** field blank.

### Image Orientation

Choose an **Image Orientation**, either 0, 90, 180, or 270 degrees.

### Reverse Order

Click **Reverse Order** if you think the texture is going to be a mirror image.

Click **OK**. Each corner of each quad will be assigned to one corner of the new **Quad Map**, so the entire picture will exist on each one. This works best for boxes and other simple shapes; but it can be very interesting on more complex objects.

If you have enabled the **Texture Map** in the **Surface Editor**, you'll be able to see it. (If you haven't, you'll need to go and do that. To get a per-polygon texture to show up, choose an **Image Texture**, with **UV Projection** as normal. Choose any **UV Map**. It doesn't even need to have any points assigned to it. Then choose (or load) your image. You'll see it on the quads in **Texture** or **Wireframe Texture** viewports. To see a normal **Vertex UV Texture**, you have to choose the same **UV Map** that you named in the dialog, of course. In either case, if you are using more than one image layer in the **Texture Layer Stack**, only the one on the bottom will be visible in the viewports. So, if you want to see your **UV Projection Image**, make sure it's on the bottom.

You might find that the orientation of the images isn't what you'd hoped, or the picture is a mirror image of itself. If that happens, select the problem quads once more, and try again. Choose a different **Orientation**, if that's the problem, or **Reverse Order** to fix the mirror image. This is much easier to do if you adjust the ones that are causing problems singly, or in groups that have the same orientation, because the corners of the quads are assigned to the corners of the **UV Map** arbitrarily. So, if you leave them all selected, you may find that you are fixing some, while others are turning the wrong way again.





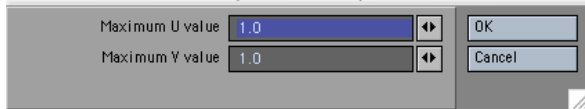
## Point Map

### Spread UVs

This button takes the corners of a selected quad, and places them in **UV** space at the minimum and maximum values you specify.

#### How do I use this tool?

First, select a single quad.



Click the button, and it will open this dialog which allows you to pick the **Maximum** value in the **U** and **V** directions. Notice that there are no minimum values; the minimums are 0,0; the lower left corner of the **UV Map**, where the U and V axes meet.

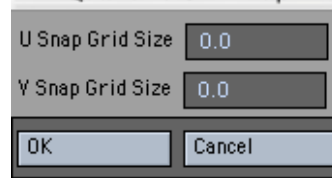
Whatever you pick for the **Maximum U** will be the right side of the quad. Whatever you pick for **Maximum V** will be the top. The left side and bottom will be against the axes, of course.

You will notice that, once the quad is spread, it will no longer have any points, which means that you cannot manipulate it with the normal Modeler **Modify** tools. Fortunately, that's easy to fix. Just use the **Unweld** command (**Detail > Points > Unweld**). This will put points on the poly, allowing you to use all the **Modify** tools, as you normally would. If your models are simple enough, you can **Unweld** selected points; if not you can effectively **Unweld** the entire model. Just be sure not to move anything in **X,Y,Z** space, and the **Merge** command (**M**) will put it all back together again when you're finished mapping.

### Quantize UVs

This button divides the **UV** space into a grid of the specified size, and then snaps selected points to the closest intersection of the grid. If no points are selected, it will snap all visible points to those lines.

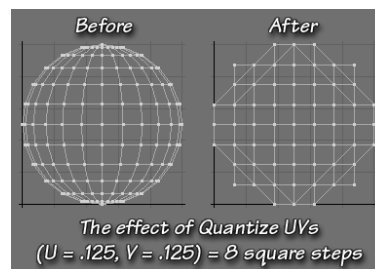
#### How do I use this tool?



To use this tool, select the points you want to quantise, and click the button. (Once again, if you don't select anything, all points will be quantised.) A dialog will open that allows you to specify the size of the **Grid Snap**, from 0.0 to 1.0. To find the number to enter, divide 1 by the number of squares you want in the grid. For instance, if you want 16 squares, you'd use 0.0625, if you want 5 squares, you'd use 0.2, and so on.

Enter the number for both the **U** (horizontal) and **V** (vertical) axis. If you don't want any adjustment on one axis, just leave it at zero.

Then click **OK**. The points will snap to the closest grid intersection.

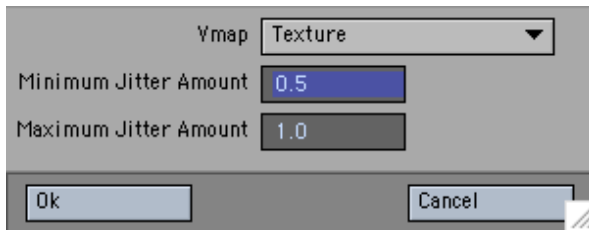




## UV Map Jitter

This button allows you to add a random **UV** value to all selected points. If no points are selected, it will act on all visible points.

### How do I use this tool?



To use it, choose a map from the **Vmap Bar** at the bottom of the screen (click **T** and choose from the drop down list.) Select the points you want to jitter, or leave them unselected if you want to jitter all the visible points. Click the button, and the dialog will open.

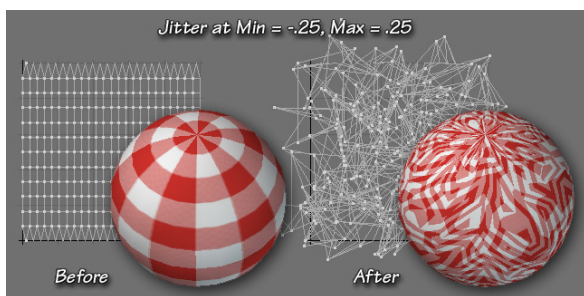
### Vmap

Choose the **Vmap** you want to use from the drop down list. The list contains all the names of all the **Vmaps** used during the current session, whether or not they are **UV Maps**, whether or not the objects remain open, and whether or not the maps have any points assigned to them. (But if you try to use anything but the **UV Map** selected in the **Vmap Bar**, you'll get an error message.)

### Minimum/Maximum Jitter Amount

Set the **Minimum** and **Maximum Jitter** amount. But remember, when you do, that the entire map is only 1.0 big. So set tiny, tiny values unless you want the map to jitter right off the UV Square, and out into the wrap area. You can also use negative values, of course, to keep it more or less around the map, if you are so inclined.

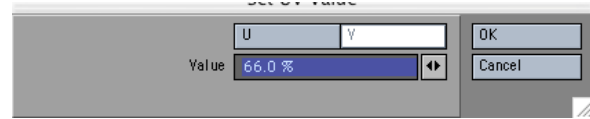
Click **OK**, and you'll see the jitter effect.



## Set UV Value

This button allows you to move selected points to a specific spot in either **U** or **V** space. If no points are selected, it will move all visible points. The effect is to line them all up.

### How do I use this tool?



To use it, choose a map from the **Vmap Bar** at the bottom of the screen (click **T** and choose from the drop down list.) Select the points you want to move, or leave them unselected if you want to move all the visible points. Click the button, and the dialog will open.

Since the points are going to be lining up, you need to move them perpendicular to the axis you want them to line up on. So, if you want a vertical line, you need to move them all to the same point on the **U** (horizontal) axis. If you want a horizontal line, you need to move them all to the same place on the **V** (vertical) axis. Choose the axis you prefer by clicking its name.

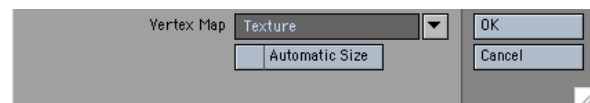
The spot on the axis is expressed as a percentage of the **UV** space. 0% is the lower left corner, where the two black axes lines meet. 100% is the top right corner.

So, if you want to move the points to line up one quarter of the way from the left, you'd enter **U 25%**. If you wanted them to line up two-thirds of the way to the top, you'd enter **V 66%**, and so on. It's all relatively straightforward.

## Polygon Normal UVs

This command will take all the selected polygons, and project them onto a **UV Map** according to their normals, so they will be mapped flat on the image. It's one of the old **Per-polygon** tools; it's similar to mapping in **Atlas** mode, except the polys wind up all piled on top of each other, without any points.

### How do I use this tool?



Select all the polys you want to map, polygons that aren't selected won't be mapped, and click on the command. A dialog will open.

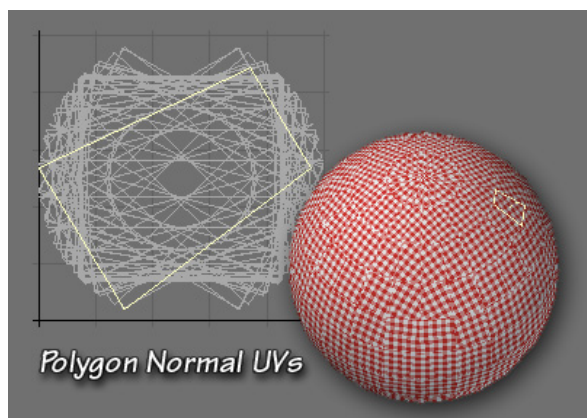
### Vertex Map



Type the name of the map into the **Vertex Map** field, or choose a name from the drop down list. The list contains all the names of all the **UV Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them. If you type a new name into the field, a new map will be created, and chosen from the **Vmap Bar**. If you use a name from the list, that map will also be automatically selected. If you don't use any name, you'll make a **Poly Map**.

### Automatic Size

Click the checkbox if you want the polys to be automatically sized. If you do, they'll be centered in the **UV** space, and come close to filling it. If you don't, they'll probably be tiny, and in the lower left corner.



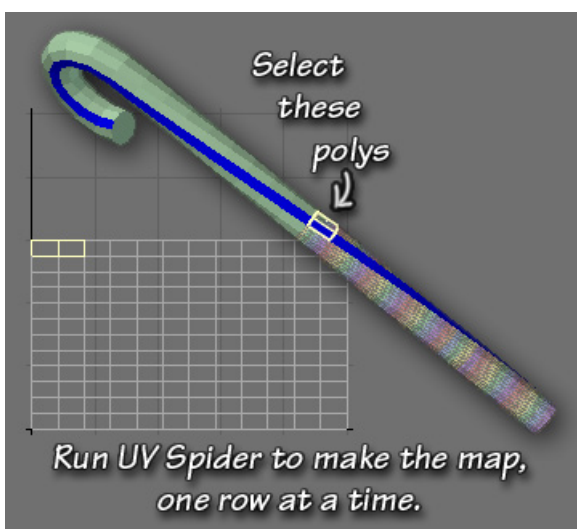
Click **OK**, and they'll be made. You'll notice that they don't have any points, which means that you can't manipulate them directly. Fortunately, that's easy to fix. Just use the **Unweld** command (**Detail > Points > Unweld**). This will put points on all the polys, allowing you to use all the **Modify** tools, as you normally would. If your models are simple enough, you can unweld selected points; if not, you can effectively **Unweld** the entire model. Just be sure not to move anything in **X,Y,Z** space, and the **Merge** command (**M**) will put it all back together again when you're finished mapping.

## UV Spider

This command will crawl along a line of selected quads, no matter how much it twists and turns, and map it flat onto a specified section of **UV** space. Think **Bandsaw**, but mapping instead of cutting.

### How do I use this tool?

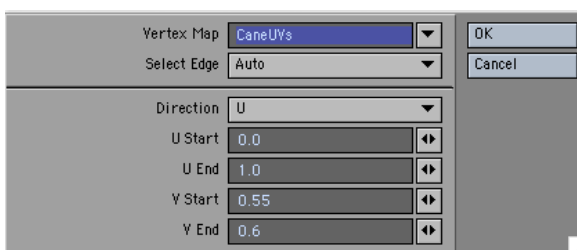
This tool is really interesting. You can use it to unwrap any regular quad surface, even if you have twisted, turned, bent and vortexed it! But it does take some planning.



The tool works by crawling along a line of quads, and depositing them neatly onto a **UV Map**. But it only crawls along one line at a time. So, in order to unwrap an object, you have to select a couple of quads to get it started, just like you would with **Bandsaw**. However, it will put the first quad you select against the edge of the map. So, if you want a usable map, you have to start with the right quad, every time. In other words, if you're unwrapping horizontally, you need to start with the next quad in the vertical row in order to get a good map. Now, if the piece has been twisted and turned, that's not simple to do. (If it hasn't, you don't need this tool; there are easier ways to map straight things.)

You should begin by selecting a line of quads running perpendicular to the direction you plan to map. You can do that easily by selecting a couple of quads in the correct direction (vertically, in our example) and going to **View > Selection: More > Select Loop**. Once you have it, change the surface to something that will contrast nicely with the other surface. (Tap **Q** or click **Surface** at the bottom of the screen.) It's only for mapping, and you can change it back later; but it will help you keep your place.

Then, select two quads horizontally in the bottom row of the area you want to map, and click the button.





## Vertex Map

Choose which **UV Map** to work on by typing the name into the **Text** field, or selecting it from the drop down list. The list contains all the names of all the **UV Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them. If you type a new name into the field, a new map will be created. If the map you pick (or create) isn't chosen in the **Vmap Bar** at the bottom of the screen, it will be when you click **OK**.

## Select Edge

This option allows you to choose in which direction the **Spider** will run. Like similar tools in LightWave, if you've chosen two polys, **Auto** will give you the correct direction. Otherwise, you can choose **Odd** or **Even** by using the menu (but it's more predictable - and therefore faster - just to grab two polys.)

## Direction

Choose the direction you want the UVs to run in. If you are mapping horizontal polys, choose the **U** direction. If you are mapping vertical polys, choose **V**.

## U Start, U End

Choose the place on the map for the polys to be laid down. You can safely leave this at 0.0 to **Start** and 1.0 to **End** for the direction you are mapping. So, in our example, where we're mapping in **U**, we would leave **U Start** at 0.0, and **U End** at 1.0.

## V Start, V End

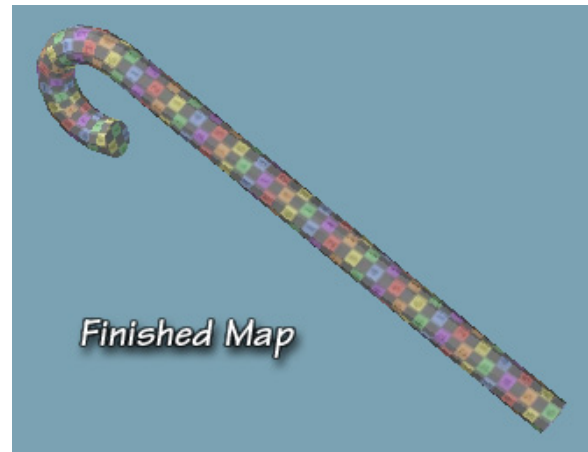
Use small increments in the other direction, so the polys will be laid down row by row. In our example, if we started with **V Start** at 0.0 and **V End** at 0.05, we'd get one row of polys, the width of the map, and one twentieth of its height. Click **OK**, and the row is made. Drop the polys.

Select the first two polys of the next row, and click the command again. The dialog box opens with the last settings intact, so you can see where you are. Simply change the **V Start** to .05, and the **V End** to .1, and make another row. Continue in that fashion, increasing both numbers by the same increment, until you have mapped the entire area.

When you are finished, you may notice two things. First, your map may be longer than the **UV** space. Second, there aren't any points on it at all. That may be fine with you; in which case you're done.

But, if you need to further manipulate the map, it's easy to do. Just use the **Unweld** command (**Detail > Points > Unweld**). This will put points on all the polys, and allow you to use all the **Modify** tools, as you normally would. If your models are simple enough, you can **Unweld** selected points; if not, you can effectively **Unweld** the entire model. Just be sure not to move anything in **X,Y,Z** space, and the **Merge** command (**M**) will put it all back together again when you're finished mapping. (Don't merge until you finish, though, so wait until you go through the next step.)

Once you have points, you can use the **Stretch** tool (**Modify > Transform: Stretch** or **H**) to pull the polys down so they fit in the **UV** space, and your image won't wrap.





## Guess Viewport UV Image

(default keyboard shortcut **Shift F9**)

This command guesses what image should be used as the background in the **Viewport** set to **UV Texture**.

### How do I use this tool?

Just set one of your viewports to **UV Texture**, and choose a **Texture Map** from the **Vmap Bar** at the bottom of the screen. Click the button, or tap **Shift F9** on your keyboard, and the image that's used in the texture will automatically load as the Background Image in the viewport.

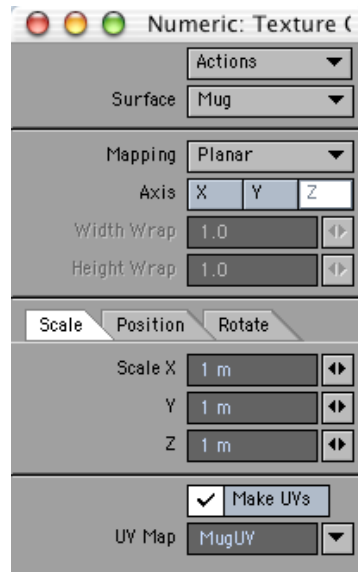
If you're using more than one image in the surface that uses that **UV Map**, it will pick the one from the **Color Channel**, if there is one.

It's just a quick, handy way of loading the Background Image. But be careful with it. If you're using a **Poly Map** or a **Color Vertex Map** instead of an image, LightWave can grind to a halt while it tries to guess that.

## Texture Guide

This command allows you to interactively create a **UV Map** for selected polygons.

### How do I use this tool?



To use this tool, make sure that one of your viewports is set to **UV Texture**, and select the polygons you would like to map. (If there are no selected polys, this tool will map all visible polygons.)

If you would like to see the texture on your object while you make the UVs, you will need to assign an existing **UV Map** to that surface in the **Surface Editor**. (Open the **Surface Editor**, Click the **T** next to **Color** in the **Basic Tab**, and then choose **Layer Type = Image Map**, **Projection = UV**, **UVMap =** the map you want to use, **Image =** the picture you want, and then **Use Texture** at the bottom. Close the **Surface Editor**, and you're good to go.



**NOTE:** if you are using more than one image map in the **Texture**, the one with **UV Projection** must be on the bottom of the layer stack in order to be visible.

Click the command, and open the **Numeric Panel**, if it's not already open. Click on the **Actions** button, and choose **Activate** from the drop down menu. The rest of the panel will light up, and the selected polys will be visible in the **UV Texture** viewport, if you are working with an existing map.

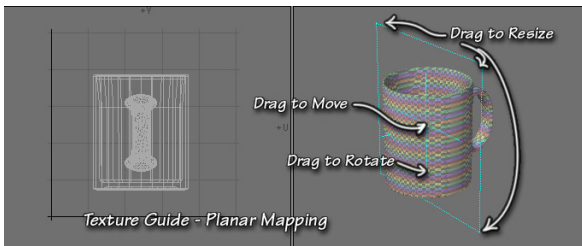
Click **Make UVs** at the bottom of the **Numeric Panel**, and either type a new name into the **UV Map** field to create a new map, or choose an existing map from the drop down menu. The list contains all the names of all the **UV Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

When you do, that map will become the active map in the **Vmap Bar** at the bottom of the screen, and you'll be able to see the UVs for the selected polys in the **UV Texture** window. The fun is about to start!

Choose the **Surface** you would like to make the UVs for. All the visible surfaces in the object will be listed.

Then choose your **Mapping** method. There are five to choose from.





## Planar

**Planar** is very similar to the **Planar Mapping** method you may already be familiar with. It projects the UVs onto the **UV Map** using a flat plane. When you choose it, you'll see that plane represented as a cyan rectangle in the viewports. You can choose the axis using the **X**, **Y**, **Z** buttons below the **Mapping** field. As you do, you will be able to see the results of the various orientations in the **UV Texture** viewport.

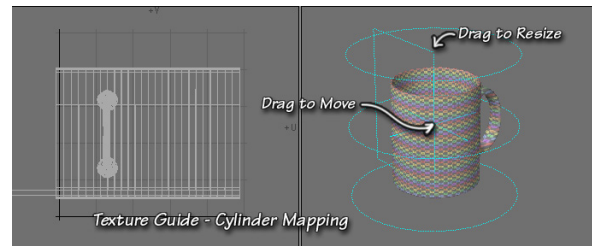
You can scale the map interactively by dragging the corners of the cyan mapping rectangle, or by using the spinners or typing numbers into the fields in the **Scale Tab**. As you do, you will notice that, the larger the mapping rectangle, the smaller the map, and vice versa. This is because the object is occupying a smaller (or larger) percentage of the mapping rectangle. If you hold down the **Ctrl** key before you begin to drag, the scaling will be constrained to the last drag direction.

You can change the position of the plane by dragging the center cross, or by using the spinners or typing numbers into the fields in the **Position Tab**. Holding the **Ctrl** key will constrain the dragging to vertical only. Notice as you drag that when you drag right, the mapped object moves left in the **UV Texture** viewport, and vice versa. This is because it's occupying the left (or right) side of the mapping rectangle.

You can rotate the mapping rectangle in the direction perpendicular to the rectangle by clicking and dragging on the dotted cyan handles that extend from the center square. However, you can only interactively rotate on the axis perpendicular to the rectangle. If you want to rotate it in all three directions, you will have to use the numbers or spinners on the **Rotate Tab**. The tab uses **Heading**, **Pitch**, and **Bank**; controls that you should be familiar with from **Layout**.

**Heading**, of course, controls movement around the **Y** axis, **Pitch** around the **X** axis, and **Bank** around the **Z** axis. So, if you have chosen the **Z** axis for mapping, you can drag to change the **Bank**, but **Heading** and **Pitch** can only be changed in the Numeric Panel.

Notice, once again, that the object appears to move in the opposite direction to your movement when you drag, because you are changing the corner of the mapping rectangle that it occupies.



## Cylinder

**Cylinder** is the same as the **Cylindrical Mapping** mode you have seen in other places. It projects the vertices onto a cylinder, and then unwraps that flat onto the **UV Map**. The cylinder is visible as a simple cyan wireframe cylinder in your viewports.

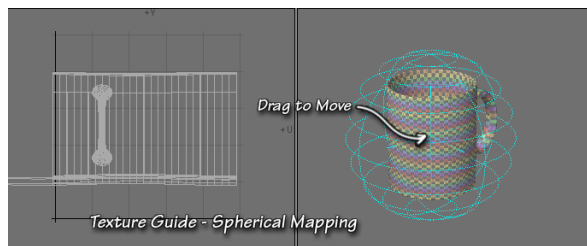
Choose the axis you would like to use as the length of the cylinder by clicking on the appropriate button (**X**, **Y**, or **Z**.) You can also choose the **Width Wrap**, which determines how far the polys will spread in the **U** direction on the map. You can adjust it to keep the polys proportional as you change the scale of the cylinder.

You can change the height of the cylinder by dragging on the top or bottom line. To change the other two directions, you must use the spinners or type the numbers into the fields in **Scale Tab** of the **Numeric Panel**. Once again, the larger the cylinder, the less space the polys take, and the smaller they appear on the map.

You can change the position of the cylinder by dragging the central cyan cross. If you want to be more exact, you can use the fields in the **Position Tab** of the **Numeric Panel**.

Finally, you can rotate the cylinder; but only by using the fields in the **Rotate Tab**.

It's interesting to note that each mapping method inherits the **Scale**, **Position**, and **Rotation** that are already present. So, if you wish to interactively rotate the cylinder, simply start with **Planar Mapping**, rotate the rectangle, and then switch. If you chose the correct directions, your rotation will be in place.



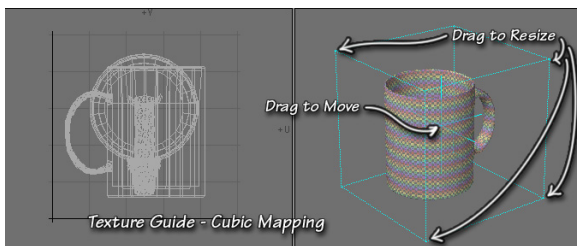
## Spherical Mapping

**Spherical Mapping**, once again like **Spherical Mapping** in other parts of LightWave, projects all the points onto a sphere, and then unwraps it onto the flat surface of the map. The sphere is represented by a cyan wireframe sphere in your viewports.

As always, choose the axis you wish to use as the poles of the sphere. You can also adjust both the **Width Wrap** and the **Height Wrap**, spreading the polys out in the U or V direction, to obtain the desired result.

You cannot interactively change the **Scale** or **Rotation** of the sphere; you have to do all of that using the tabs in the **Numeric Panel**. You can, however, change the position by dragging the center blue cross, and you can watch the UVs update in real time as you do so.

The regular caveats about size and direction which seem to be backwards apply again, of course, as does the note about the persistence of previous adjustments, as long as you haven't closed the tool.

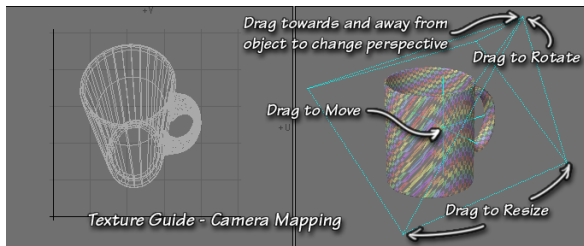


## Cubic Mapping

**Cubic Mapping** corresponds to the **Cubic Mapping** you may know from the **Surface Editor**. The projection space is represented by a cyan cube in your viewports.

You can scale the cube interactively by dragging on any of the corner handles, or you can use the spinners or type numbers into the fields in the **Scale Tab** for greater accuracy. You can position it by dragging the center cyan square, as always, or by using the **Position Tab**. To rotate, though, you have to use the **Rotate Tab**, and type or spin. The usual caveats apply here, too.





## Camera Mapping

**Camera Mapping** is an extremely flexible planar map that uses perspective. It's represented by a cyan pyramid in the viewports. You can think of the flat, rectangular part of the pyramid as the picture that will be used for the map, and the point as the camera lens.

You can scale the rectangle by dragging on its corners, or by using the **Scale Tab** of the **Numeric Panel**. You can change the position by dragging the center cyan across in any viewport, or by using the **Position Tab**. You can change the rotation by dragging the point of the pyramid in any viewport or by using the **Rotate Tab**.

You can change the perspective amount by moving the point towards or away from the object. This changes the **Camera** field of **View**. As is normal with cameras, the wider the field of view, the more perspective you have. (You can do a kind of fisheye **UV Map**, if you are so inclined.) The narrower the **FOV**, the less perspective.

All the usual caveats apply here, as well.

As you do all of this, no matter which **Mapping** type you are using, you may find that the UVs have moved out of the **UV** space. It's not at all uncommon for them to do so. If you can't find them, zoom out. (and if all else fails, close and reopen it, they will show up again)

When you are pleased with your results, tap the **Return** key or the **Spacebar** to drop the tool and make the UVs.

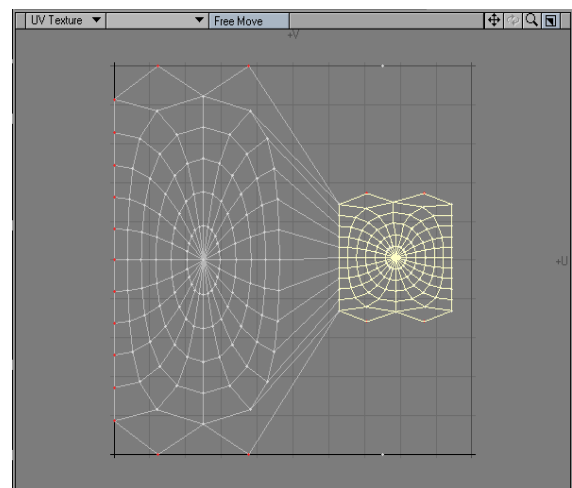
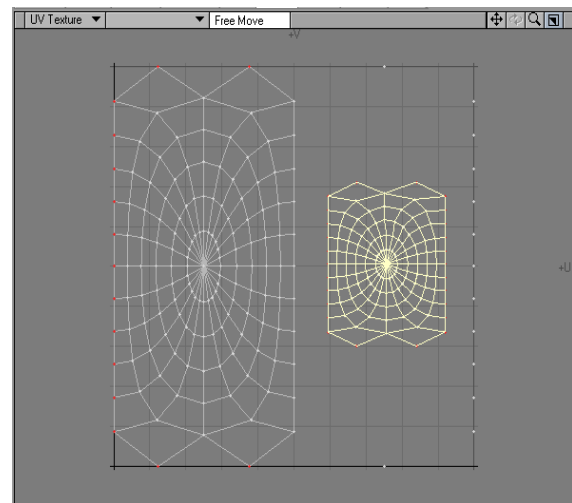
You will notice that there are no points in them anywhere; which means that you can't perform any direct manipulations on them. They might be exactly what you want, in which case, you're finished.

But, if you would like to do more, it's easy to turn this map into a normal one, with points. Just use the **Unweld** command (**Detail > Points > Unweld**). This will put points on all the polys, allowing you to use all the **Modify** tools, as you normally would. If your models are simple enough, you can **Unweld** selected points; if not, you can effectively **Unweld** the entire model. Just be sure not to move anything in **X,Y,Z** space, and the **Merge** command (**M**) will put it all back together again when you're finished mapping.

## Free Move

When in UV Texture mode in the viewports and Free Move is selected, selected polygons will separate when modified.

*Above, Free Move On; Below, Free Move Off*





## Morph

### What is an Endomorph?

Sometimes, when you are changing the shape of your objects in Layout, you'll find that the deformations you can get with bones just aren't accurate enough. Fortunately, you can also force an object to change its shape into the shape of another object. This is called morphing. Morphing gives you total control, since you can manually place all the points exactly where you want them. You can use morphing to help create character facial expressions, to make a flower bloom, to transform a boat into a car, and then into a rocket; anything you want to do, as long as both objects have the same point count and order.

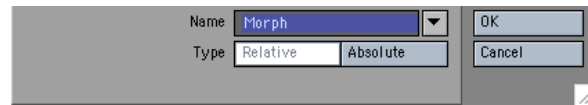
The different states, or poses, of a morph are generally referred to as morph targets. They are made using morph maps, which are a form of **Vertex Maps** that define the different positions for the points. The original state of the object is known as the Base. The map is known as an **Endomorph**, and is incorporated right into the Object file, like any other **VMap**. If you copy and paste the object, all its morphs are copied and pasted right along with it.

When you make a morph, you can push and pull the points around, and place them anywhere you like. However, although it's possible to add or subtract geometry within a morph, it's strongly recommended that you refrain from doing so, and make all such modifications to the Base object instead. Otherwise, you can end up with incorrect point orders.

## New Endomorph

This command allows you to create a new **Endomorph**, or **Morph Target**.

### How do I use this tool?



When you click on the command, a dialog box will open that's not unlike the dialog you get when you click on **W** in the **Vmap Bar** and choose (**new**) from the drop down menu.

### Name

Enter a name for your new morph in the **Name** field, or choose an existing name from the drop down menu. The menu contains all the names of all the **Morph Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them.

It's suggested that you name your morph using the group.pose format; like Eyes.Open or Mouth.Smile. That will cause the morph to appear on the same tab with all the other poses in that group when you use **Morph Mixer** in **Layout**.

### Type

Choose the **Type** for the new morph. Relative targets record point displacement relative to the base object, so they are affected when the base object moves. Absolute targets are not affected by positional changes to the base object, although they do reflect changes to the number of points in the object.

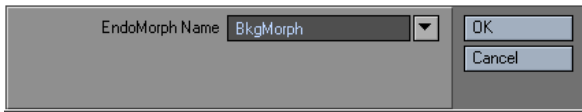
Click **OK** to create the new **Endomorph**, and start to move the points around. (If you don't, and you change the map shown in the **Vmap Bar**, the new **Endomorph** will vanish. )



## Bkg to Morph

This command allows you to turn a background layer into a new **Endomorph**.

### How do I use this tool?



Assuming that the background layer has the same number of points and point order as the foreground layer, you click the button to open the dialog.

Enter a name for your new morph in the **EndoMorph Name** field, or choose an existing name from the drop down menu. The menu contains all the names of all the **Morph Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them. If you choose the name of an existing morph for this object, the new one will replace the old one, which is similar to overwriting any other kind of file.

It's suggested that you name your morph using the group.pose format; like Eyes.Open or Mouth.Smile. That will cause the morph to appear on the same tab with all the other poses in that group when you use **Morph Mixer** in **Layout**.

Tap **Enter** or click **OK**, and you'll have another morph!

This tool is best used for changes made to the base geometry in other places; for instance, when you have saved an object with deformations using **Save Transformed Object** in **Layout** (**File > Save > Save Transformed Object**). It can also be useful when you need to see the object while making the morph. Simply copy the object into a new layer, make the changes, then use this command to create the morph. (Don't worry about any other morphs on the copy, they won't affect the final outcome.)

## Apply Morph

This command allows you to add a specified percentage of a morph to any other morph target or to the Base object.

### How do I use this tool?



Choose the morph target you would like to modify from the **Vmap Bar**, or create a new one, and click the command. A dialog box will open.

Choose the morph you want to copy from the drop down list. The list contains all the names of all the **Morph Maps** used during the current session, whether or not the objects remain open, and whether or not the maps have any points assigned to them. (But you won't get anything if you don't choose a valid morph, of course.)

You can decide how much to blend the morph with the geometry that's already there by entering a percentage in the **Strength** field.

Be aware that if you choose to apply a morph to the Base object, that displacement will appear in all the morphs associated with that base, including the one you used to change it.

You can also use this command to increase a morph by applying it to itself.



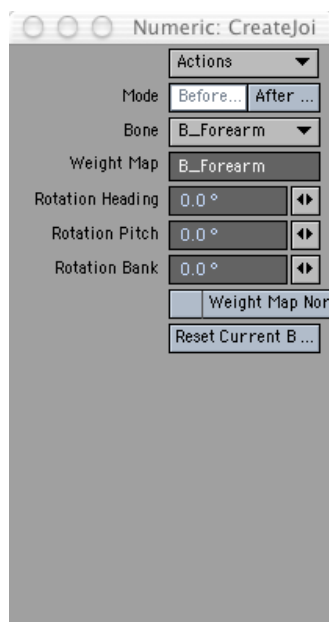
## Create Joint Morph

This tool moves the mesh as if you were rotating **Skelegons**, but without actually moving them at all. This allows you to easily make an **Endomorph** that you can use to smooth mesh deformation as you animate. (Morphing is another way to move your objects, by forcing them to conform to a different shape. The point placement for that shape is stored in a **Vertex Map**, called an **Endomorph**.)

This is something you will want to do after you already have your figure's geometry, **Skelegons** and **Weight Maps** in place.

Begin by creating an **Endomorph**. You can either do this by using the **M** button in the **Vmap Bar** and choosing (**new**), or by clicking on the **Map > Endomorph: New Endomorph** button. In either case, a dialog will appear that allows you to name the new morph. It's recommended that you give it a name that will allow you to easily identify it in Layout. Something like LeftArm.M\_90 is good; that will put the morph into the LeftArm group, and identify it as the morph that makes a bend of 90° look good.

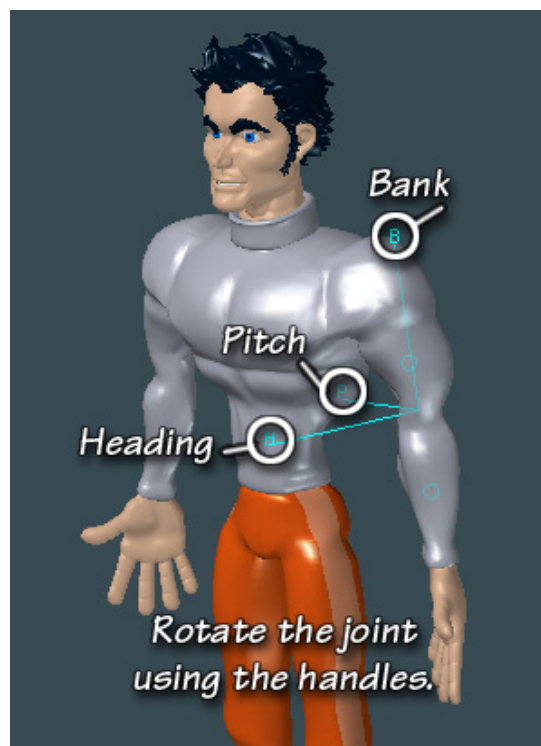
Once you have the morph created, you're ready to move the mesh. (It's a good idea to open the **Numeric Panel** for the next steps.)



Click the **CreateJointMorph** command. If you have a **Skelegon** selected, you'll see an active panel. If you don't, then simply select a **Skelegon**.

You need to start in **Before... Mode**, so make sure that button is the one that's highlighted.

Select the bone you want to work with from the **Bone** drop down menu, if it's not already selected. The **Weight Map** that is used by the bone will automatically show in the **Weight Map** field. (The field is for informational purposes only; you cannot change it.)



When you do this, you will notice that cyan rotation handles have appeared at the joint. They are labelled **H**, **P** and **B**, for **Heading**, **Pitch** and **Bank**. (**Heading** is rotation around the Y axis, **Pitch** is rotation around the X axis, and **Bank** is rotation around the Z axis. These terms should be familiar to you from Layout.)

You can move the arm by dragging these handles, or you can enter the values you want directly into the **Rotation Heading**, **Rotation Pitch**, and **Rotation Bank** fields in the **Numeric Panel**.





Adjust the joint by the desired amount and direction. For instance, in our example, bend the arm so the pitch is  $90^\circ$ . As you do, you will notice two things. First, there are the usual problems that occur in the elbow area. Secondly, although the mesh has moved, the **Skelegon** has not!

Drop the **CreateJointMorph** tool, either by tapping the **spacebar**, or simply getting the first of the **Modify** tools you plan to use for the **Endomorph**.

Go ahead and manipulate the mesh, reworking the area until the elbow looks more like a bent elbow, and less like a piece of folded foam rubber. (You can even make the bicep bulge, if you are so inclined.)



**HINT:** It's often easier to select the points in the crease if you flip back and forth between the Base and the **Endomorph**. Any points you select in the base will remain selected in the **Endomorph**. (But move at least one point before you switch, or LightWave will think you didn't want the morph, and it will vanish.) It's probably a good idea to hide the **Skelegons**, so you don't accidentally move one of the points on them. (Select them in the **Polygon Statistics** and tap the - key to hide them.) You might want to hide everything else, too, except the arm you're working on. When you are finished, show everything that's been hidden by tapping the \ key.

When you are pleased with the results, click on **CreateJointMorph** again. Select the same bone, but this time, bend it by the same amount in the opposite direction. In our example, that would be a pitch of  $-90^\circ$ . The arm will now be in the same orientation that it was when you started.



Click the **After** button; you will probably see a bulge where you worked on the joint; but that's alright. The **Joint Morph** is finished, and ready to take into Layout. Drop the tool, and save the object.







---

## Chapter 9: Setup Tab

---

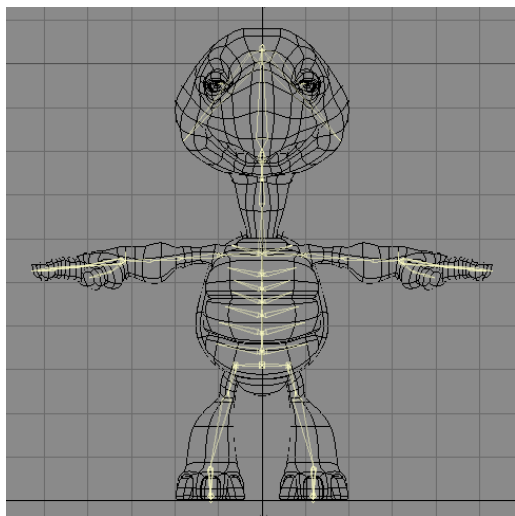


## Setup Tab

## Skelegons

### Create Skelegons

One way to create a bone hierarchy is to use LightWave's **Skelegon** feature in Modeler. **Skelegons** are a polygon type that can be added to your object in Modeler; they look like bones. You can create a complete skeletal structure out of polygons, hence the name **Skelegons**. **Skelegons** set up only placeholders for the bones. When you load an object with **Skelegons** into Layout, the **Skelegons** must first be converted to bones before you can use them for skeletal deformation.



There are several advantages to this system. First, you can use existing points in a character to create the **Skelegons**, if you wish, or you can draw your own **Skelegons** manually. Having this function as a Modeling tool makes the creation process much easier, as building a skeleton is inherently a Modeling task. Since the **Skelegons** are considered polygons, you can use any of the Modeling tools to create and modify the skeleton.

Another benefit is that the **Skelegons** are saved in the object file itself. This means that you no longer need to create a separate scene file to hold bone positions.

One of the most beneficial features of this system is that when you modify the shape of the model, the bone structure can be modified at the same time, so you do not need to make secondary modifications to that skeleton.

Imagine that you build a character and spend a lot of time creating the bone structure. The client then decides he/she would rather have the character with shorter legs and really long arms. Typically, the Modeling changes would be simple but recreating the skeletal system would be an arduous task at best. With **Skelegons**, the process becomes one simple step which only involves modifying the geometry.

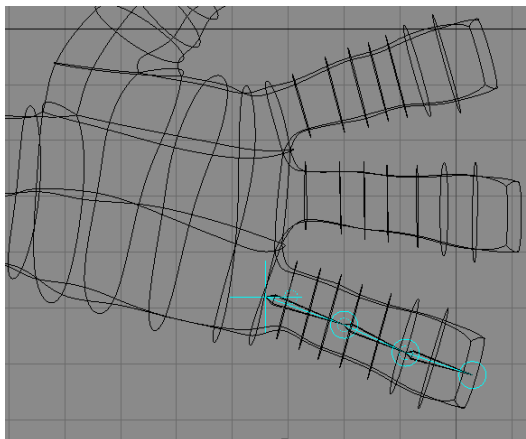
With **Skelegons**, a series of base skeletons can also be stored and used as templates. The rubber band editing style of the **Skelegons** makes it very simple to modify an existing skeleton to fit all types of characters.



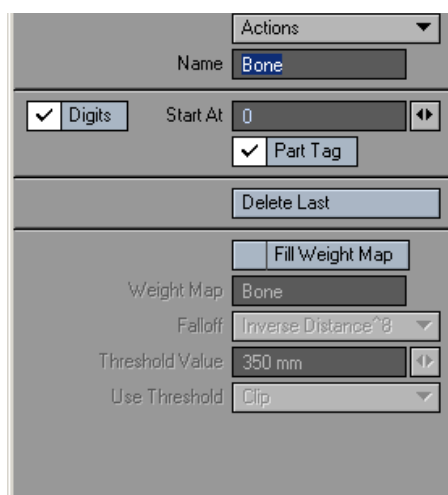
NOTE: Remember that **Skelegons** are only placeholders for bones.

### Creating Skelegons

Like drawing bones in Layout, you can use the **Create Skelegons** tool to draw **Skelegons** directly into a Modeler layer. Just choose **Setup > Skelegons: Create Skelegons**, Left Click and drag your mouse. When you release the mouse button a **Skelegon** is created. With the tool still active, you can drag your mouse again to create yet another **Skelegon** attached to the first. Select the tool again to turn it off.

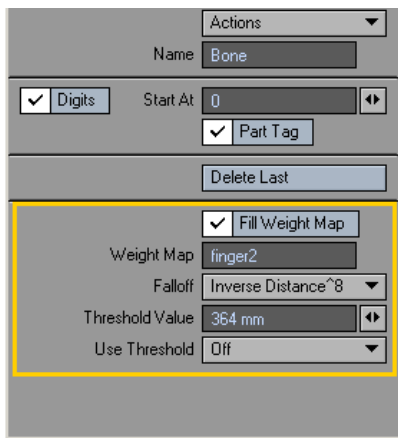


You can customise the **Name** used for each **Skelegon**, as well as the **Weight Map** name, on the **Numeric Panel**. When **Digits** is active on the **Numeric Panel**, bone names are incremented, like Bone01, Bone02, and so on. You can specify the initial number in the **Start At** field. If the **Part Tag** option is active, the bone name will also be used as a Part name automatically. Click the **Delete Last** button to delete the last **Skelegon** being created.

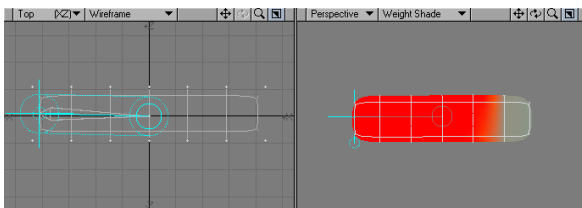




The lower portion of the dialog combines the functions of the **Bone Weights** plugin (**Map > Bone Weights**). Activate **Fill Weight Map** to gain access to these functions. Remember that **Weight Maps** are part of geometry, not **Skelegons**. Thus, you need some geometry in the foreground to use this feature. The **Weight Maps** will be automatically created for the geometry.



Note that you can also activate the **Fill Weight Map** option after you have finished drawing your **Skelegons** or even toggle it on and off. This, however, must be done prior to making the **Skelegons**, by dropping or switching the tool.



Define the **Base Weight Map** name in the **Weight Map** field. The maps will be named sequentially (e.g., Bone01, Bone02, and so on). The **Falloff** setting corresponds to the falloff that you can normally use on bones when you are not using **Weight Maps**. (See "Falloff Type and Limited Range" in the Bones Properties panel, chapter 15).

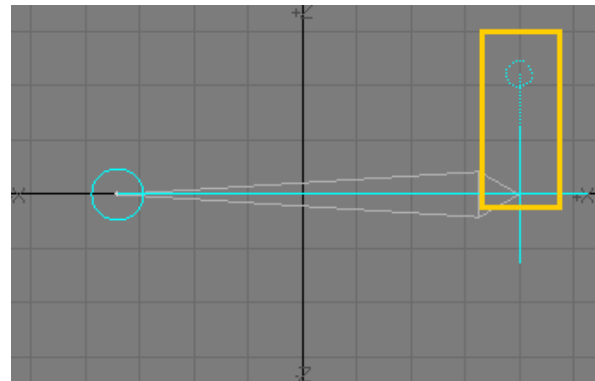
The resulting **Weight Maps** will give you a nice starting point for modifying bone influences.

The **Threshold Value** is a distance from the bone that defines a capsule-shaped region.

The **Use Threshold** pop-up has several settings. This will take the **Weight** value at the **Threshold Value** distance and either: ignore it (**Off**); set all weights outside the distance to 0 (**Clip**); subtract the threshold weight from all weights, so the weights go smoothly to 0 at the threshold, then start to become negative (**Subtract**); or subtract the **Threshold Weight** value as above, then clip the negative weights to 0 (**Blend**).

## Bank Rotation Handle

When you edit **Skelegons**, a bank rotation handle will appear. The default angle for the handle is perpendicular to the viewport the **Skelegon** is first drawn in. Drag it to change the bank rotation to apply when it's converted into a bone.



## Converting Skelegons to Bones

**Skelegons** cannot be used directly in Layout. They must be converted into bones first.

### To convert Skelegons into bones:

**Step 1:** In Layout, select the **Skelegon** object.

**Step 2:** Choose (**Setup Tab > Cvt Skelegons**). This converts the **Skelegons**, which are only placeholders, into actual bones. Note that the bones are immediately ready for action. Rest positions are set automatically and bones are activated.

**Step 3:** If your **Skelegons** were in a Modeler layer by themselves, you need to point the regular object to the newly created bones. Select the regular object and then open its **Bone Panel**. Select the **Skelegon** object in the **Use Bones from Object** pop-up menu.

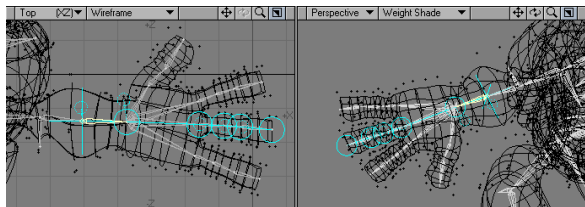


**NOTE:** If you want, you can simplify the process by cutting and pasting the **Skelegons** into the same layer as the object before saving. Then you can avoid the **Use Bones from Object** feature.



## Edit Skelegons

You can edit existing **Skelegons** (chains) by simply selecting the lowest **Skelegon** in the chain you wish to edit. Then, choose **Setup > Skelegons: Edit Skelegons**. You can also edit **Skelegons** along with an object's normal polygons by selecting them at the same time.



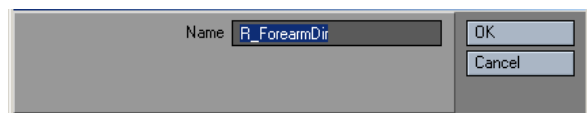
If you want the **Skelegons** to use the same points as your object, copy the points you want to use as beginning and end points for the **Skelegons** to an empty layer. Use the **Make Skelegons** command. Then cut and paste the **Skelegons** into the normal object layer. Finally, merge points.

## Changing Skelegon Direction

The **Flip** command (**Detail > Flip** or **F**) will flip the direction of selected **Skelegons**. (This command also flips the direction of a polygon's normal.)

## Rename Skelegon

You can rename a **Skelegon** by selecting it in the viewport and then executing the **Rename Skelegon** command. (**Setup > Skelegons: Rename Skelegon**)



## Split Skelegon

You can split selected **Skelegons** in two using the **Split Skelegon** command (**Setup > Skelegons: Split Skelegon**).



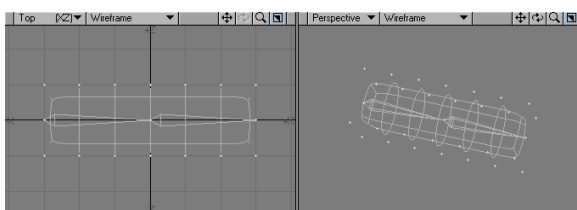
NOTE: You are not limited to applying this operation to just one **Skelegon** at a time.

## Rotate Skelegons

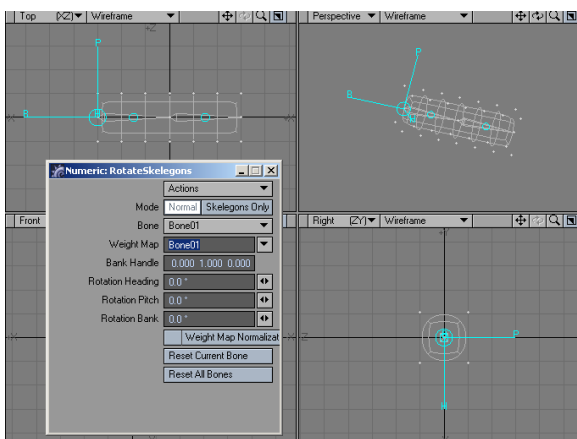
**Rotate Skelegons** (**Setup > Skelegons: Rotate Skelegons**) allows you to quickly adjust the bank handles of **Skelegons** as well as preview rotational deformations. This includes the deformation of sub-patches. This is a great tool to test your **Weight Maps** and deformations in Modeler.

### Using Rotate Skelegons

Load a model which contains **Skelegons** and make sure that both the **Skelegons** and the mesh are in the primary layer (the primary layer is the lowest numbered active foreground layer). Make sure that the **Skelegons** are not hidden, but it doesn't matter whether any points or polygons are selected.

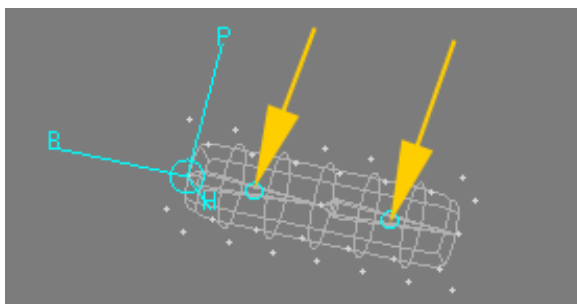


Now click on **Rotate Skelegons** and open up the **Numeric Panel (N)**



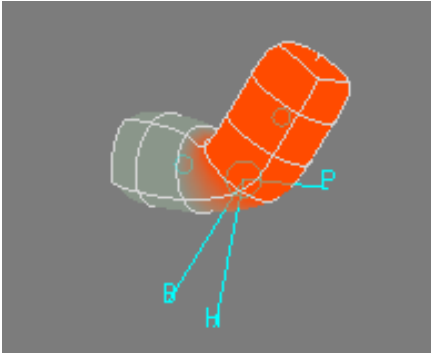
The first entry of the **Numeric Panel** is the **Mode** options. **Normal** will allow you to rotate the **Skelegons** as well as the mesh. Choosing the **Skelegons Only** option will move the **Skelegons** without moving the mesh.

The second entry is **Bone**. **Bone** is the **Skelegon** that is currently active. You can select other **Skelegons** by selecting from the drop down list or by clicking onto the small blue circles in the middle of each **Skelegon**.



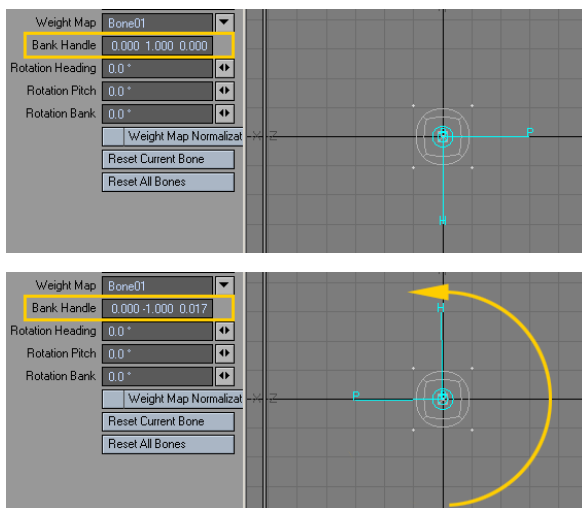


The third entry is **Weight Map**. **Weight Map** is the name of the **Weight Map** assigned to the **Skelegon**. The currently selected **Skelegon** has three lines with the letters H, P and B as well as a bigger circle attached to its stubby end. Holding the mouse down over one of the letters and dragging to the left or right rotates the **Skelegon** and its children and (if the associated **Weight Maps** contain non-zero values) deforms the mesh.



While you are in **Normal Mode**, you can have multiple **Skelegons** rotated around any axes. You can rotate the **Skelegons** also by changing the corresponding values in the **Numeric Panel**.

The **Bank Handle** (sometimes also the term **Pitch Plane** is used) determines how the rotational axes are aligned when the model is loaded into Layout and the pivot rotation for every bone is recorded. You can align the **Bank Handle** with the tool by holding the mouse down over the stubby end of the selected bone (inside the bigger circle) and dragging to the left or right. The **Heading** and **Pitch** axes will circle around the bone, i.e. these axes are now aligned differently. The **Bank Handle** alignment will be shown in the **Numeric Panel** as well.



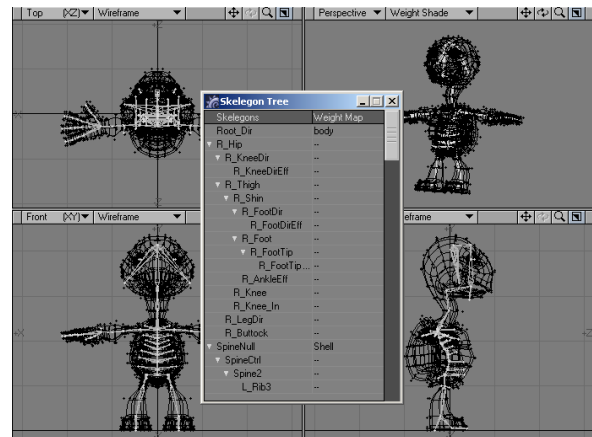
If you click the button **Reset Current Bone** in the **Numeric Panel**, the **Skelegon** will be reset to its rest position (i.e. the position it had when the tool was activated). The value of the **Bank Handle** will be reset as well.

If you click **Reset All Bones**, all **Skelegons** will be reset. If you hit the **SPACE** bar or the **RETURN** key the rotational changes will be assigned to the mesh and the tool will be deactivated.

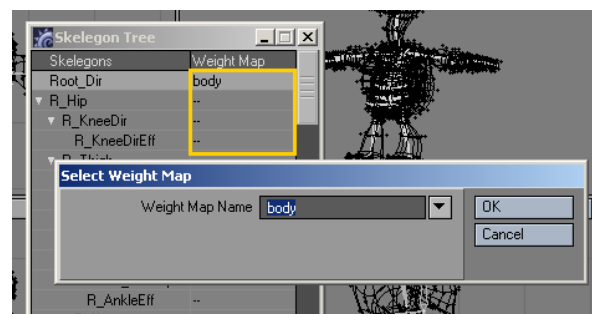
If you hit Modeler's **UNDO** button or click in an inactive toolbar area all **Skelegons** will be reset and the tool handles disappear, but the tool is still active. By simply clicking into one of Modeler's viewports the handles will become visible again.

## Skelegon Tree Panel

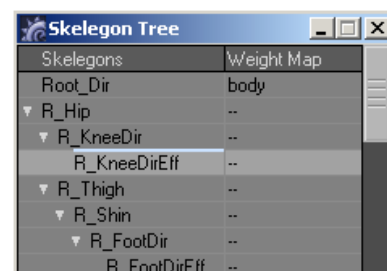
If you open the **Skelegon Tree Panel** (**Setup > Skelegons**: **Skelegon Tree**), you can view the hierarchy of **Skelegons**. **Skelegons** will be automatically parented if, for example, **Draw Skelegons** is used.



The **Weight Map** column lists associated **Weight Map** names. Double-click in the **Weight Map** column to specify a **Weight Map**.



You can also change the hierarchy by dragging and dropping individual **Skelegons** in the list.



The same points will be used, but the connections will change to reflect the revised hierarchy.

You can rename a **Skelegon** by double-clicking its name in the list and entering the new name in the dialog that appears. The names you assign become bone names when the **Skelegons** are converted to bones in Layout.

You can also rename a **Skelegon** by selecting it in the viewport and then executing the **Skelegon Rename** command located under the **Setup Tab**.

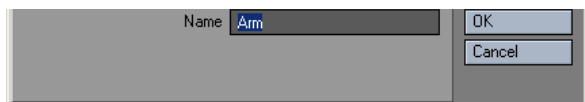


## Selection

Two commands can help you select **Skelegons**: select **Child Skelegon** (**Setup > Skelegons: Selection > Select Child Skelegon**), which selects the child of the currently selected **Skelegon**, and select **Parent Skelegon** (**Setup > Skelegons: Selection > Select Parent Skelegon**), which selects the parent of the currently selected **Skelegon**.

## Skelegon Weight

Set **Skelegon Weight** command (**Setup > Skelegons: More > Skelegon Weight**) will set up the **Skelegons** association to map names. You still need to create and edit the weighting values as you would normally. The associated map names are automatically assigned to the bones and will appear as the **Bone Weight Map** setting on the **Bones Panel**.



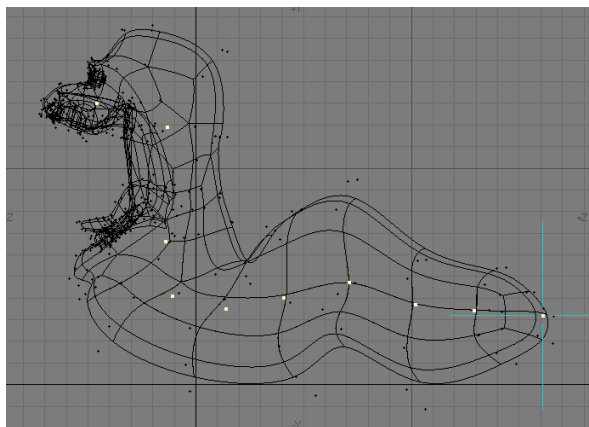
NOTE: The **Skelegon Tree Panel** will list map assignments.

## Convert Skelegons

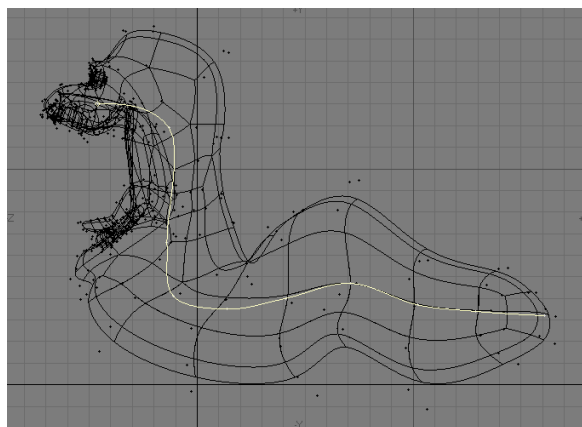
One way to create **Skelegons** is to create curves first. You can then convert the curves into **Skelegons**, which will be placed between the curve knots (points). With **Convert Skelegons**, (**Setup > Skelegons: More > Convert Skelegons**) you can even use points from the object to create the curves.

### To use Convert Skelegons:

**Step 1:** In an open Modeler layer, create points that will correspond to the bases and tips of the bones you want to eventually create. (You'll probably want the object to be boned in the background for reference.)

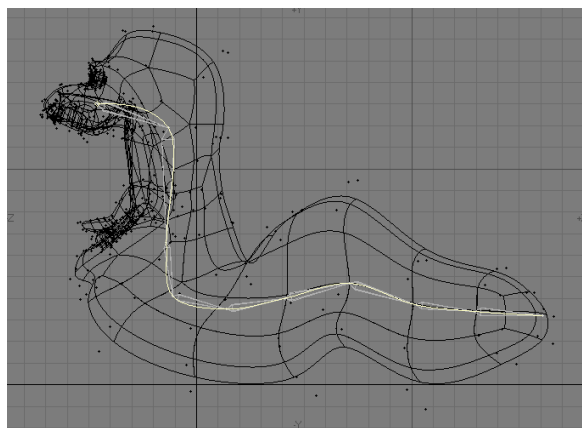


**Step 2:** Select points, working from the (theoretical) base of the first child bone to the tip of the last bone in the chain. Then choose **Create > Make Curve > Make Open Curve** to create an open-ended curve with your points.



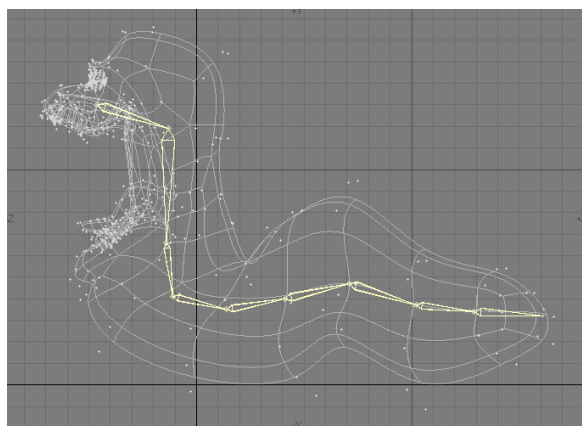
**Step 3:** Repeat the previous step for all chains, if you have more than one. You can use the same point in multiple chains.

**Step 4:** Choose **Setup > Skelegons: More > Convert Skelegons** to create the skelegon from the curves. You can edit the base/tip of the bones using standard modeling tools. The **Skelegon** bones will stay attached to each other.



If you desire, you can select and delete the curve afterward.

**Step 5:** If you want, you can optionally cut and paste the **Skelegons** into the layer with the normal object geometry.



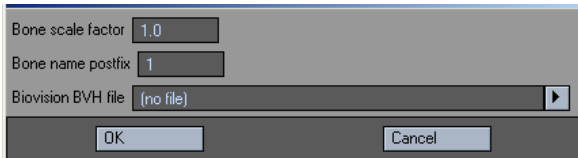




## Mocap Skelegons

LightWave provides a couple of plugins to support the BioVision (BVH) motion capture file format. The MoCap\_BVH\_Setup generic Layout plugin reads a BioVision BVH file, creates bones, and applies the motion capture data to them.

**MoCap Skelegons (Setup> Skelegons: More > Mocap Skelegons)** is a Modeler plugin that creates **Skelegons** in Modeler that match the initial rest position of the Biovision BVH data. Use it to determine the correct scale, position, etc. for your object mesh. This object can then be used with the bones created using the MoCap\_BVH\_Setup generic plugin in Layout.



### Bone Scale Factor

This option gives you control over the overall size of the **Skelegons** that will be created.

### Bone Name Postfix

This option will put a number after each Bone name in Layout for scenes with multiple characters. This is a setting that only functions in Layout so there is no need to change it in Modeler.

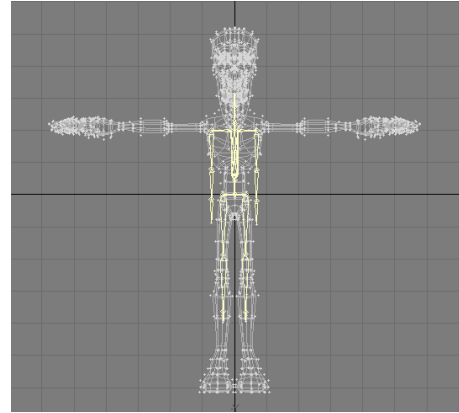
### Biovision BVH File

This field is where you will load the BVH Mocap file that will be read to create the **Skelegons**.

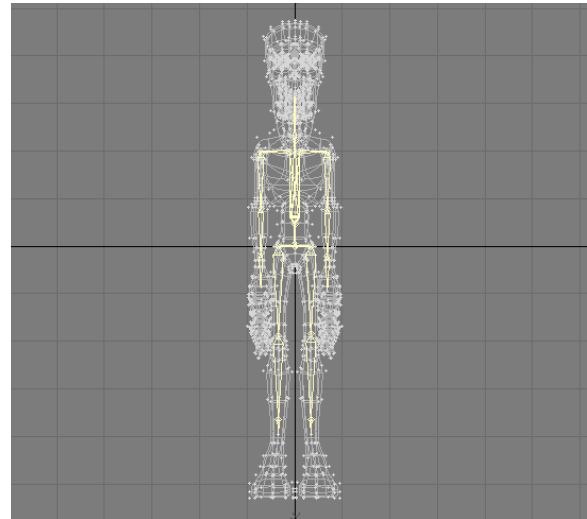
### Steps to use Mocap Skelegons:

**Step 1:** Click MoCap Skelegons (**Setup>Skelegons: Mocap Skelegons**).

**Step 2:** Click **OK**.



**Step 3:** Manipulate your model to match the **Skelegons**.



Your model is now ready to load into Layout!



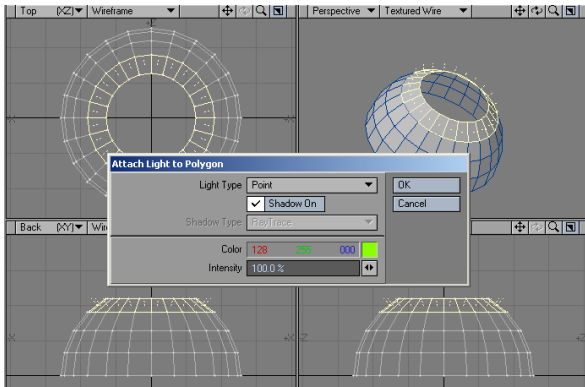
**NOTE:** Some models will need **Weight Maps** applied to assign geometry to each specific bone, and prevent distortion when using motion capture for animation.



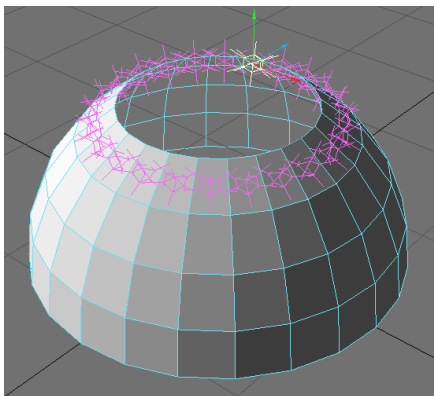
## Layout Tools

### Add Luxigon

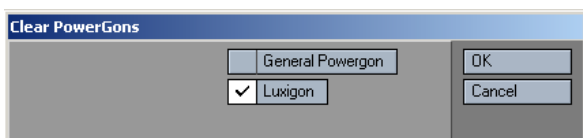
**Luxigons** (**Setup> Layout Tools: Add Luxigon**) are very similar to **Powergons** (discussed later), but are specialized just to add lights and set certain light properties. To use, in Modeler, select the desired polygons and choose **Add Luxigon**. When the dialog appears, select the type of light you wish to add and set its properties.



Next, load the object into Layout and choose **Items>Add>Lights: Convert Luxigons**. If you wish to use an existing light's properties, select it in the **Clone Item** pop-up menu. Enter the desired name of the new lights in the **Light Name** field. After you click **OK**, the defined lights will be created, positioned at the center of the appropriate polygons, aligned with the polygon's normal, and parented to the object.



**NOTE:** You can clear **Luxigons** from selected polygons using the **Setup >Layout Tools: Clear Powergons** command.

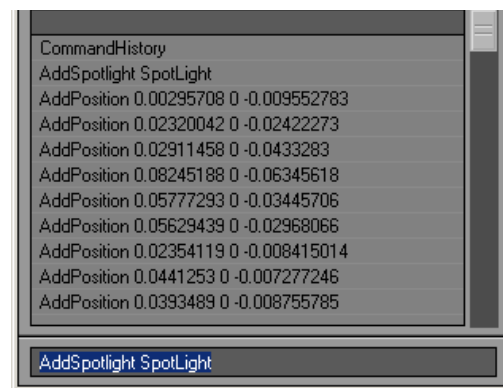


### Add Powergon

Before we discuss **Powergons**, it is important to understand Layout commands. If you have ever customised keyboard shortcuts or menus, you have touched on Layout's underlying command structure. Nearly all of your actions in Layout are executed by sending discrete commands.

The commands listed in the keyboard shortcut and menu configuration panels only show the command name, but not any parameters the command might support. For example, the **Render Options** command causes the **Render Option Panel** to display. It obviously needs no additional information to execute. The **Add Spotlight** command, on the other hand, can take an optional argument that names the added light. Thus, executing "Add Spotlight CoolLight" would add a light named CoolLight.

You can get a much better idea of what we are talking about here by choosing **Utilities>Commands: Cmd History** (In Layout). This opens the **Command History** window. Leave it open and execute some operations in Layout. You will see commands whiz by as you click buttons and drag scene items. Here, you can see not only the command names, but also any parameters. You'll also see some commands that are not listed in the keyboard shortcut and menu configuration panels, like **Position** and **Rotation**.



### What are Powergons?

**Powergons** allow you to execute a short Layout command script, which is attached to selected polygons in Modeler. You can use this feature to quickly add, say, lights using polygons for positioning. Moreover, all of the light's properties can be defined as well.

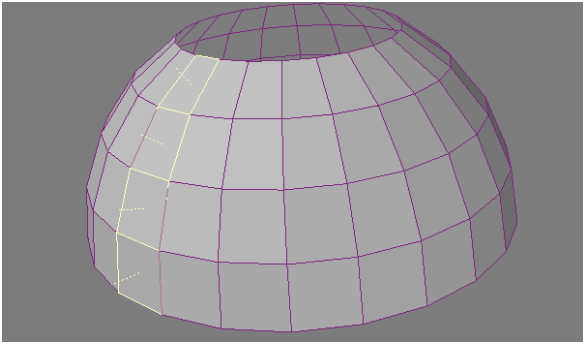


**NOTE:** Once the script is executed in Layout, you can clear the object if it is not needed otherwise. However, the **Default** command parents the lights to the object, so be careful since you don't want to delete the children.

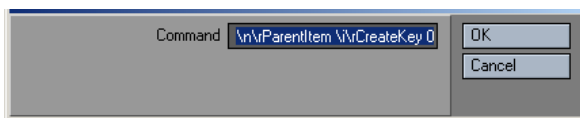


## To Attach Commands

To attach a **Powergon** command script to polygons, first select the target polygons in Modeler.



Then choose **Setup > Layout Tools: Add Powergon**. This brings up a dialog where you can enter the command script.



There are certain special “escape” characters you can use.

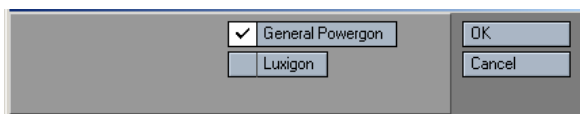
- \a** Returns the area of the polygon.
- \c** Returns the position of the polygon's center in the form “x y z”.
- \d** Returns the polygon's normal vector as “nx ny nz”.
- \i** Returns the item ID of the object containing the polygon.
- \n** Returns the polygon's normal converted into heading, pitch and bank angles (bank is always 0) in the form “h p b”.
- \r** Creates a line return to signify the end of the current command and the start of the next. Note that it is not needed at the very end of the last command.

The default command is “AddSpotlight LGON\rPosition \c\rRotation \n\nParentItem \n\nCreateKey 0”. This adds a spotlight named LGON, positions it at the center of the polygon, rotates it to match the polygon's normal, parents the light to the object, and creates a keyframe at frame 0.

Once you click **OK**, the command script is attached to the selected polygons. (Technically, this action adds a PWRG type polygon tag to the polygons.)

## To Clear Commands

To clear **Powergon** commands from polygons, simply select the polygons and choose **Setup> Layout Tools: Clear Powergons**. Activate the **General Powergon** option and click **OK**.

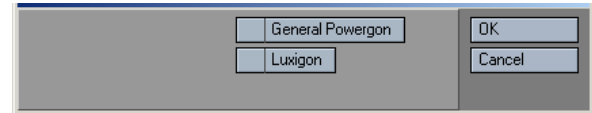


## Executing the Commands

Once the **Powergon** commands are attached, load the object into Layout. Then choose **Items> Cvt Powergons**. This executes any command scripts attached to the selected object.

## Clear Powergons

**Clear Powergons (Setup> Layout Tools: Clear Powergons)** will clear **Powergon** commands from polygons, simply select the polygons and choose **Setup>Clear Powergons**. Activate the **General Powergon** option and click **OK**. You can clear **Luxigons** from selected polygons using the **Setup>Layout Tools: Clear Powergons** command.







---

## Chapter 10: Utilities Tab

---



## Utilities Tab

## Commands

### Edit User Commands

**User Commands** are user-defined names that are associated with a particular tool/plugin. You can also provide arguments for the plugin. Once defined, they can be used as standard commands and assigned to a menu and/or key.



NOTE: You can set up multiple **User Commands** using the same plugin with different arguments.



NOTE: The command arguments can be derived from the LScript documentation available in PDF format on the LightWave 9 CDs.

#### To add a user command:

**Step 1:** Choose Utilities > Edit Commands. This will display the Edit User Commands dialog.

**Step 2:** Enter the name you want to use. We suggest you also use the prefix CMD (e.g., CMDSTRETCHPROTON). This will keep all of your defined **User Commands** together when they appear in a list together with standard commands.

**Step 3:** Select the command you want to use on the **Action** pop-up menu.

**Step 4:** If the plugin requires arguments, enter them in the **Argument** field.



NOTE: You can save a list of commands to a file by choosing **Utilities > Save Cmd List**. This file will list every action that is available as well as any arguments for the actions when applicable.

**Step 5:** Click the **Add** button. Your command will appear on the **User Command** pop-up list, along with any other **User Commands** you have created.

Click **Delete** to delete the selected **User Command**. Selecting the **Clear All** button will delete all defined **User Commands**, but don't click this unless you really mean it! There is no implicit undo for this. You will have to either re-add commands manually or use **Edit > Preferences > Revert to Startup Preferences**.

**Step 6:** Click **OK** to close the dialog. The **User Command** will now appear in any command lists.



NOTE: If you need to change the plugin associated with a **User Command**, you must delete it and then re-add it.

### Maintaining User Command Sets

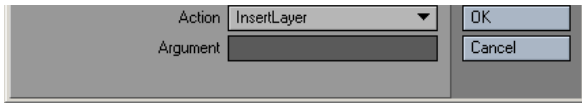
Although the **User Commands** are a part of the main preferences, the **Load** and **Save** buttons let you retrieve and store **User Command Sets** that you develop.





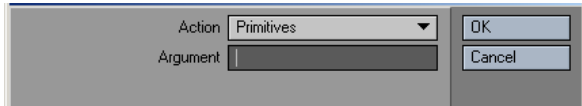
## Edit Startup

You may define any tool/plugin (with arguments) to run automatically when you first run Modeler. Simply choose **Utilities > Edit Startup** and define the command and argument, if applicable.



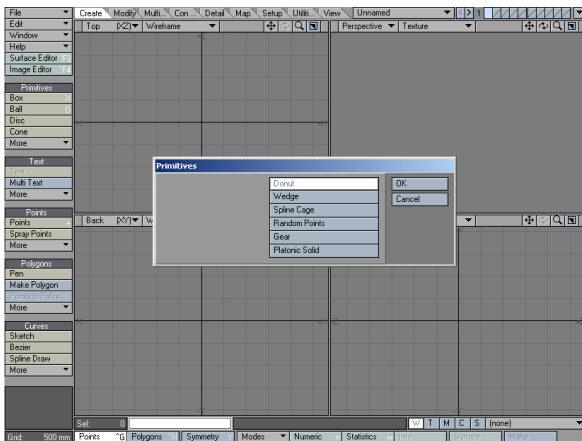
### Steps to Defining a Startup Command:

**Step 1:** Choose **Utilities > Edit Startup** and select an action.



**Step 2:** Click **OK** and close Modeler.

**Step 3:** Launch Modeler and your action should take place.



**NOTE:** To remove a **Startup Command**, select **(none)** from the **Action** drop-down menu.



**NOTE:** You can save a list of commands to a file by choosing **Utilities > Save Cmd List**. This file will list every action that is available as well as any arguments for the actions when applicable.

## Save Command List

You can save a list of commands to a file by choosing **Utilities > Commands: Save Cmd List**. This file will list every action that is available as well as any arguments for the actions when applicable.

### Example Cmd List file:

```
delete
cut
copy
paste
undo
redo
exit
new
load filename<str>
save filename<str>
close
close_all
revert filename<str>
sel_hide type<SELECTED|UNSELECTED>
sel_unhide
invert_hide
skinpols
morphpols segments<int>
togglepatches
mergepoints [range<dst>]
flip
changesurface surface<str>
change part part<str>
triple
alignpols
splitpols
mergepols
subdivide type<FLAT|SMOOTH|METAFORM>, [max-angle<ang>]
jitter radius<vec>, [type<UNIFORM|GAUSSIAN|NORMAL|
RADIAL>, center<vec>]
smooth interactions<int>, [strength<flt>]
axisdrill operation<CORE|TUNNEL|STENCIL|SLICE>,
axis<X|Y|Z>, [surface<str>]
solidrill operation<CORE|TUNNEL|STENCIL|SLICE>,
[surface<str>]
boolean operation<UNION|INTERSECT|SUBTRACT|AD
D>
pathextrude filename<str>, [step<flt>, start<flt>, end<flt>]
toggleccstart
toggleccend
make4patch u-segments<int>, v-segments<int>
railextrude segments<int>,
[interpolation<KNOTS|LENGTHS>,
flags<o=orient|s=scale>, strength<flt>]
unifypols
quantize snap<vec>
smoothcurves
removepols
freezecurves
railclone segments<int>,
[interpolation<KNOTS|LENGTHS>,
flags<o=orient|s=scale>, strength<flt>]
pathclone filename<str>, [step<flt>, start<flt>, end<flt>]
```



## LScript

### LScript

Executes an un-compiled **LScript**.



NOTE: The un-compiled **LScript** file format is (\*.ls).

**LScript** is a high-level wrapper for the LightWave plugin Application Programming Interface (API). It encapsulates the complex underpinnings of the API away from the plug-in developer, allowing them to concentrate more fully on the task to be accomplished. **LScript** also provides added features not available in the plugin API, making plugin development faster.

Because **LScript** has its roots in the C language, the transition between scripting and native-language (binary) plugin development is eased a great deal. Scripts written in **LScript** can often be ported into C with far less effort. This makes it possible to use **LScript** as a rapid prototyping tool for plugin development.

Nearly all of the LightWave plugin architectures have scripting capabilities through **LScript**.

**LScripts** can be installed in the same way plugins are. The **LScripts** then become commands that can be added to menus or assigned to keyboard shortcuts.

**LScript** also provides a run-time system, allowing scripts to be compiled into an encrypted binary form that prevents modification or reverse engineering. Facilities for timed or counted execution are also provided by the run-time system.

Most important, **LScript** is a *virtual machine* system. Scripts written on one platform should work directly and immediately on any other platform supported by LightWave. This differs from traditional plugin development in that each platform must have its own compiler, and each plugin must be compiled and maintained on that platform. **LScripts** are platform independent.

## LScript/RT (Run Time)

Executes a compiled **LScript**.



NOTE: The compiled **LScript** file format is (\*.lsc).

## LSCompiler

This command will translate an un-compiled **LScript** to a **Binary Compiled LScript**.

The LSCompiler dialog box contains the following fields and options:

- Compiled file: \model\Proton.lsc
- Use compression? Yes (selected) No
- Recovery password: [empty text field]
- Enable timeout? Yes (selected) No
- Days until expiration: 15
- Enable countdown? Yes (selected) No
- Usage count: 20
- Remind users of expiration? Yes (selected) No
- Expiration message: [empty text field]
- Embed ctimage() image data? Yes No

Buttons: OK, Cancel

## Execute Command

This command will allow an **Lscript** command to be typed into the field and executed.

The Execute Command dialog box contains the following fields and buttons:

- Command: [empty text field]

Buttons: OK, Cancel

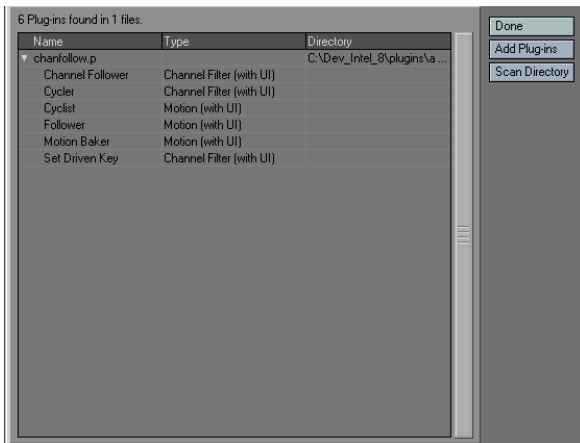


## Plugins

### Add Plugins Command

(default keyboard shortcut **F11**)

To add plugins from Modeler, choose **Utilities > Plugins: Add Plugins**. Navigate to the location of the plugins you want to add. Select the plugins, hold the **Shift** key to select an entire range or hold **Ctrl** to select multiple non-contiguous files. In Modeler, the **Add Plugins Panel** appears after you select the .p file(s).



The number of individual plugins found will appear in an informational dialog. The added plugins, their type and related .p filenames are then listed in the window.



**NOTE:** Re-adding a plugin that is already added will not do any harm.

#### To add additional plugins from the Add Plugins Panel:

Click the **Add Plugins** button and select one or more .p plugin files from a directory.



**NOTE:** A single plugin file can have many functions, some internal and not directly accessible by you.

Thus, when you add one, it may report back that it has added more than one plugin. This is normal.

#### To add a directory of plugins:

You can quickly add all of the plugins in a directory (including subdirectories) using the **Scan Directory** button. (It may take a few seconds to scan all of your plugin files. Be patient!)

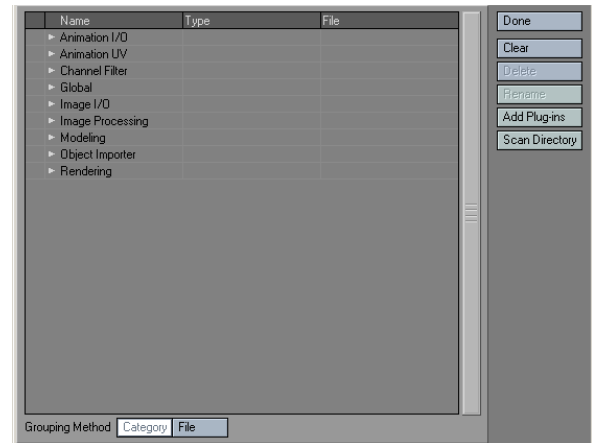


**NOTE:** You can access the **Scan Directory** feature from the **Edit Plugins Panel**.

### Edit Plugins Command

(default keyboard shortcut **Alt F11**)

Use the **Edit Plugins** command to review and delete plugins that have been added. In Modeler, choose **Utilities> Plugins: Edit Plugins**.



#### To delete a plugin command:

You can delete a plugin command by selecting it and clicking the **Delete** button. (Other commands from the plugin will continue to exist.) Click **Clear** to delete all plugins listed. These operations have no effect on the actual files stored on your hard drives.

#### To rename a plugin command:

You can rename a plugin command's name by selecting it, clicking the **Rename** button and entering a new name.



**NOTE:** You can also add plugins on this panel. The **Add Plugins** and **Scan Directory** buttons operate the same as they do for the **Add Plugins Panel**, discussed earlier.

#### File Grouping Method

If desired, you can list the **Plugin** commands by their .p plugin filenames. Just select the **File Grouping Method** button at the bottom of the **Edit Plugins Panel**.

### Additional

The **Additional** drop down menu (**Utilities> Additional**) will contain any tools added to Modeler.





---

## Chapter 11: Selection Tab

---



## Selection

### Volume Selection Mode

(default keyboard shortcut **Ctrl + J**)

You may select a portion of an object using a definable bounding box (i.e., a volume). The **Volume** (**Ctrl + J**) button is located under the **View Tab**.

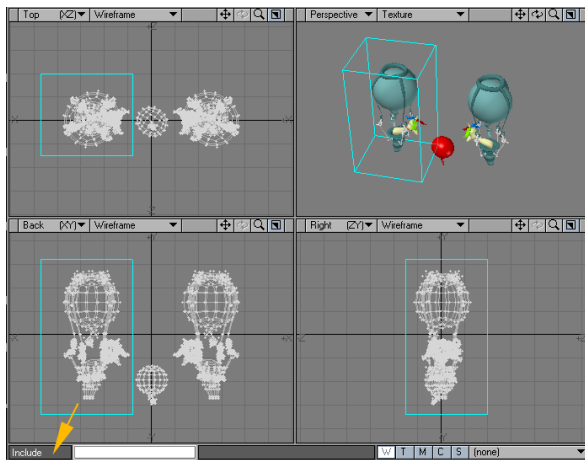
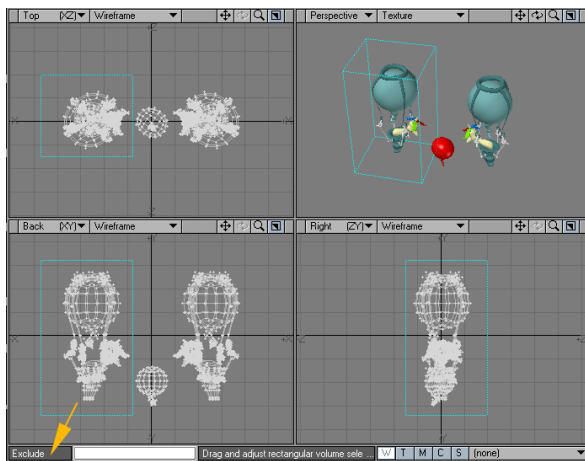


NOTE: Polygons/points selected in this manner do not become highlighted.

#### To use Volume Selection:

**Step 1:** Click the **View > Selection: Volume** button.

**Step 2:** The initial state of the **Volume Select** is usually **Exclude**. This means that polygons touching the bounding box are excluded. Clicking the **Volume** button again will toggle to the **Include** state where polygons touching the bounding box are included in the selection. You can also judge the state by looking at the small text field right above the **Points** button.



**Step 3:** Drag out a bounding box in a viewport with your **LMB**. If one already exists, you can drag it around from the center or drag the edges and corners to re-size. You can reset the bounding box by clicking in an inactive part of the interface.

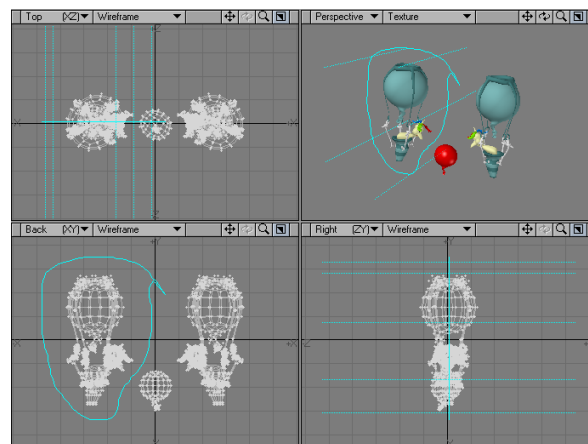
The selection is infinite along the axis perpendicular to the viewport you dragged the bounding box in.

If you open the **Volume Statistics Panel** (Open/Close by using the **Statistics** button located at the bottom of the interface), you can also use the + and - buttons to formally select/unselect the listed points/polygons.

+	-	Name	Num
+	-	Points Inside	2603
+	-	Points Outside	2741
+	-	Polygons Inside	2560
+	-	Polygons Outside	2696

#### Lasso Volume Select

You can also lasso-select a volume by dragging your **RMB** in a viewport. A big difference between this and normal volume selection is that in a **Perspective** viewport, the lassoed area is perpendicular to the viewport. Moreover, the angle will be retained even when the viewport is subsequently rotated.



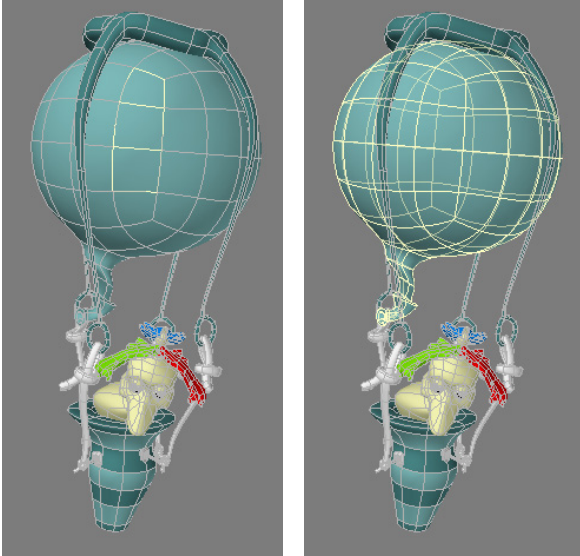
HINT: If you are selecting from among many items within a cluttered space, it's probably easier to use the **Lasso** tool for **Polygon Selection Mode** (i.e., **RMB**).



## Select Connected

(default keyboard shortcut **J**)

The **Select Connected** command (**View > Selection: Connected**) automatically selects all points or polygons that are connected to the currently highlighted points or polygons. This is a powerful tool and one you should add to your memory banks. An object that is surrounded by others may not be easily selected using a volume, and may have several surface names (making name selection tedious).



NOTE: Polygons that have overlapping points, but don't share any of those points, will not be affected. Use the **Merge Points** command

(**Detail > Points: Merge Points**) to make the edge shared.

## Drop Selection

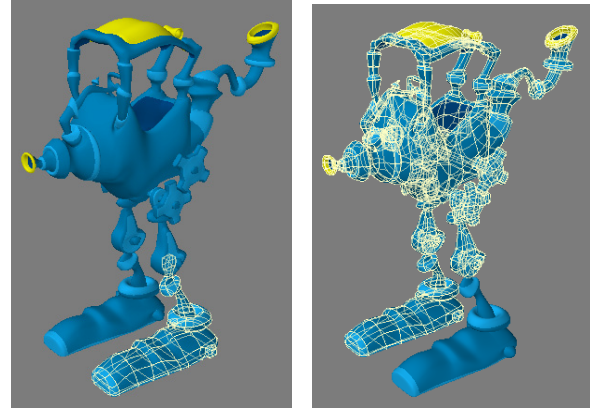
(default keyboard shortcut **/**)

Choose **Drop Selection** (**View > Selection: Drop Selection**) to unselect the currently selected points or polygons in the **Active Mode**.

## Invert Selection

(default keyboard shortcut **~**)

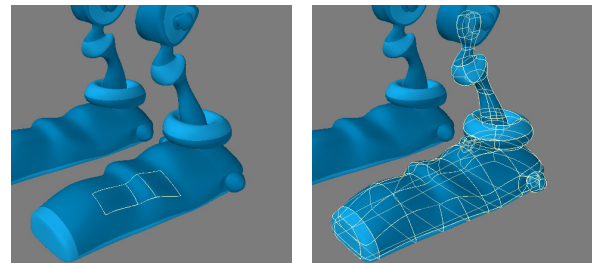
To reverse the state of selected and unselected points or polygons, simply choose **Invert Selection** (**View > Selection: Invert**).



## Invert Connected

(default keyboard shortcut **?**)

The **Select Invert Connected** command (**View > Selection: Invert Connected**) automatically selects all points or polygons that are connected to the currently highlighted points or polygons and deselects the current selection. Think of this command as a mix between **Select Connected** and **Invert Selection**.



This is a powerful tool and one you should add to your memory banks. An object that is surrounded by others may not be easily selected using a volume, and may have several surface names (making name selection tedious).

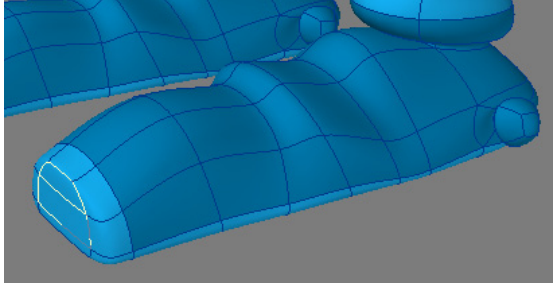




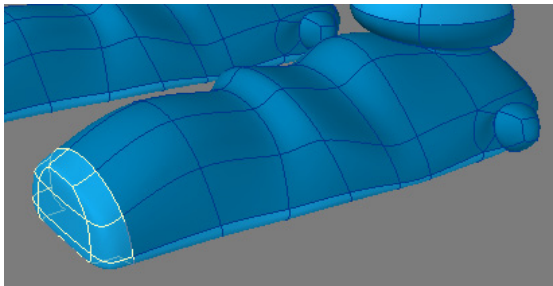
## Expand Selection

(default keyboard shortcut **J**)

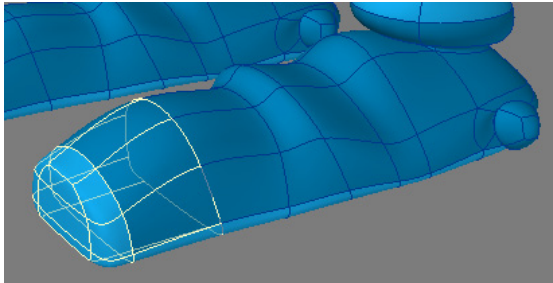
The **Expand** command (**View > Selection: Expand**) is a single-step version of the **Select Connected** command. All points/polygons that are adjacent to selected elements are themselves selected.



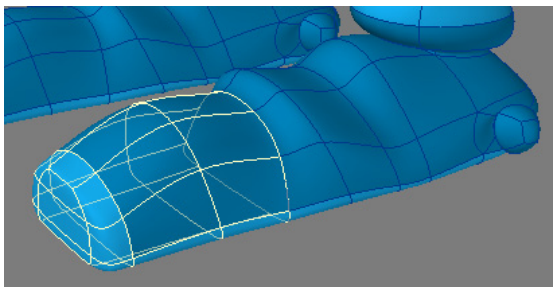
Selection



Expand Applied



Expand Applied Again

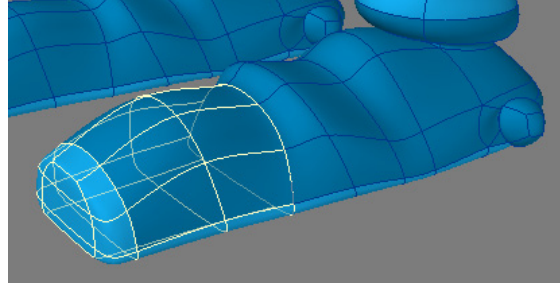


Expand Applied Again

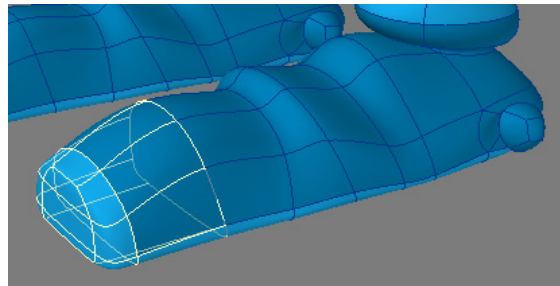
## Contract Selection

(default keyboard shortcut **L**)

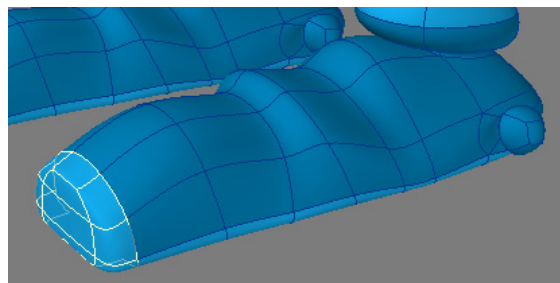
The **Contract** command (**View > Selection: Contract**) is the exact opposite of the **Expand** command, except that any elements that are adjacent to unselected elements are deselected.



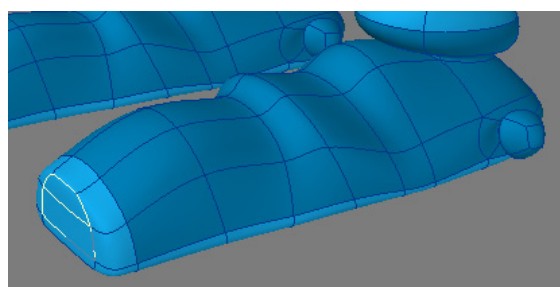
Selection,



Contract Applied



Contract Applied Again



Contract Applied Again



## Maps

There are several **VMap Selection** commands that can be found in the **View Tab** under **Maps**.

### Select by Map

The **Select by Map** command (**View > Selection: Maps > Select by Map**) selects or deselects points that have entries in the chosen **Vertex Map**.

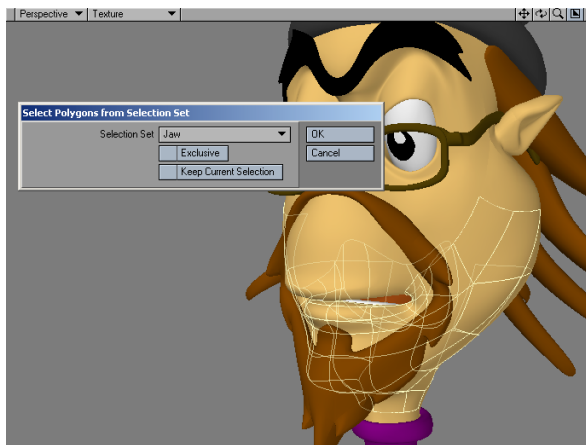


### Select by Polygon Map

You can select polygons using per-poly UVs by choosing the **Select by Poly Map** command (**View > Selection: Maps > Select by Polygon Map**).

### Select Polygons from Selection Set

The **Select Polygons from Selection Set** command (**View > Selection: Maps > Select Polygons from Selection Set**) lets you select polygons based on VMap names.



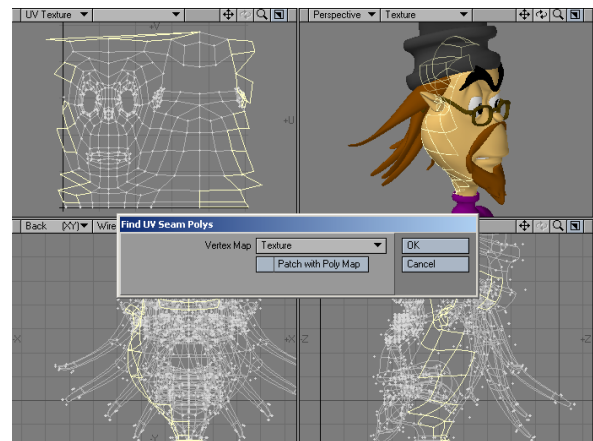
Use the command while in Polygon edit mode by simply selecting the **VMap** name in the pop-up menu and clicking **OK**. Any polygon that includes a point with that **VMap** is selected. If you select **Exclusive**, all of the polygon's points must use the **VMap** to be selected.

Normally, the new selection replaces the current selection. To add the new selection to your current selection, activate **Keep Current Selection**.

### Select UV Seam

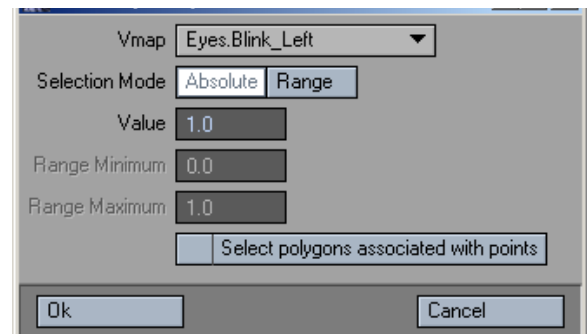
The **Select UV Seam** command (**View > Selection: Maps > Select UV Seam**) automatically selects polygons where the **UV** values span more than half the image — thus assuming that they are *seam polygons*. If you activate **Patch with Poly Map**, it will automatically turn these into per-polygon UVs.

**Select UV Seam** is limited, because it will not set **UV** values past 1.0 or less than 0. Thus, it is possible that some important portion of your image will get missed (e.g., you really needed a 1.2 U or V value).



### Select By Map Influence

The **Select By Map Influence** command (**View > Selection: Maps > Select By Map Influence**) will allow you to select points in a **Weight Map** based on their **Weight** values. To use, first select the target **Weight Map** on the **Vmap** pop-up menu. If you want to select a specific value only, enable **Absolute**. If you want to define a range, enable **Range**.



You may also select polygons associated with the selected points. This will include any polygon that uses at least one selected point.

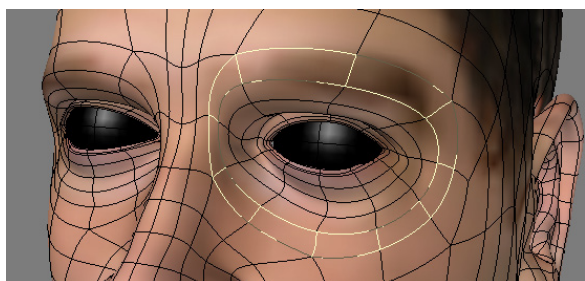
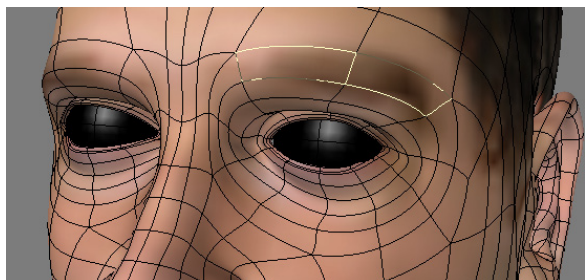


## More

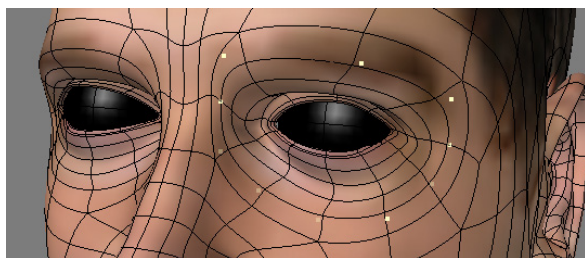
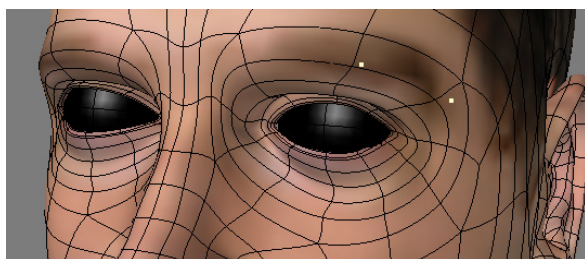
### Select Loop

**Select Loop** (**View > Selection: More > Select Loop**) is a point loop-detecting tool that works in both **Point** and **Polygon Selection Mode**. Simply select a few points or polygons and choose **Select Loop**. This tool will automatically select any edge loop which includes the current selection.

#### *Polygon Mode:*

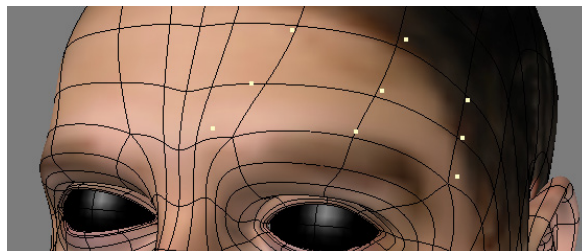
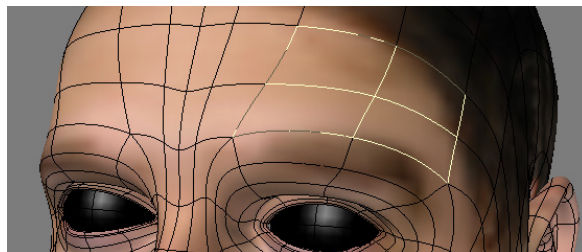


#### *Point Mode:*



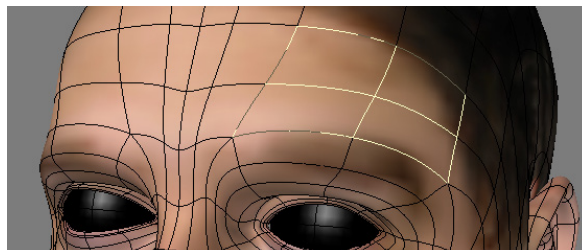
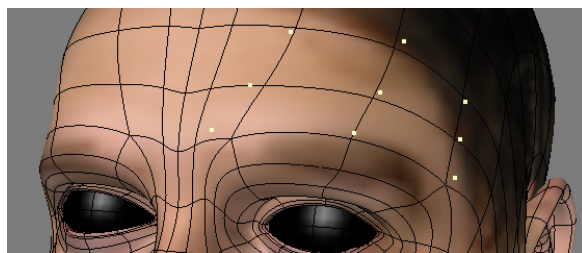
### Select Points

**Select Points** command (**View > Selection: More > Select Points**) will convert your polygon selection to points by retrieving all the points making up the selected polygons.



### Select Polygons

**Select Polygons** command (**View > Selection: More > Select Polygons**) will convert your point selection to polygons by retrieving all the polygons enclosed by the selected points.



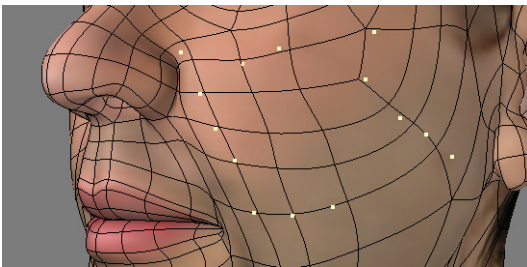
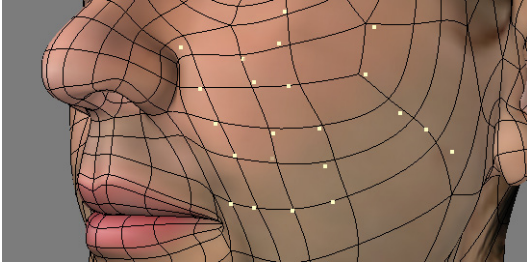




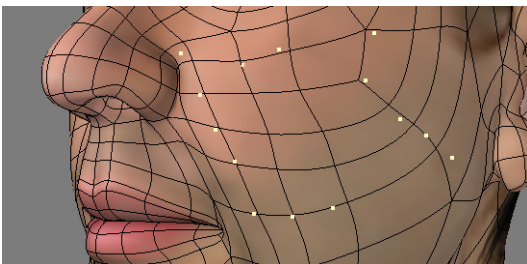
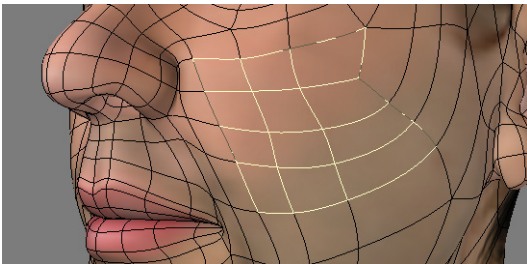
## Select Outline

The **Select Outline** command (**View > Selection: More > Select Outline**) works in both Point and Polygon Mode.

In **Point Mode**, this tool removes all the inner points in your selection and selects an outline bordering those inner points.



In **Polygon Mode**, this tool selects an outline of all the polygons you have selected, not selecting the inner points.

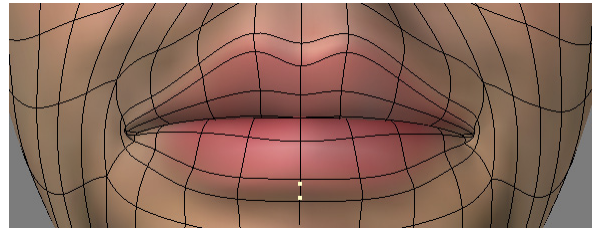


## Select Ring

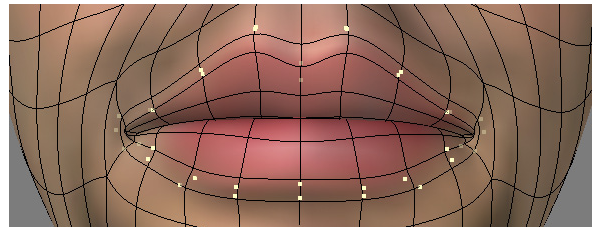
The **Select Ring** command (**View > Selection: More > Select Ring**) will select a pair of loops along two selected points.

**To Use Select Ring:**

**Step 1:** Select two points making up an edge.

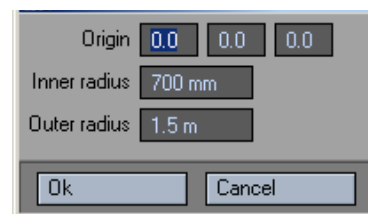


**Step 2:** Choose **Select Ring** and the tool will select a pair of loops along those two points.



## Radial Select

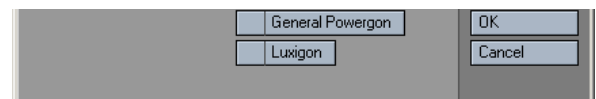
The **Radial Selection** command (**View > Selection: More > Radial Select**) will select the points that fall within a defined radius.



**Origin** is the XYZ center of the selection area. Points will be selected if they fall within the **Inner radius** and **Outer radius**.

## Select Powergons

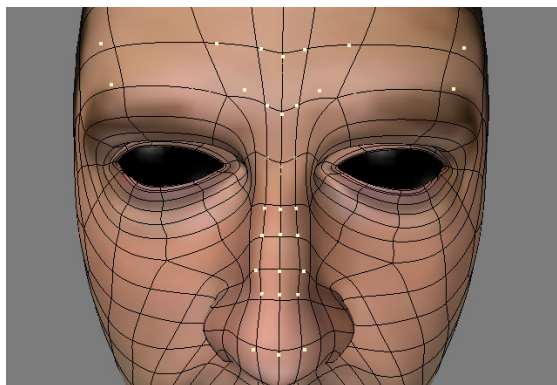
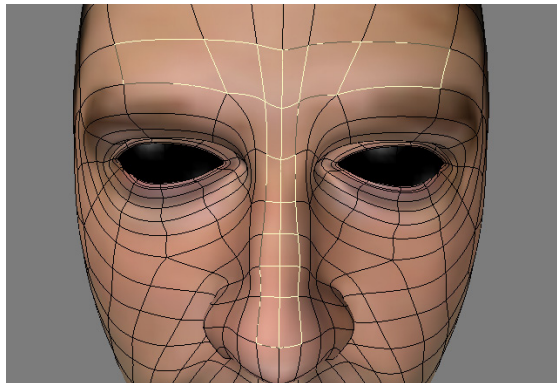
The **Select Powergon** command (**View > Selection: More > Select Powergons**) will give you the ability to select **Powergons** and **Luxigons** in an object.





## Selection Switch

Choose the **Selection Switch** command (**View > Selection: More > Sel Switch**) to convert your polygon selection to points by retrieving all the points making up the selected polygons or convert your point selection to polygons by retrieving all the polygons enclosed by the selected points.



## Selection Sets

### Selection Sets

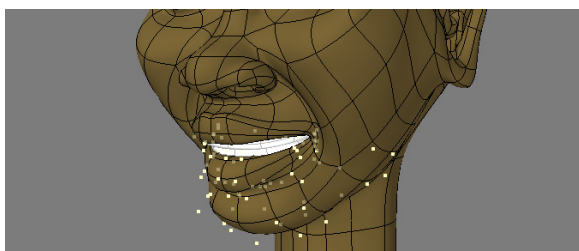
**Selection Sets** (**View > Selection Sets: Selection Sets**) will save groups of points that can be reselected by a user-defined name.



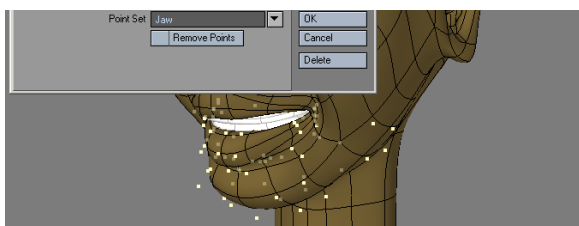
NOTE: A point can be a member of multiple sets.

**To create a point selection set:**

**Step 1:** Select a group of points.

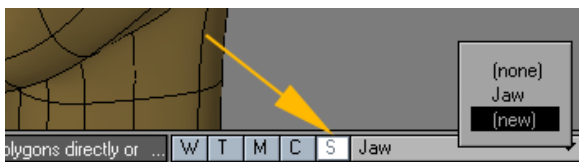


**Step 2:** Click **View > Selection Sets > Selection Sets**. Enter the desired name in the **Name** field and click **OK**.



NOTE: You can add to an existing set by selecting the additional points and then selecting the set's name from the pop-up menu.

You can also create and manage your **Selection Sets** in the **Vmap** area of the interface.



**To select points in a selection set:**



**Step 1:** In **Points Selection Mode**, open the **Point Statistics Panel** located at the bottom of Modeler's interface.

**Step 2:** Click on the pop-up menu triangle on the last line in the panel. (If it isn't visible, drag the bottom of the panel down). Select the desired selection set.

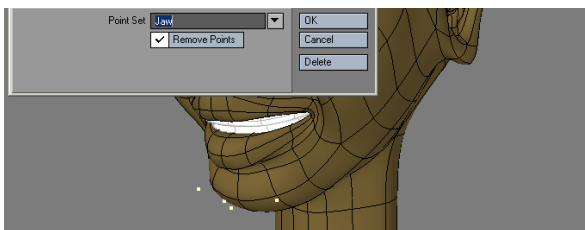
+	-	Name	Num
+	-	Total	980
+	-	0 Polygons	0
+	-	1 Polygons	4
+	-	2 Polygons	26
+	-	3 Polygons	100
+	-	4 Polygons	780
+	-	>4 Polygons	70
+	-	Jaw	81

**Step 3:** The selected set's name will appear on the line. Click the plus sign (+) to select all of the points in the set.

### To remove points from a selection set:

**Step 1:** Select the points you wish to remove.

**Step 2:** Click **View > Selection Sets > Selection Sets**. Select the set from the pop-up menu.



**Step 3:** Activate the **Remove Points** option and click **OK**.

You can delete the entire set reference by selecting it and then clicking the **Delete** button.



**NOTE:** Neither deleting operation will physically delete the points.

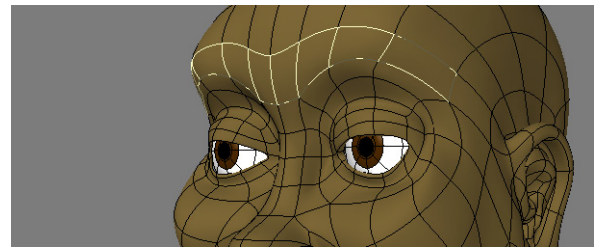
## Create Part

**Parts** are the polygon version of point selection sets. (One distinction, however, is that a polygon can be assigned to only one part name, while a point can be a member of multiple selection sets.) You can quickly select a user-defined set of polygons by

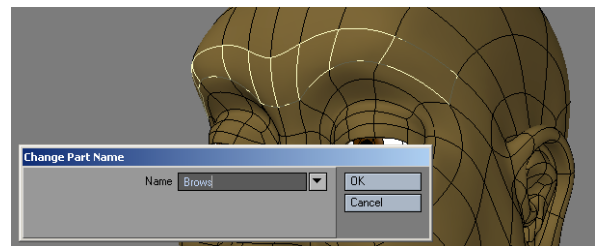
name as you would with surface names. This feature operates independently and in parallel with surface naming.

### To name polygon parts:

**Step 1:** Select a group of polygons.



**Step 2:** Click **View > Selection Sets > Create Part**. Enter the desired name in the **Name** field and click **OK**.



You can add to an existing part group by selecting the additional polygons and then selecting the group's name from the pop-up menu.

### To select parts:

**Step 1:** In **Polygon Selection Mode**, choose the **Statistics** button at the bottom of the interface to open the **Polygon Statistics Panel**.

**Step 2:** Click on the pop-up menu triangle on the last line in the panel. (If it isn't visible, drag the bottom of the panel down.) Select the desired part name.



		Name	Num
+	-	Total	964
+	-	Faces	0
+	-	Curves	0
+	-	SubPatches	964
+	-	Skelegons	0
+	-	Metaballs	0
+	-	1 Vertex	0
+	-	2 Vertices	0
+	-	3 Vertices	16
+	-	4 Vertices	948
+	-	>4 Vertices	0
+	-	Non-planar	0
+	-	Surf: (none)	0
+	-	Part: Brows	8
+	-	Col: (none)	140

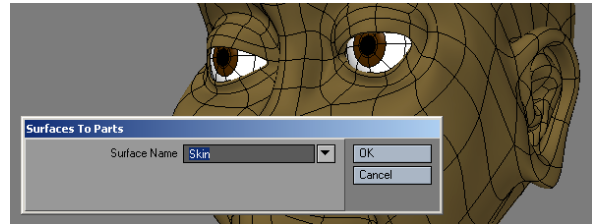
**Step 3:** The selected part's name will appear on the line. Click the plus sign (+) to select all of the polygons defined by the part.



**NOTE:** Like surfaces, any one polygon can have only one part name. Therefore, if you create a new part that uses polygons from an existing part, those polygons will become associated with the new part only.

## Surface to Parts

The **Surfaces To Parts** command (**View > Selection Sets: Surfs to Parts**) can convert surfaces to part names. The **Part** names are taken from the original **Surface** name.







---

## Chapter 12: View Tab

---



## View Tab

## Viewports

### Toggle Background Display

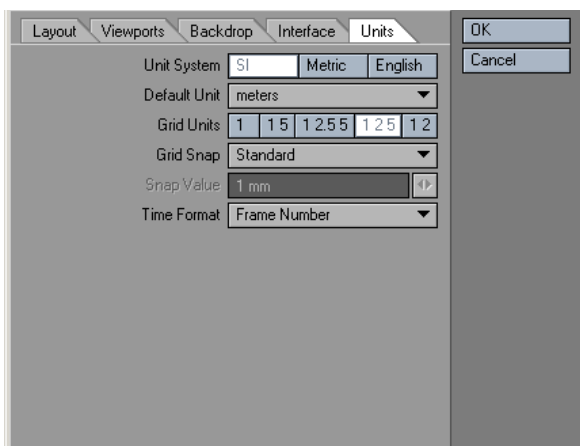
A new command for turning on/off the background image display.

### Pan

You may pan a viewport by holding the **Alt** key down and dragging a viewport. This feature is also activated with the **Pan** tool (**View > Viewports: Pan**). If the title bars are visible, you can drag the **Move** drag button to pan.



Each viewport is a window to a larger world. You can also pan the viewport beneath your mouse by using the cursor keys — one grid square for each press. Holding the **Shift** key will move the view four grid squares. Holding the **Alt** key will *nudge* by one-tenth of a grid square or the **(Fixed) Snap Value** amount on the **Units Tab** of the **Display Options Panel (Edit > Display Options)**, if that option is active. **Alt + Shift** will nudge ten times this amount.



Pressing the **G** key will center your view around the mouse pointer. This is a very quick way to manoeuvre around. *Remember this shortcut!*



NOTE: Other viewports will also change to stay in sync with the window you are changing (unless they are Center/Zoom independent).



NOTE: It may help you to think of this as moving a window around as opposed to moving the object.



## Magnify Tool

The **Magnify** tool (**View > Viewports: Magnify**) provides a way to interactively zoom in and out. This feature lets you smoothly zoom in or out of your pointer position by dragging your mouse left and right.

If the title bars are visible, you can drag the zoom drag button to zoom.



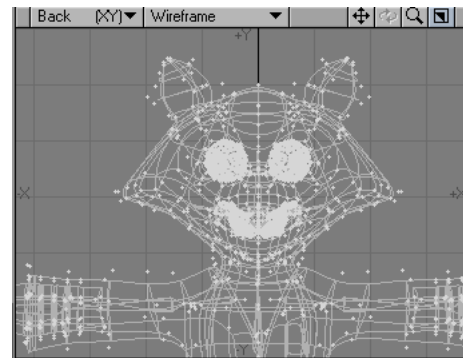
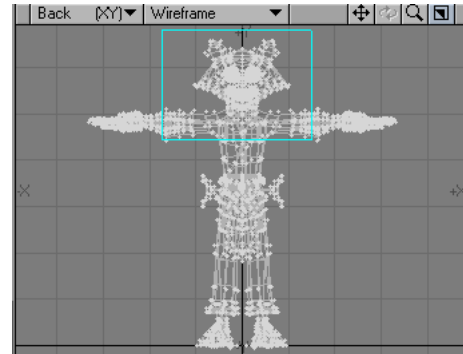
## OpenGL Display Clipping

In some cases, if you zoom in too far, OpenGL will *clip out* points and polygons — essentially causing them to disappear. Unfortunately, this is a *feature* of OpenGL. (Actually, anything that is not in the exact center of the viewport gets clipped. If you center your viewport on an item, it continues to be visible.)

## Zoom Tool

(default keyboard shortcut **Ctrl Q**)

One way to adjust magnification of a viewport is to use the **Zoom** tool (**View > Viewports: Zoom**). You can drag out a rectangular box using the mouse and the view will automatically zoom in so that the rectangular box fills the view window.



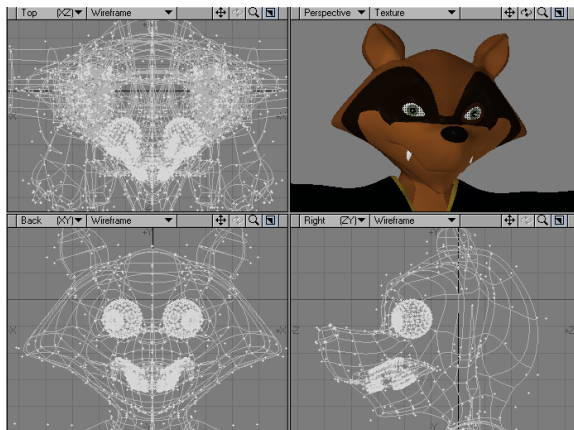


## Fit All

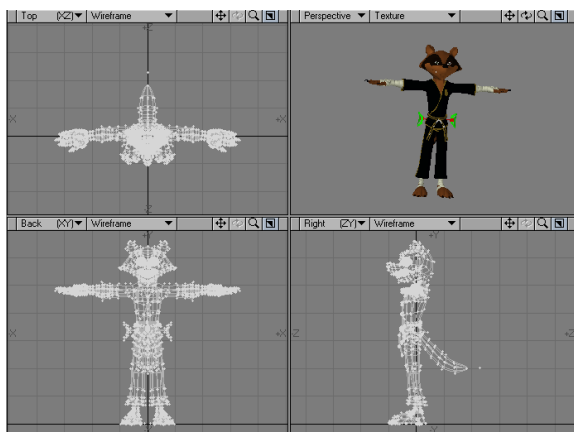
(default keyboard shortcut **A**)

### Fit All Items in Views Items in Views - Automatic Pan and Zoom

This command will automatically fit objects into the viewport. To fit the object in all viewports, use the **A** key, a shortcut for **View > Viewports: Fit All**. You'll often want to use this to *center-up* an object when it is first loaded.



Before Fit All



After Fit All

The **Fit One View** command (**Ctrl + A**) will fit the object in the viewport under the mouse pointer into that viewport.

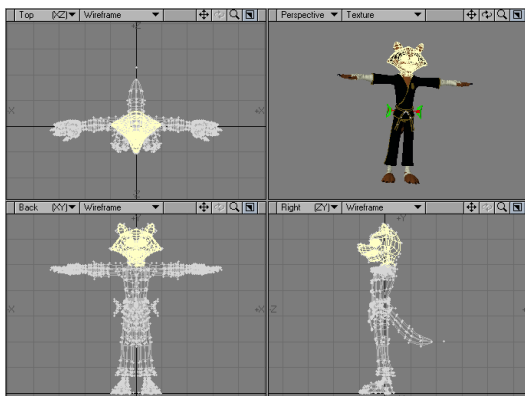


NOTE: Other viewports will also change to stay in sync with the window you are changing (unless they are Center/Zoom independent).

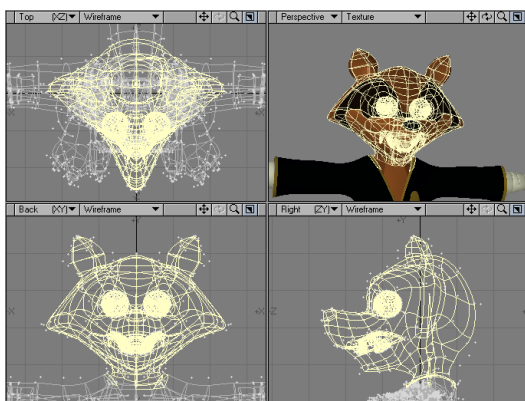
## Fit Selected

(default keyboard shortcut **Shift A**)

Choose **View > Viewports: Fit Selected** (or Press **Shift + A**) to fit only the *selected* parts of an object into all viewports.



Before Fit Selected



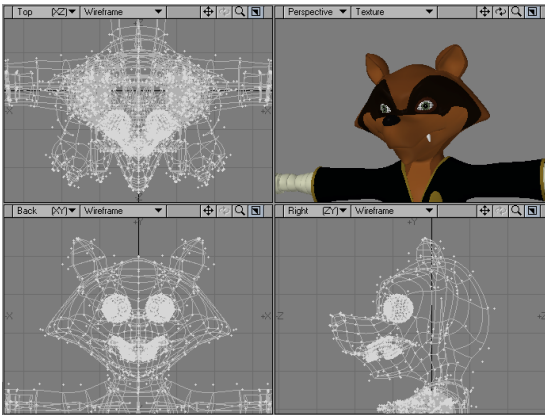
After Fit Selected



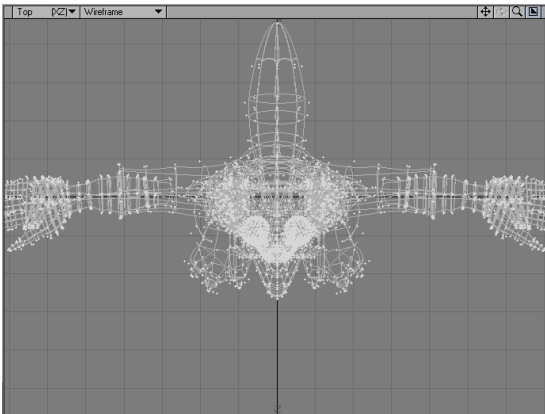
## Single View

(default keyboard shortcut **Numeric keypad 0**)

Choose **View > Viewports: Single View** to toggle between single and multiple viewports.



Single View Off



Single View On



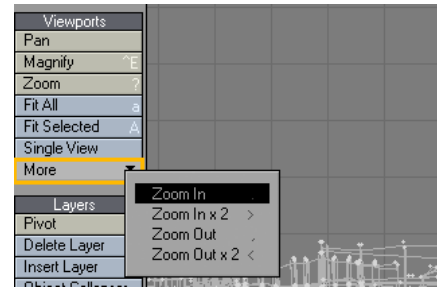
**NOTE:** The shortcut key for this function is 0 on the numeric keys. This shortcut allows you to choose the viewport you would like to have in single view. Do this by placing your cursor over the viewport you would like to use and press the **0** key on the numeric keypad.

## Zoom In/Zoom Out

(default keyboard shortcuts **<, . >**)

Modeler provides many ways to zoom in and out of your viewports. First, you can adjust viewport magnification by pressing the period key (.) to zoom in or the comma key (,) to zoom out. (These are shortcuts for **View > Viewports: More > Zoom In / Zoom Out**.) Holding the **Shift** key while pressing either key (i.e., the **>** and **<** keys, on a US keyboard) will double the zoom amount.

These tools are located under the **View Tab** in Modeler if you choose not to use the shortcut keys.

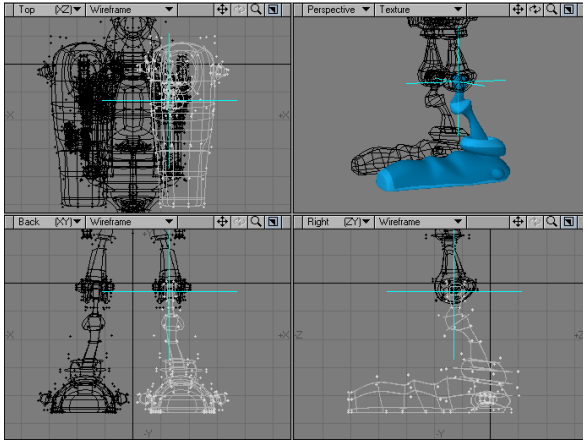




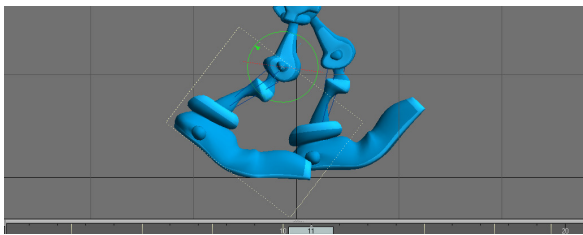
## Layers

### Pivot Point Tool

Generally, the pivot point is the positional focal point of the layer's geometry when animated in Layout. By default, it is located at the Origin. Use the **Pivot Point** tool (**View > Layers: Pivot**) to move the pivot point.



Once you have placed the Pivot in the new location for the object, press the **Enter** key. Your object's pivot will be ready to go when you load it into Layout.



### Delete Layer

(default keyboard shortcut **Home**)

**Delete Layer** (**View > Layers: Delete Layer**) deletes the selected layer and shifts the following layers up the layer list.

#### Steps for using Delete Layer:

**Step 1:** Select Layer.



**Step 2:** Choose **Delete Layer**. All layers are then shifted up the list.



### Insert Layer

(default keyboard shortcut **INS**)

**Insert Layer** (**View > Layers: Insert Layer**) inserts a layer in between two layers, shifting the layers down the layer list.

#### Steps to Insert Layer:

**Step 1:** Select the layer above which you would like to have a new layer inserted.



**Step 2:** Choose **Insert Layer** and the selected layer and all following layers will be shifted down.



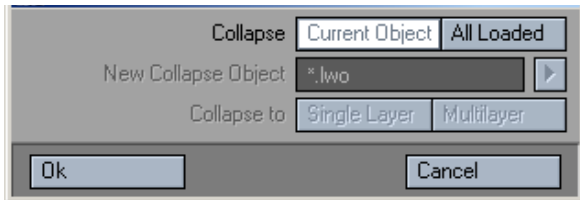




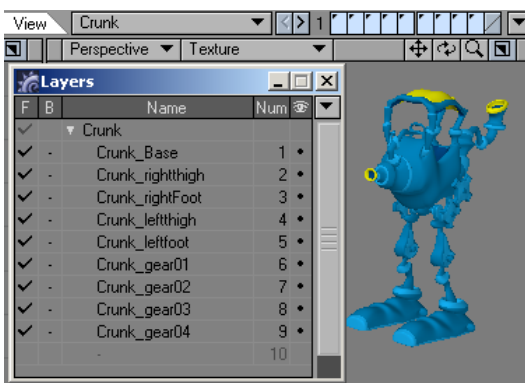
## Object Collapser

(default keyboard shortcut **END**)

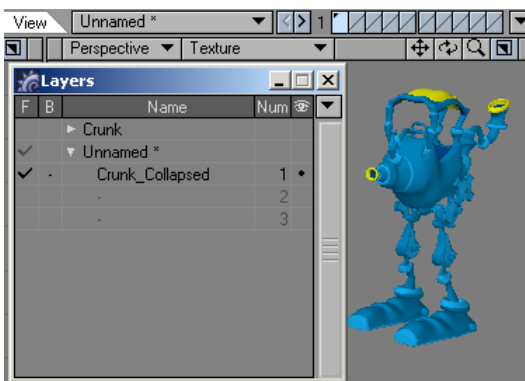
**Object Collapser (View > Layers: Object Collapser)** allows for several different ways to manipulate the Layers of an object or series of objects.



If **Current Object** is selected as the **Collapse Mode**, the current object will have all of its layers combined into the first layer of a new object. That layer will then be named after the object with a “\_Collapsed” assigned to the name.



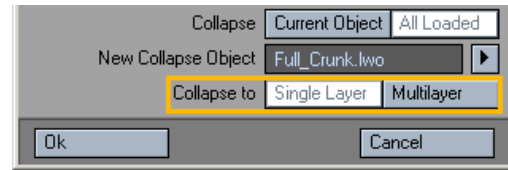
Before Object Collapser,



After Object Collapser

If **All Loaded** is selected as the **Collapse Mode**, all of the objects

loaded into Modeler will be collapsed down to one-layered objects and then combined into a single New Collapse Object.



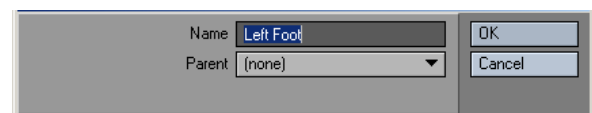
You can give the new object that will be created a name using the **New Collapse Object** name field. The **New Collapse Object** can be created in two different styles.

If **Collapse to** is set to **Single Layer**, the newly created Collapse Object will be a single layered object with all of the objects placed in that one layer.

If **Collapse to** is set to **Multilayer**, the newly created Collapse Object will be a multi-layered object with each object in Modeler placed in its own layer. Each layer of the new Collapse Object will be named after the object it represents.

## Layer Settings

Choose **View > Layers: Layer Settings** to bring up the dialog to set the layer name and parent or double-click on a layer name (*unnamed* by default) in the **Layers Panel**.



**NOTE:** Naming layers is a great way to stay organised in Modeler and will be very important for Multi-layered object in Layout. Each layer in Modeler opens as a separate object in Layout, so unique names will improve workflow.

## Swap Front and Back Layers

(default keyboard shortcut **`**)

Choose **View > Layers: Swap Layers** to swap the foreground and background layers. This is a handy command to use with tools that require background objects like **Boolean** and **Drill**.

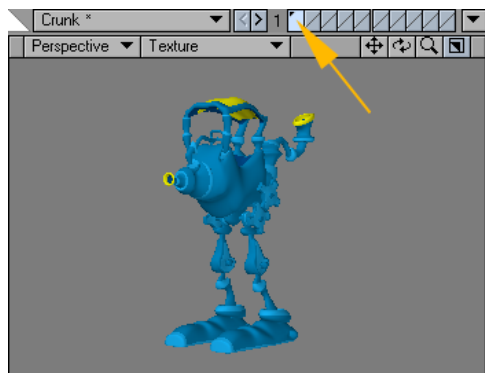
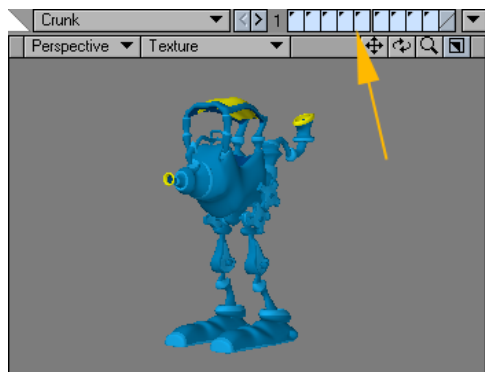


Left: Before Swap, Right: After Swap



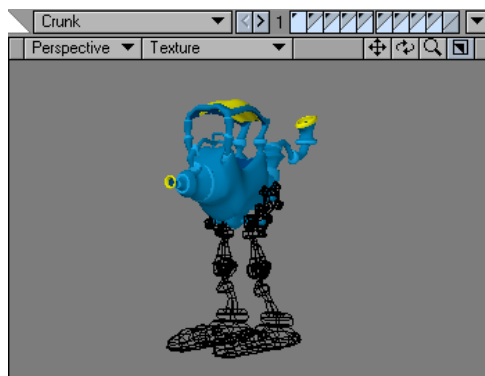
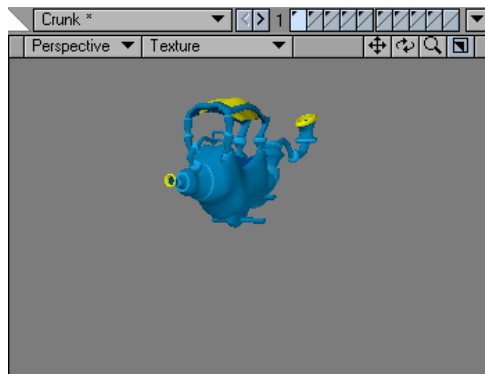
## Flatten Layers

Choose **View > Layers: Flatten Layers** to flatten multiple foreground layers into a single layer in one step.



## All BG Layers

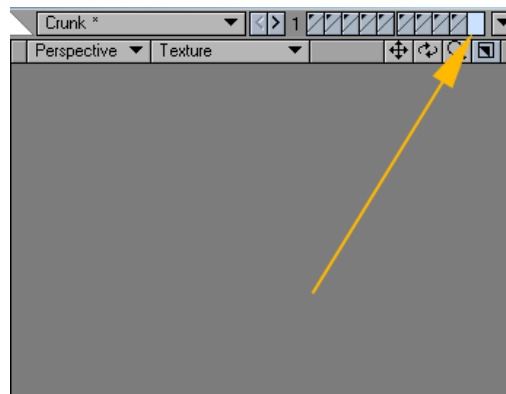
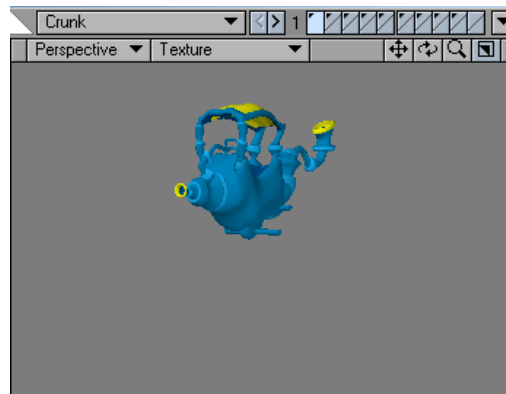
Choose **View > Layers: More > All BG Layers** to place all unselected layers in the Background.



NOTE: If this tool does not appear in your interface, perform a Scan Directory of the Legacy\_Plugins folder, using the Edit Plugins panel.

## Next Empty Layer

Choose **View > Layers: More > Next Empty Layer** to select the first empty layer.

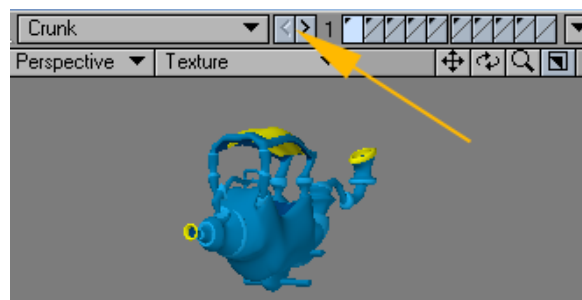


NOTE: If this tool does not appear in your interface, perform a Scan Directory of the Legacy\_Plugins folder, using the **Edit Plugins** panel.

## Shift Layers Up/Down

(default keyboard shortcuts **PgUp PgDn**)

Choosing **View > Layers: More > Shift Layers Up** or **View > More > Shift Layers Down** will change the bank of layers. The **Bank Selection** buttons are mapped to your **PAGEUP** and **PAGEDN** keys by default. The highest bank number is 99, but you can access higher layers using the **Layer Browser Panel**.





## Selection

### Volume Selection Mode

(default keyboard shortcut **Ctrl + J**)

You may select a portion of an object using a definable bounding box (i.e., a volume). The **Volume** (**Ctrl + J**) button is located under the **View Tab**.

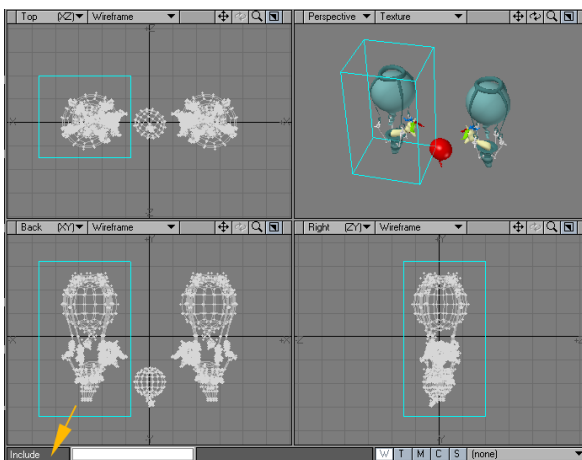
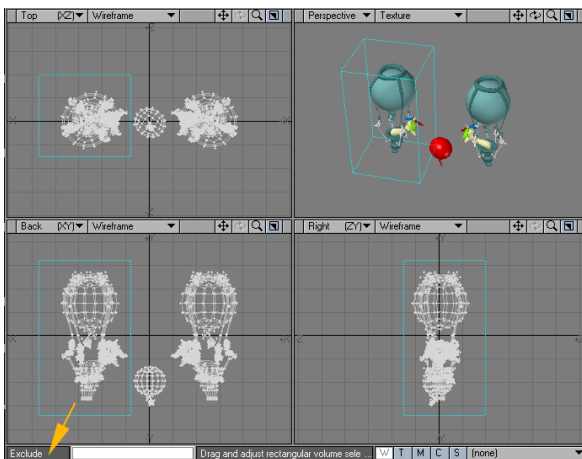


NOTE: Polygons/points selected in this manner do not become highlighted.

#### To use Volume Selection:

**Step 1:** Click the **View > Selection: Volume** button.

**Step 2:** The initial state of the **Volume Select** is usually **Exclude**. This means that polygons touching the bounding box are excluded. Clicking the **Volume** button again will toggle to the **Include** state where polygons touching the bounding box are included in the selection. You can also judge the state by looking at the small text field right above the **Points** button.



**Step 3:** Drag out a bounding box in a viewport with your **LMB**. If one already exists, you can drag it around from the center or drag the edges and corners to re-size. You can reset the bounding box by clicking in an inactive part of the interface.

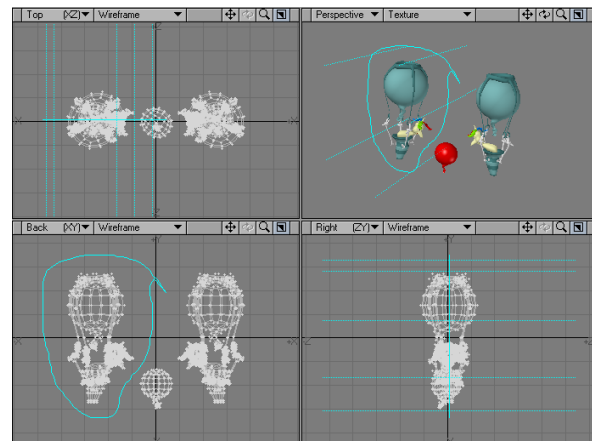
The selection is infinite along the axis perpendicular to the viewport you dragged the bounding box in.

If you open the **Volume Statistics Panel** (Open/Close by using the **Statistics** button located at the bottom of the interface), you can also use the + and - buttons to formally select/unselect the listed points/polygons.

+	-	Name	Num
+	-	Points Inside	2603
+	-	Points Outside	2741
+	-	Polygons Inside	2560
+	-	Polygons Outside	2696

#### Lasso Volume Select

You can also lasso-select a volume by dragging your **RMB** in a viewport. A big difference between this and normal volume selection is that in a **Perspective** viewport, the lassoed area is perpendicular to the viewport. Moreover, the angle will be retained even when the viewport is subsequently rotated.



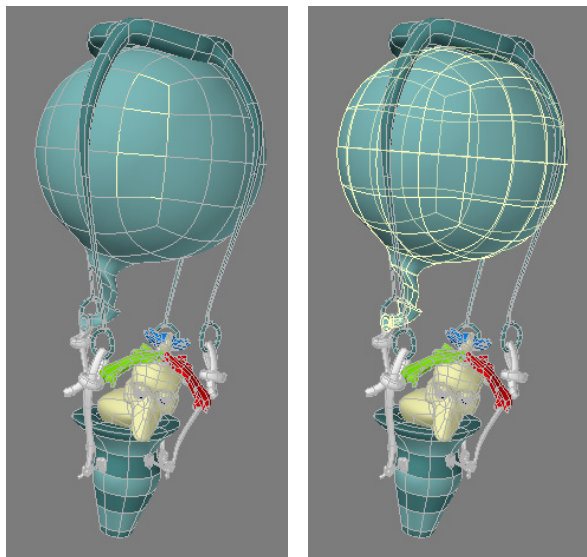
HINT: If you are selecting from among many items within a cluttered space, it's probably easier to use the **Lasso** tool for **Polygon Selection Mode** (i.e., **RMB**).



## Select Connected

(default keyboard shortcut J)

The **Select Connected** command (**View > Selection: Connected**) automatically selects all points or polygons that are connected to the currently highlighted points or polygons. This is a powerful tool and one you should add to your memory banks. An object that is surrounded by others may not be easily selected using a volume, and may have several surface names (making name selection tedious).



NOTE: Polygons that have overlapping points, but don't share any of those points, will not be affected. Use the **Merge Points** command (**Detail > Points: Merge Points**) to make the edge shared.

## Drop Selection

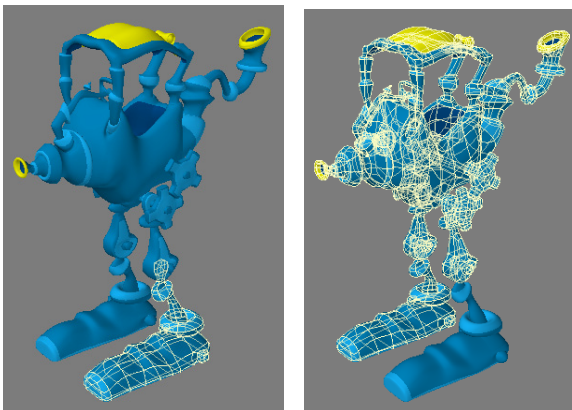
(default keyboard shortcut /)

Choose **Drop Selection** (**View > Selection: Drop Selection**) to unselect the currently selected points or polygons in the **Active Mode**.

## Invert Selection

(default keyboard shortcut ~)

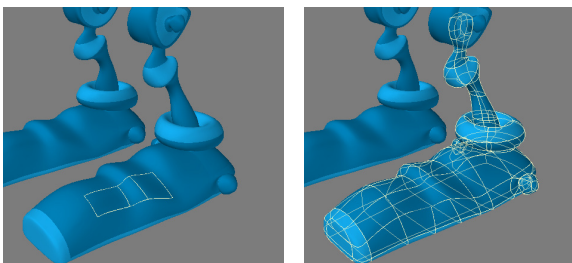
To reverse the state of selected and unselected points or polygons, simply choose **Invert Selection** (**View > Selection: Invert**).



## Invert Connected

(default keyboard shortcut ?)

The **Select Invert Connected** command (**View > Selection: Invert Connected**) automatically selects all points or polygons that are connected to the currently highlighted points or polygons and deselects the current selection. Think of this command as a mix between **Select Connected** and **Invert Selection**.



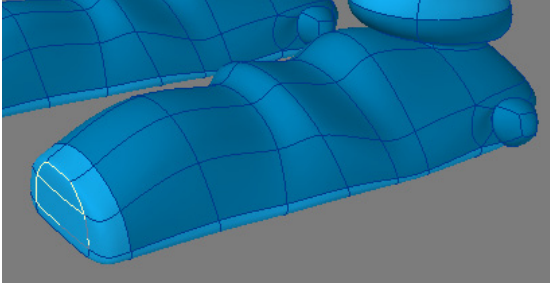
This is a powerful tool and one you should add to your memory banks. An object that is surrounded by others may not be easily selected using a volume, and may have several surface names (making name selection tedious).



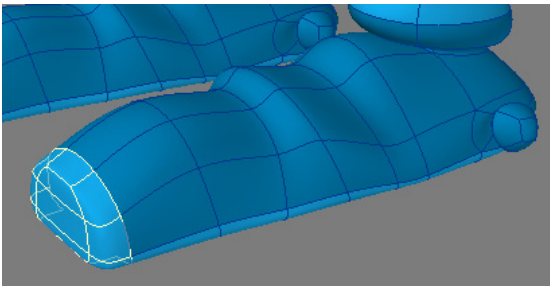
## Expand Selection

(default keyboard shortcut **J**)

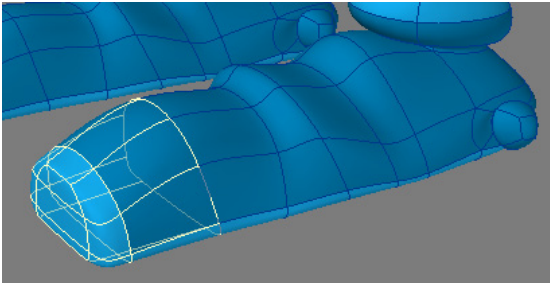
The **Expand** command (**View > Selection: Expand**) is a single-step version of the **Select Connected** command. All points/polygons that are adjacent to selected elements are themselves selected.



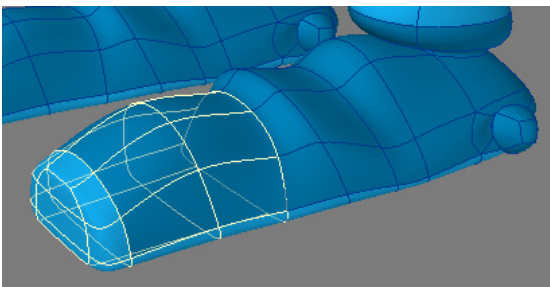
Selection



Expand Applied



Expand Applied Again

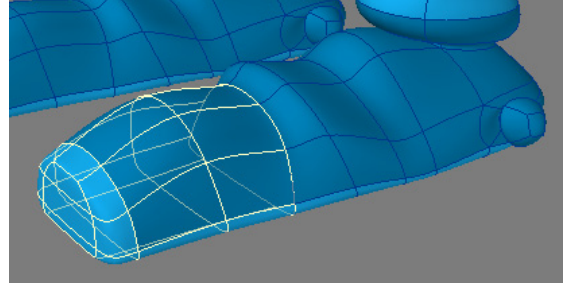


Expand Applied Again

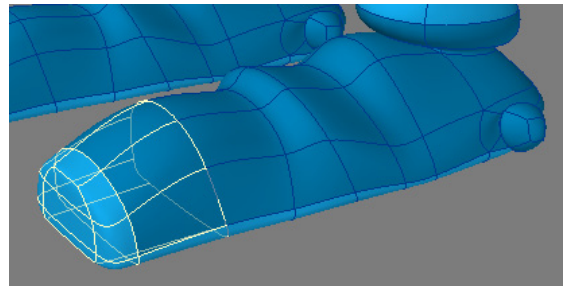
## Contract Selection

(default keyboard shortcut **I**)

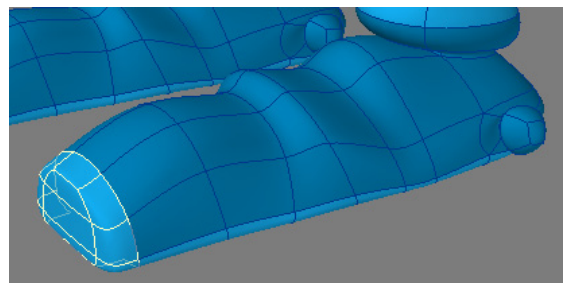
The **Contract** command (**View > Selection: Contract**) is the exact opposite of the **Expand** command, except that any elements that are adjacent to unselected elements are deselected.



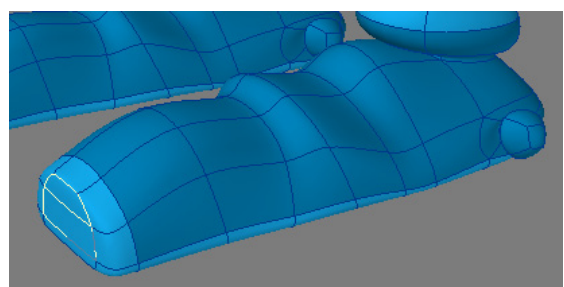
Selection,



Contract Applied



Contract Applied Again



Contract Applied Again





## Maps

There are several **VMap Selection** commands that can be found in the **View Tab** under **Maps**.

### Select by Map

The **Select by Map** command (**View > Selection: Maps > Select by Map**) selects or deselects points that have entries in the chosen **Vertex Map**.

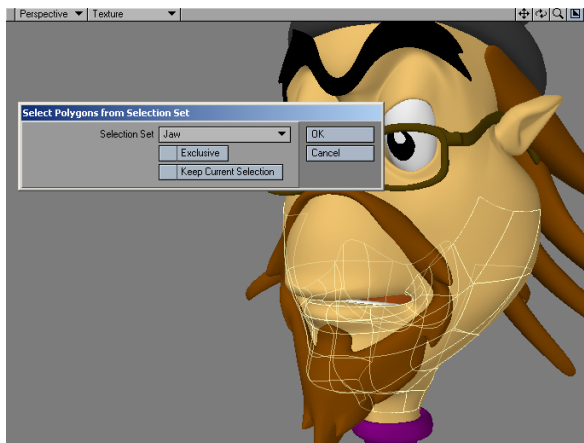


### Select by Polygon Map

You can select polygons using per-poly UVs by choosing the **Select by Poly Map** command (**View > Selection: Maps > Select by Polygon Map**).

### Select Polygons from Selection Set

The **Select Polygons from Selection Set** command (**View > Selection: Maps > Select Polygons from Selection Set**) lets you select polygons based on VMap names.



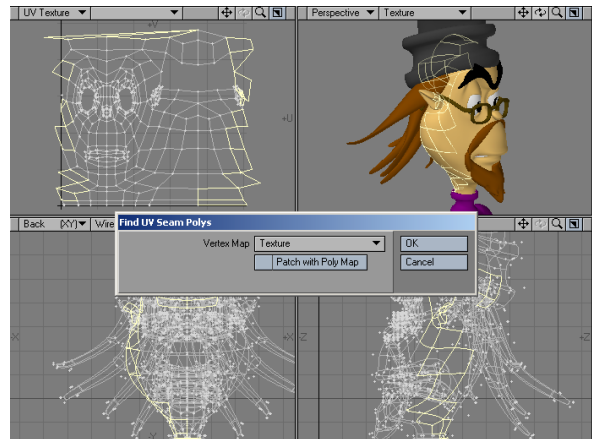
Use the command while in Polygon edit mode by simply selecting the **VMap** name in the pop-up menu and clicking **OK**. Any polygon that includes a point with that **VMap** is selected. If you select **Exclusive**, all of the polygon's points must use the **VMap** to be selected.

Normally, the new selection replaces the current selection. To add the new selection to your current selection, activate **Keep Current Selection**.

### Select UV Seam

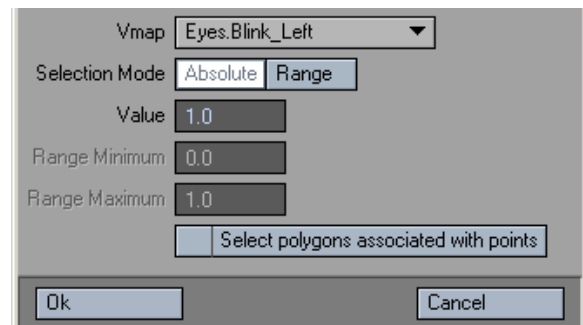
The **Select UV Seam** command (**View > Selection: Maps > Select UV Seam**) automatically selects polygons where the **UV** values span more than half the image — thus assuming that they are *seam polygons*. If you activate **Patch with Poly Map**, it will automatically turn these into per-polygon UVs.

**Select UV Seam** is limited, because it will not set **UV** values past 1.0 or less than 0. Thus, it is possible that some important portion of your image will get missed (e.g., you really needed a 1.2 U or V value).



### Select By Map Influence

The **Select By Map Influence** command (**View > Selection: Maps > Select By Map Influence**) will allow you to select points in a **Weight Map** based on their **Weight** values. To use, first select the target **Weight Map** on the **Vmap** pop-up menu. If you want to select a specific value only, enable **Absolute**. If you want to define a range, enable **Range**.



You may also select polygons associated with the selected points. This will include any polygon that uses at least one selected point.



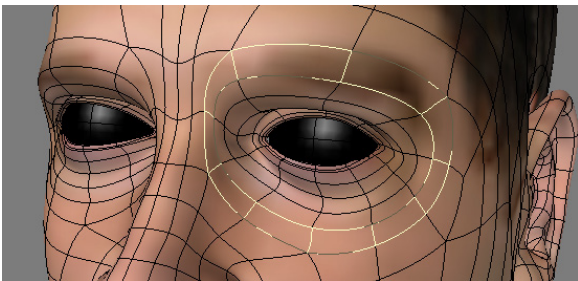
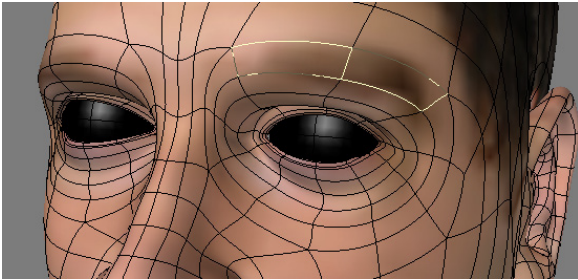


## More

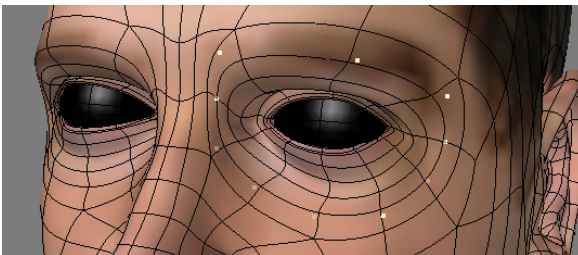
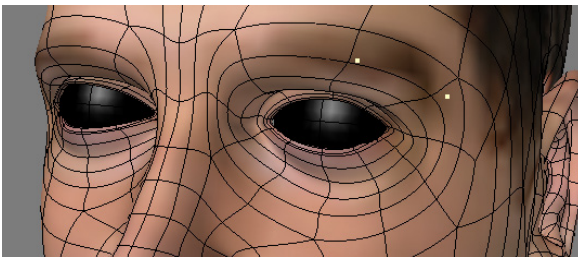
### Select Loop

**Select Loop** (**View > Selection: More > Select Loop**) is a point loop-detecting tool that works in both **Point** and **Polygon Selection Mode**. Simply select a few points or polygons and choose **Select Loop**. This tool will automatically select any edge loop which includes the current selection.

#### *Polygon Mode:*

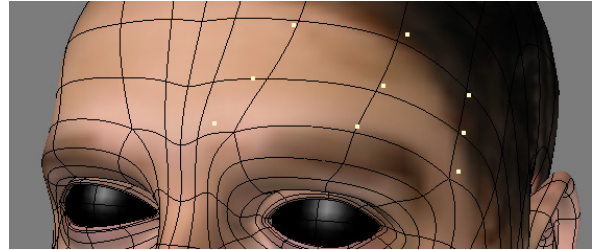
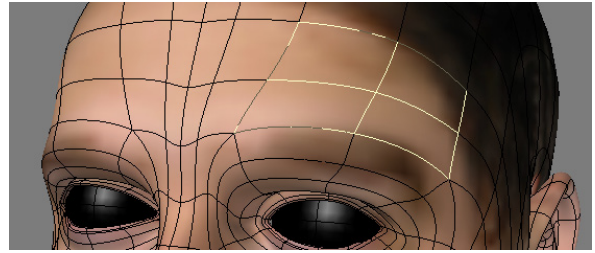


#### *Point Mode:*



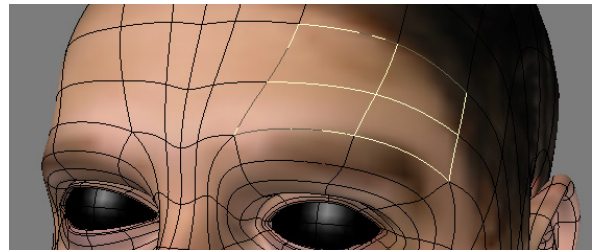
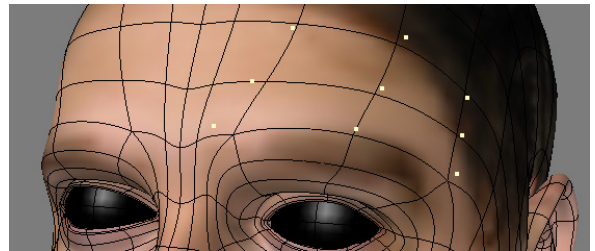
### Select Points

**Select Points** command (**View > Selection: More > Select Points**) will convert your polygon selection to points by retrieving all the points making up the selected polygons.



### Select Polygons

**Select Polygons** command (**View > Selection: More > Select Polygons**) will convert your point selection to polygons by retrieving all the polygons enclosed by the selected points.

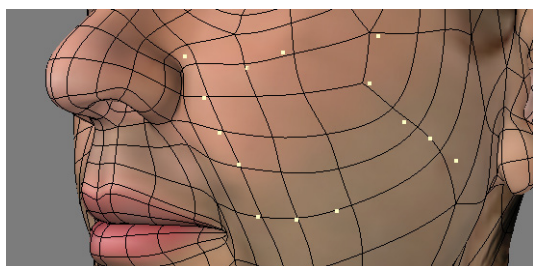
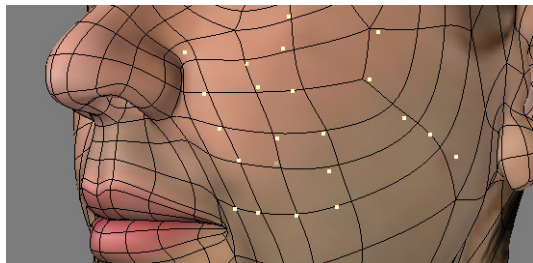




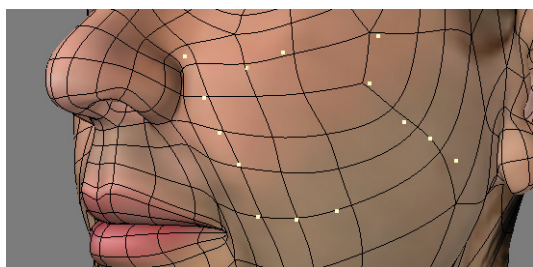
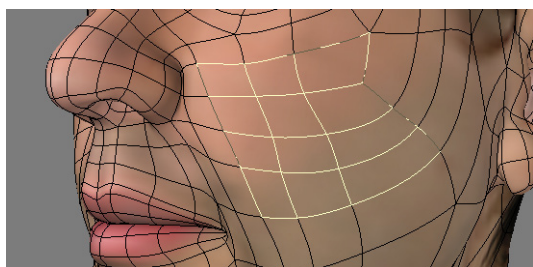
## Select Outline

The **Select Outline** command (**View > Selection: More > Select Outline**) works in both Point and Polygon Mode.

In **Point Mode**, this tool removes all the inner points in your selection and selects an outline bordering those inner points.



In **Polygon Mode**, this tool selects an outline of all the polygons you have selected, not selecting the inner points.

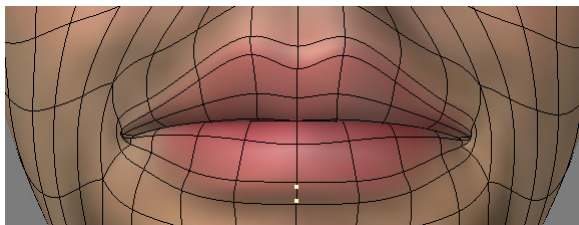


## Select Ring

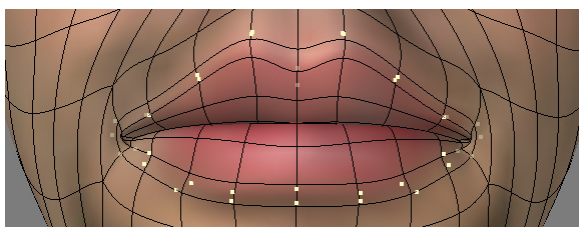
The **Select Ring** command (**View > Selection: More > Select Ring**) will select a pair of loops along two selected points.

**To Use Select Ring:**

**Step 1:** Select two points making up an edge.

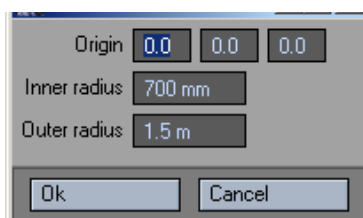


**Step 2:** Choose **Select Ring** and the tool will select a pair of loops along those two points.



## Radial Select

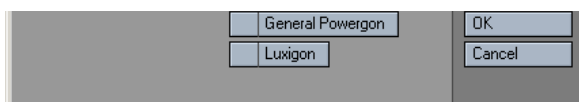
The **Radial Selection** command (**View > Selection: More > Radial Select**) will select the points that fall within a defined radius.



**Origin** is the XYZ center of the selection area. Points will be selected if they fall within the **Inner radius** and **Outer radius**.

## Select Powergons

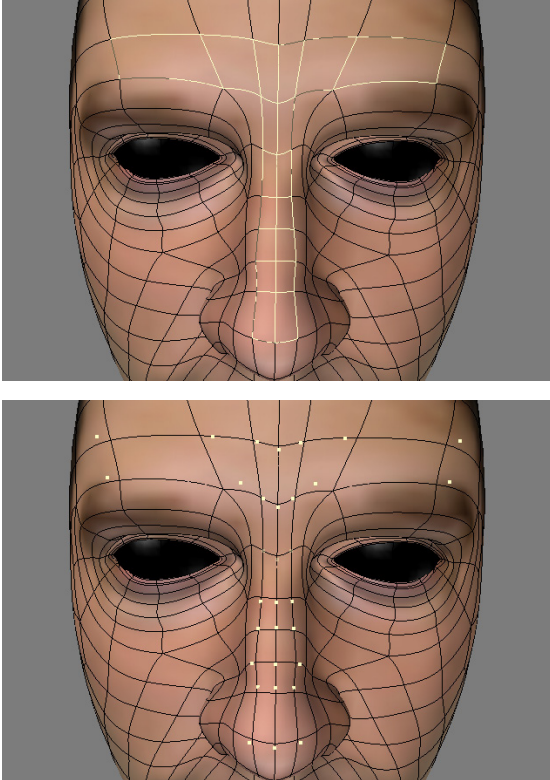
The **Select Powergon** command (**View > Selection: More > Select Powergons**) will give you the ability to select **Powergons** and **Luxigons** in an object.





## Selection Switch

Choose the **Selection Switch** command (**View > Selection: More > Sel Switch**) to convert your polygon selection to points by retrieving all the points making up the selected polygons or convert your point selection to polygons by retrieving all the polygons enclosed by the selected points.



## Selection Sets

### Selection Sets

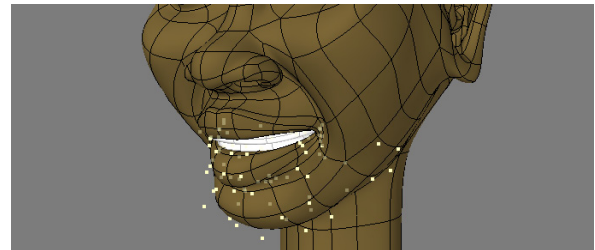
**Selection Sets** (**View > Selection Sets: Selection Sets**) will save groups of points that can be reselected by a user-defined name.



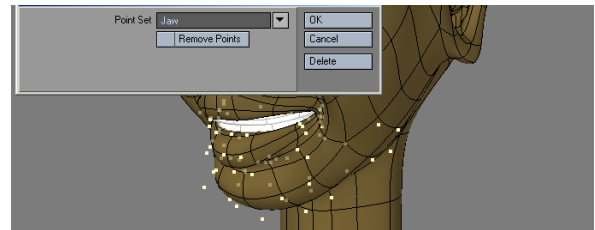
NOTE: A point can be a member of multiple sets.

**To create a point selection set:**

**Step 1:** Select a group of points.

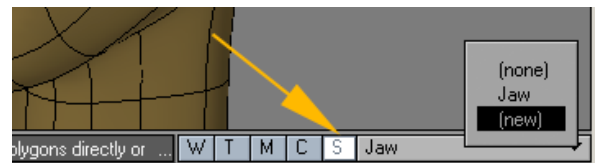


**Step 2:** Click **View > Selection Sets > Selection Sets**. Enter the desired name in the **Name** field and click **OK**.



NOTE: You can add to an existing set by selecting the additional points and then selecting the set's name from the pop-up menu.

You can also create and manage your **Selection Sets** in the **Vmap** area of the interface.







### To select points in a selection set:

**Step 1:** In **Points Selection Mode**, open the **Point Statistics Panel** located at the bottom of Modeler's interface.

**Step 2:** Click on the pop-up menu triangle on the last line in the panel. (If it isn't visible, drag the bottom of the panel down). Select the desired selection set.

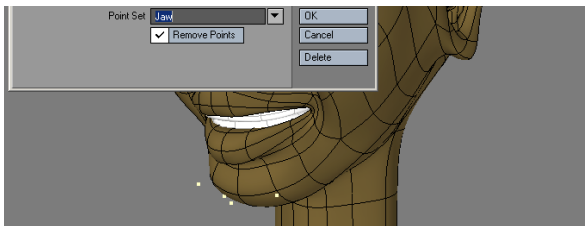
	Name	Num
+ -	Total	980
+ -	0 Polygons	0
+ -	1 Polygons	4
+ -	2 Polygons	26
+ -	3 Polygons	100
+ -	4 Polygons	780
+ -	>4 Polygons	70
+ -	Jaw	81

**Step 3:** The selected set's name will appear on the line. Click the plus sign (+) to select all of the points in the set.

### To remove points from a selection set:

**Step 1:** Select the points you wish to remove.

**Step 2:** Click **View > Selection Sets > Selection Sets**. Select the set from the pop-up menu.



**Step 3:** Activate the **Remove Points** option and click **OK**.

You can delete the entire set reference by selecting it and then clicking the **Delete** button.



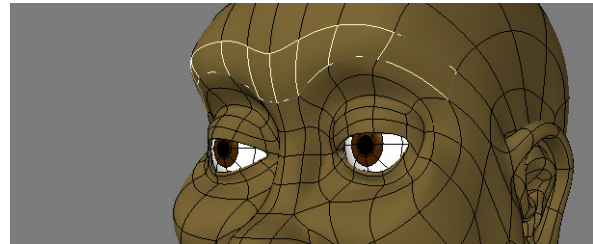
NOTE: Neither deleting operation will physically delete the points.

## Create Part

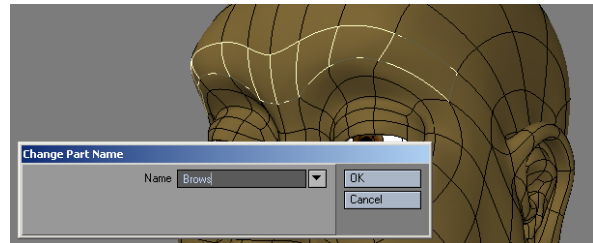
**Parts** are the polygon version of point selection sets. (One distinction, however, is that a polygon can be assigned to only one part name, while a point can be a member of multiple selection sets.) You can quickly select a user-defined set of polygons by name as you would with surface names. This feature operates independently and in parallel with surface naming.

### To name polygon parts:

**Step 1:** Select a group of polygons.



**Step 2:** Click **View > Selection Sets > Create Part**. Enter the desired name in the **Name** field and click **OK**.



You can add to an existing part group by selecting the additional polygons and then selecting the group's name from the pop-up menu.

### To select parts:

**Step 1:** In **Polygon Selection Mode**, choose the **Statistics** button at the bottom of the interface to open the **Polygon Statistics Panel**.

**Step 2:** Click on the pop-up menu triangle on the last line in the panel. (If it isn't visible, drag the bottom of the panel down.) Select the desired part name.

	Name	Num
+ -	Total	964
+ -	Faces	0
+ -	Curves	0
+ -	SubPatches	964
+ -	Skelegons	0
+ -	Metaballs	0
+ -	1 Vertex	0
+ -	2 Vertices	0
+ -	3 Vertices	16
+ -	4 Vertices	948
+ -	>4 Vertices	0
+ -	Non-planar	0
+ -	Surf: (none)	0
+ -	Part: Brows	8
+ -	Col: (none)	140

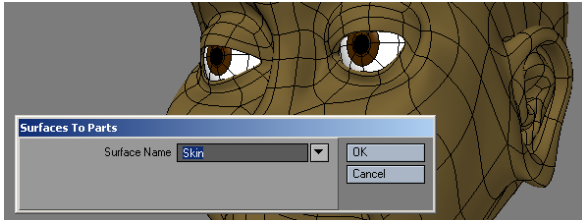
**Step 3:** The selected part's name will appear on the line. Click the plus sign (+) to select all of the polygons defined by the part.



NOTE: Like surfaces, any one polygon can have only one part name. Therefore, if you create a new part that uses polygons from an existing part, those polygons will become associated with the new part only.

## Surface to Parts

The **Surfaces To Parts** command (**View > Selection Sets: Surfs to Parts**) can convert surfaces to part names. The **Part** names are taken from the original **Surface** name.



## View

### Hide Selected

(default keyboard shortcut -)

**View > View: Hide Selected** will hide selected polygons that you don't want edited. This makes it easier to work on parts of a complex object without splitting it up between layers. The polygons are still there, but any editing operations will not directly affect the hidden polygons unless they share points that are edited.

### Hide Unselected

(default keyboard shortcut =)

**View > View: Hide Unselected** will hide all unselected polygons. Select all of the polygons that you don't want to hide and then **Hide Unselected** to hide all the unselected polygons and keep the selected ones visible.



NOTE: Although you cannot directly affect hidden polygons, you can affect them if you modify visible polygons that share points.

### Hide Invert

(default keyboard shortcut I)

**View > View: Hide Invert** will swap the polygons presently visible with polygons that have had **Hide Selected** applied to them.



NOTE: Although you cannot directly affect hidden polygons, you can affect them if you modify visible polygons that share points.

## Unhide

(default keyboard shortcut I)

Selecting **Unhide All** (**View > View: Unhide**) will reveal all hidden geometry.







---

## Glossary

---



## Glossary of 3D Terms

**1-Sided** — When a polygon is created, unless otherwise set up, it will have only one side. If you were to look at a playing card, it has a front and a back. A 1-sided polygon only has a front, and therefore only one surface normal.

**2D Map** — Two-dimensional map consisting of either a bitmap or a procedural map. An object using a 2D map needs texture coordinates. See **UV** for further details.

**2-Sided** — Like a playing card. A polygon that has a front, and a back, is 2-sided. A 2-sided polygon will have two surface normals, facing opposite directions.

**3D** — A three-dimensional medium, display, or performance, especially a cinematic or graphic medium in three dimensions.

**3D Accelerator Card** — A graphics card specifically designed for 3D graphics. LightWave uses a system of 3D graphics called OpenGL, and your graphics card must support this.

**3D Map** — Three-dimensional map either built up from multiple layers of bitmaps or, more often, generated in three dimensions with a procedural texture. These are algorithms that can generate 3D maps resembling marble or wood, and when applied to an object, the grains of the marble, and the fibres of the wood, will be correctly mapped to the surface in all three dimensions. If you split a 3D-mapped cube in two halves, the cross section surface will match the neighbouring faces. A 3D map does not require texture coordinates.

**3D Object** — Anything with a position and representation in 3D space. Some objects have a special role, for instance a camera or a light, while others serve as controls for other objects, for instance splines or manipulators. The most common 3D objects are geometric objects, classified according to whether they are polygon meshes, surfaces, curves, implicit objects, or nulls.

**3D Shutter Glasses** — 3D glasses made with electronic liquid crystal shutters. They are powered by the computer they are attached to and use this power to turn on and off the liquid crystal in each of the lenses creating a 3D effect, instead of the usual 2D display a computer monitor can offer.

**3DS** — Aged file format used by Autodesk 3D Studio and discreet 3d Studio max for three-dimensional scenes. It contains geometry, textures, lights and cameras as well as animation data, but polygons cannot contain more than three points.

**Absolute Coordinates** — The location of a point in terms of distances and/or angles from a fixed origin. See **Relative coordinates**.

**Adaptive Supersampling** — Way of antialiasing an object's surface by decreasing the oversampling rate for those pixels that do not require the oversampling. The results of adaptive supersampling are slightly more localised, and computing time is often shorter than other sampling methods.

**Additive Mixing of Colored Light** — There are two sorts of mixing of colors. One is called additive, or sometimes transmissive, and refers to the fact that the more red, green and blue you

add together the nearer to white your final color will be. This is the normal light scheme for LightWave or other graphics packages with output mainly through the medium of a screen. Subtractive mixing indicates that the fewer colors you mix the nearer to white you are and is used for reflective color, such as printed material.

**Additive Opacity** — Type of opacity that adds the background color to the material color of the transparent object.

**Aggregate Object** — An object that is made up of a number of other objects. A normal aggregate object will be made up of primitives. A more complex aggregate object may be made up of primitives, other aggregate objects, or both.

**Algorithm** — A problem-solving method that involves using a multi-step process.

**Aliasing** — When referring to pictures, aliasing is the effect that occurs when a line looks jagged instead of smooth because of a contrast in colors. Usually, you can tell when this happens because the line between the colors looks very jagged, as if it were a flight of stairs, in fact it is often referred to as a "stairstepping" effect. For contrast, see **Antialiasing**.

**Alpha Channel** — One of the four channels (or components) of information that make up every pixel in an image. There are three channels for red, green, and blue (RGB) and one alpha channel. The alpha channel is really a mask — it specifies the transparency of each pixel, which allows portions of the foreground image to reveal the background when two images are overlaid.

**Alpha/Matte Image** — Generally refers to an image where the brightness of each pixel is used to cut or partially dissolve out another image. These are generally greyscale or black-and-white images, but the brightness values can also be extracted from a color image.

**Ambient Component** — Part of the reflection-illumination model. A common surface shader parameter that adds consistency to the color of an object's surface to simulate an ambient light that reaches all points in a scene. An ambient value is determined for individual surfaces. Scene ambience is multiplied with an object's ambient color. If the scene ambience is set to black, nothing alters the ambient color of an object except, of course, a light. The careful balance of ambient and direct light sources is the key to convincing lighting. Global illumination is an alternative to ambient light that is more accurate but takes longer to render.

**Ambient Map** — Allows manipulation of the ambient component of an object's reflection-illumination model. Usually the ambient component is given a value near that of the diffuse component.

**Ambient Light** — All-directional light illuminating every object uniformly from all sides.

**Anaglyph** — Moving or still pictures in contrasting colors that appear three-dimensional when superimposed. Anaglyph works well for printed matter or computer display, but color problems inherent in television displays (both PAL and NTSC) result in poor 3D broadcasts.

**Anamorphic Distort** — An option referring to the width of a lens flare. When selected, the larger the distort factor, the wider the lens flare will become.



**Angle of Incidence** — The relative angle between a lit surface and the light source. The more the surface is turned away from the light source, the less light it receives and the darker it becomes. When the angle of incidence is 90 degrees, the light shines directly on the surface and it is illuminated with maximum intensity.

**Animate, Animation** — The movement of elements through time and space. Also, the process of creating and recording images that change over time. Everything in a scene is represented by numeric values and, as such, animation is also the process of changing these values — position, color, or any other property — over time. A method of creating the illusion of life or movement in inanimate objects or drawings. Through animation the artist's drawing comes to life. The most well known works are cartoon comedies, like *Ren & Stimpy* or *The Simpsons*.

**Animatics** — Preliminary animated versions of a final video or film presentation.

**Animation Channel** — Animation Channel refers to the different position, rotation, and scaling settings an item can have in Layout. It can also refer to other envelope elements like light intensity. See also **motion channel**.

**Annotation** — The process of inserting text or a special note, explanation or to provide relevant detail to a surface, a rig or a point in your scene in LightWave.

**Antialiasing** — A method for blending harsh contours and preventing staircasing or stairstepping. It is achieved by taking the surrounding areas into account when assigning a color value to pixels lying on an object's contour.

**Antisymmetry Surface Restraint** — The restraint of a surface tangent to the surface. This implies that the structure is symmetrical about this plane, and the load on the implied symmetrical part is equal to, but in a direction opposite to, the modelled part.

**Aperture** — The opening size of a camera lens. The greater the aperture, the smaller the depth of field and the greater the amount of light entering the lens.

**API** — Abbreviation for application programming interface.

**Arc** — Curved open element with a constant radius around a single center point. Section of a circle.

**Area Light** — A special kind of point or spotlight. The rays emanate from a geometric area instead of a single point (entire surface uniformly emits light). This is useful for creating soft shadows with both an umbra (the full shadow) and a penumbra (the partial shadow).

**Array** — A set of elements put together into a single entity. A pixel array is an ordered set of colored elements used for display purposes. In a 3D program, the array tool is usually used to create ordered copies of an object within 3D space. This tool is so named because it creates arrays of objects (creates an ordered set consisting of multiple copies of the same object).

**Aspect Ratio** — Description of the proportion of an image by comparing its width to its height. 35 mm slides have the aspect ratio of 4:3 (1.33:1). Images become distorted if forced

into a different aspect ratio during enlargement, reduction, or transfers. It should not be confused with the pixel aspect ratio, explained further on.

**Atmosphere** — Volumetric effect that simulates reduced visibility in air over distances.

**Attenuation** — The strength of light diminishes with distance when it travels through air. The further light travels, the dimmer it becomes. In real life, light attenuates by the inverse square of the distance. If attenuation is turned on for a light, only the geometry in its proximity will be lit. Not only is this more realistic for your renderings, it also helps speed up rendering time since only the geometry close enough to be affected by the light needs calculation time. See also **Decay**.

**AVI** — Audio Video Interleave. A popular animation file format that combines video and audio.

**Axis of Motion** — In 3D space, the line that an object follows during movement.

**Axis of Rotation** — In 3D space, the line that an object rotates around.

**Axis, Axes** — Axis refers to the XYZ coordinates used as the basis for positioning elements in LightWave's 3D space. It is somewhat like the concept of left/right, up/down, and near/far.

**B Rep** — See **Boundary Representation**

**Backface Culling/Elimination** — A process included in most 3D graphics pipelines, backface culling eliminates triangles facing away from the camera. This is most efficiently performed by checking the orientation of the triangle normal in relation to the camera. The technique ignores geometry seen from behind so that only the fronts of objects that are facing the camera are rendered. Both faces of an object are rendered by default; that is, the ones whose normals are facing the camera as well as those that are not. You can choose which faces of the object you want to render as part of the rendering options: front, back, or both faces. Backface culling (rendering only the front) can improve performance because less geometry needs to be rendered.

**Background Color** — The color that occupies all space not occupied by text, images, or any other objects. LightWave's default background color is black.

**Ball** — (Sphere) A 3D circle or oval created by user-defined dimensions and settings.

**Bandwidth** — How much information a network can carry. Think of the network as a highway, and each message as a car. The more lanes in the highway, and the higher the speed limit, the more traffic it can carry. So the wider the bandwidth of a network, and the faster its speed, the more information it can carry.

**Banking** — This is when an object following a path rotates about the path when it changes direction.

**Barn Doors** — The flaps on the front of a movie light to assist in limiting the beam of light.



**Baud** — Bits per second. Hence kilobaud or Kbaud, thousands of bits per second. The technical meaning is 'level transitions per second'; this coincides with bps only for two-level modulation with no framing or stop bits.

**Bend** — To deviate from a straight line or position: The lane bends to the right at the bridge. To assume a curved, crooked, or angular form or direction: The saplings bent in the wind.

**Bevel** — A method of eliminating sharp edges from objects by extending an object's faces to round off corners.

**Bezier Curve** — A technique for creating curves that was attributed to and named after a French engineer, Pierre Bézier, who used them for the body design of Renault's cars in the 1970s.

**Bilinear Filtering** — Blurring the pixels in a bitmap when it is zoomed in so that it seems smoother than it really is.

**Bilinear Intensity Calculation** — A high-speed algorithm for generating shaded faces. Used in Gouraud shading and Phong shading.

**Binary Space Partition (BSP)** — Also known as BSP, this is a technique used in real-time rendering for resolving in which order polygons should be drawn. The technique requires that a so-called BSP tree is built before the scene may be rendered. As this build process is very costly in terms of execution speed, BSP trees cannot usually be calculated in real-time and thus essentially only support highly complex yet static 3D worlds.

**Bit** — The building blocks of computer data. Has either the value of 1 or 0 (current or no current). Bits can be grouped together to carry larger values.

**Bitmap** — Two-dimensional monochrome raster image. A bitmap is a black and white image marking boundaries. It is often used for clip maps in LightWave.

**Blade** — A thin plane placed in front of a light to cast a shadow, taking light off of an object. A device to create a shadow.

**Blending** — Mixing of two (or more) textures into one final texture that is displayed in rendering.

**Blitting** — The copying of a virtual frame buffer to the displaying screen.

**Bone Hierarchy** — Bones can be arranged to build a Bone Hierarchy, also called a Skeleton. The hierarchy defines how the movement of one bone affects other bones (up and down the hierarchy). If you then also add Constraints to the bone hierarchy, you have a Rig.

**Bones** — For any object, you can define a skeletal system composed of bones. By moving bones around, you can change the shape of an object.

**Boolean** — A mathematical system developed by English mathematician George Boole that expresses logical relationships between things. The results in a Boolean operation can be either true or false. Boolean is used in 3D to add, subtract, and other operations that involve Boolean calculations.

**Boolean Operations** — **Modeling** technique that uses two overlapping objects to create a new object. There are three kinds of boolean operations: subtraction, union and intersection. By taking the first shape and subtracting/unifying/intersecting it to the second — a new shape is created.

**Boom Shot** — A camera move. Usually describes a shot in which the camera is mounted on a crane. The camera can move in all axes of movement.

**Boundary Representation (B Rep)** — A polygonal mesh representation. A polygonal mesh is, most commonly, a simplification of a shape using facets to describe curvatures. Its surface, or boundary, is built up from several faces that describe the shape. If it is a polyhedron the polygon model can be identical to the shape, whereas an organic shape is represented by a more or less simplified version that mimics the curvature using facets with variable density.

**Bounding Box** — A cubic shape that exactly circumscribes a (more complex) 3D model and is used to optimise three-space calculations like ray tracing. By representing a more complex shape with a box, possible ray intersections can be investigated much more swiftly. Also used to represent complex objects for proxy animation and setup to speed up operations.

**Bounding Volume** — A way of speeding up ray tracing operations involving intersection calculations, by inscribing a complex mesh in a considerably less complex shape like a box or sphere. Often used when rendering must be done in a short amount of time. Instead of having to check the intersection of a more complex mesh, like a space ship or a teapot, the bounding box works as a stand-in, with the same maximum height, width and length as the mesh it substitutes for. Therefore a possible ray intersection can be either ruled out (if the ray doesn't pass through the bounding box, it doesn't pass through the mesh either), or let a more time-consuming algorithm take over working with the complex mesh instead.

**Box** — (Cube) A six-sided 3D object that can be thought of as a 3D square or rectangle. Boxes are created based on user-defined input as to the dimensions and locations desired.

**BSP** — See **Binary Space Partition**.

**B-Spline** — A free-form curve that is defined with parameters in which each separate vertex on the curve has an influence over a portion of the curve. In 3D, B-splines allow a user to control a curved line on two axes at once.

**Bump Array** — The purpose of a bump array is to create an ordered series of bumps in a surface. This tool means exactly what its name implies - an array of bumps. See also **Array**.





- Bump Map** — Creates the illusion of three-dimensionality of a surface (protrusions and cavities) by recalculating the normals of the object, without changing the mesh itself. It is very common in 3D renderings and suitable for creating effects like wrinkles, creases, crumples, cracks, seams etc. The silhouette of a bump mapped object is a give-away since, in these areas, it is obvious that the mesh is left unaffected (if trying to create an orange by using a perfect sphere with an orange peel texture applied to it for bump mapping it will still have an impeccably round silhouette). In LightWave areas in a bump map that are black are unaffected and areas that are white are raised.
- Byte** — 8 bits. Multiples of bytes make up the terms kilobyte (1024 bytes), megabytes (1024 kilobytes) and gigabyte (1024 megabytes).
- CAD** — Computer Aided Drafting (or Design); A system that lets a designer use a computer screen instead of a drafting table to make plans and blueprints. However, CAD packages don't often have superb rendering abilities, so packages like LightWave are turned to, for their abilities.
- CAM** — Computer Aided Manufacturing; the process of using a computer to create a physical product from a computer-created design. CAM is usually used to control robots that perform tasks that would be tedious or dangerous to human workers. See also **Rapid Prototyping**.
- Camera** — An apparatus for taking photographs, generally consisting of a lightproof enclosure having an aperture with a shuttered lens through which the image of an object is focused and recorded on a photosensitive film or plate. Digital cameras use CCDs to focus light and create a digital picture that can be seen and transferred to a computer immediately. In LightWave, the camera is the conduit through which objects and scenes are turned into still images or animations.
- Capping** — Surface closing the upper and/or bottom side of an object such as a cylinder.
- Cartesian Coordinate** — Mathematical representation of Euclidean space. Every point can be described by three coordinates (X, Y, Z) representing the position along the orthogonal X, Y, and Z axes. The point (0, 0, 0) is called the origin, which is the global center of the 3D world.
- Cartesian Space** — A space in which positions are denoted by a three-coordinate system (x, y, and z coordinates) relating to a central origin (0,0,0).
- Catacorner** — Slanted across a polygon on a diagonal line; "set off in a catty-corner direction across the vacant lot". syn: cata-cornered, catercorner, cater-cornered, catty-corner, catty-cornered, kitty-corner, kitty-cornered.
- Cattiwompus** — Weird. Mixed up. Unusual. Distorted.
- Caustics** — Light pattern created by specular reflection or refraction of light, such as the patterns of light on the bottom of a swimming pool, or light through a glass of wine.
- CD** — Compact Disc storage media. Also the chemical symbol for Cadmium and the sticker on the back of a car with the diplomatic corps.
- Center of Projection** — The point in perspective projection where all projectors intersect.
- Center of the World** — Is the absolute center of a 3D space, represented by X, Y, and Z points (0, 0, 0). Also referred to as the Origin.
- CenterPoint** — A point that represents the center of an object. This point is used in some programs for a point of reference for rotation and position. The center point of a polygon is where the line representing the normal comes out from.
- Chamfer** — To cut off the edges of the geometry with a planar cut, creating a more blunt shape, typically at a 45 degree angle. A bevelled edge or corner between two intersecting lines or surfaces.
- Child** — An object whose movements are influenced by another object, called the "parent".
- Chord Length Parametrisation** — See **Non-Uniform Parametrisation**.
- Clean Modeling** — Refers to the practice of removing geometry from a model that is not wanted or needed. Also refers to the use of proper geometry construction techniques, such as creating continuous surfaces, minimising narrow faces, and avoiding small corner angles, that facilitates downstream processes.
- Clipping** — More often than not, much of the graphics drawn for a specific scene does not fit into the viewport of the camera. Accordingly, those which fall outside of the viewport must be clipped so as they are not drawn. Depending on the nature of the application, there are two kinds of clipping: 2D and 3D. The earlier simply compares each pixel against the extents of the rendering viewport, while the latter technique uses the six sides of the view frustum to determine whether a 3D vertex is inside the viewport or not.
- Clone** — This tool creates copies of an object based on user-defined parameters for offset, motions, morphing, shadows, etc. This tool can be used to make ordered sets of objects, but is different from the array command because not all new objects need be exactly the same as the original.
- Cloud of Points, or Point Cloud** — A set of x-y-z coordinates obtained from a 3D scanner or digitiser. The data can then be turned into a continuous surface and used as a 3D model.
- CODEC** — Short for "COmpressor/DECompressor". This is the term used to reference the way that software programs handle different movie files, such as Quick Time, AVI, etc. The CODEC can control image quality, and can assign the amount of space given to the movie file. First, a multimedia publisher uses a codec to squeeze more sound and video into less file space. These compressed files are easier to fit on a CD-ROM and transfer to your computer. Then, your computer uses a codec to expand these files back to their original size and replay them on screen.
- Coincidence** — Refers to geometry that occupies the same spatial location. For example, coincident vertices are points that occupy the same x, y, and z coordinates. Coincident lines can have differing lengths while one occupies the same location as the other.



**Color Bleeding** — When the color from one surface reflects onto another surface.

**Color Depth** — The number of bits used to represent a color. For example an 8-bit image uses  $2^8=256$  colors. The bits build up the three primary colors red, green and blue.

The bits are divided into red, green and blue (optionally an alpha channel as well).  
For example a 16-bit color could look like this R: 4-bit (16), G: 4-bit (16), B: 4-bit (16), Alpha: 4-bit (16) — together they add up to 16-bits. The number of bits can also be unevenly divided (R:5, G:5, B:5 Alpha:1).

This is why a GIF (max 8-bit=256 colors) only sports 128 colors if it is transparent (1 bit is used to represent transparency in the alpha channel, 7-bits =  $2^7=128$ ).

The following table indicates the number of colors an image can have.

8-bit =  $2^8 = 256$   
16-bit =  $2^{16} = 65536$   
24-bit =  $2^{24} = 16$  million  
32-bit =  $2^{32} = 4.3$  billion

You should also be aware of FP, or HDR images.

**Color Model** — A system used to specify colors. In LightWave, you can set color according to the following color models: RGB (red, green, blue), HLS (hue, lightness, saturation), HSV (hue, saturation, value) or integer values.

**Color Keying** — An old technique of combining two images by replacing one color of the destination image by the corresponding pixels of the source image.

**Column Interleaved Format** — The 3D image format used by the VRex VR-4200 projector ([www.vrex.com](http://www.vrex.com)). Left and right view image data are encoded on alternate columns of the display.

**Compiled Vertex Array (CVA)** — Array of geometry data on a vertex level that is optimized (compiled) for faster access by the graphics card. (Note that this is an OpenGL term, and is known by other names in other APIs.

**Compositing** — A layering technique that places one image on top of another, properly taking into account transparent pixels, apparent depth, shadowing and other elements that make up an image.

**Concentric** — Having a common center or origin point with varying radii.

**Cone Angle** — The angle at the peak of a cone.

**Conic** — Element having the form of a cone.

**Conic Section** — Curve formed by the intersection of a plane with a cone.

**Constraints** — Values in a geometric model that define relationships, i.e. a line is tangent to a circle. Constraints are often used to drive parametric or variational geometry-based systems; the algorithms used to work with constraints are known as constraint management.

**Continuous LoD** — Short for continuous Level-of-Detail, this method is based on the observation that 3D objects located far off in the distance may be approximated by simpler versions without loss of visual quality, thus increasing the rendering performance. The “continuous” refers to having the algorithm constantly recompute the detail level of the 3D object depending on the distance to the camera instead of having a pre-computed set of objects to choose from. Computationally expensive, this method is most often used in height field rendering applications. LightWave can approximate continuous LoD by using an Object list.

**Control Mesh** — A cage of points used to shape SubPatches.

**Convex Volume** — A convex volume can be defined as a volume whose every corner can be visible from all other corners in the same volume. Another way of defining the convexity is that all faces in the volume will be lit by a point light located anywhere within the volume.

**Cookaloris, Cookie** — A device put in front of a light, to break the light up. Common cookaloris look like leaves on trees, or blinds on windows.

**Coons Patch** — A free-form surface that is determined by the four curves that define its edges.

**Coplanar** — Refers to two or more entities that lie on the same plane. Two planar surfaces, for example, that lie on the same 3-dimensional plane are considered coplanar. If these coplanar surfaces share a common edge, it is recommended that they be joined into a single surface.

**Cross Product** — Using two vectors to calculate their normal.

**Cross-Section** — A view of the interior of an object as it is sliced along a plane.

**Cubic Image Map** — One of the many methods of applying a texture to a surface. This method applies the texture to an object as it would apply the texture to a cube. There are many other methods of texturing objects, such as Planar and Cylindrical image mapping.

**Curvature Continuity** — A curvature continuity with smooth transition of the edges of two meeting surfaces (the highlights of the two surfaces blend together seamlessly, forming the illusion of a single shape). If a curve (or surface) has tangent continuity and both the 2D curves (or 3D surfaces) have the same radius a very smooth transition is created with curvature continuity. Curvature is defined as  $1/\text{radius}$ . Hence, small radius equals high curvature.

**Curve** — In computer graphics, there are different ways of representing a curve, such as NURBS and Bezier curves. See also **NURBS** and **Bezier**.

**Curve Parametrisation** — See Parametrisation.

**CVA** — See **Compiled Vertex Array**.

**Cylindrical Image Map** — One of the many methods of applying a texture to a surface. This method applies the texture to an object as it would apply the texture to a cylinder. There are many other methods of texturing objects, such as Cubic and Planar image mapping.





**Decay** — Phenomenon where light intensity decreases with distance. The further away from the light source, the less intense are its rays. In the real world the decay is proportional to the inversed square of the distance (quadratic decay), but there is also directional (one-dimensional) decay (slower than in real life) as well as cubic decay (faster than in real life). See also **Attenuation**.

**Decompression** — Process of returning a compressed file to its full size.

**Default Unit** — The Default unit is the unit of measure (ex. meter, feet, etc.) that is assumed, usually when no unit of measure is entered with the numeric data. In Layout, it is determined by the setting on the General Options Tab of the Preferences panel. In Modeler, the setting is on the Display Options panel.

**Depth Buffer** — See **Z-Buffer**.

**Depth Cueing** — The process of reducing the apparent sharpness of an object the farther away it is from the viewer or camera. This often enhances the perception of depth.

**Depth of Field (DoF)** — The total distance, on either side of the point of focus, which, when viewed from an appropriate distance, appears sharp in the final print.

**Depth Sorting** — Sorting all triangles in the world depending on diminishing depth (lower and lower z-value) so when rendered, the triangle closest to the viewer obscures those behind it.

**Deskew** — Process used to remove skew or distortion through a small angle rotation.

**Diffuse Component** — Part of the reflection-illumination model. The diffuse is concerned with the amount of light that is reflected back.

**Diffuse Light** — A component of the reflective model that is the result of direct illumination.

**Diffuse Map** — Replaces the diffuse component of the reflection-illumination model, basically giving the illusion of being painted onto the surface. To create a material resembling wood or marble, this map is used. Generally, when you talk about the “texture map” in an application, this is the map actually referred to.

**Dimension** — A measure of spatial extent, especially width, height, or length.

**Directional Light** — See **Distant Light**.

**Director of Photography (DP)** — Person on set that determines how to photograph the movie.

**Disc** — Shape that is referred to in mathematics as a cylinder. This shape is composed of two circular or oval-shaped bases and the space contained between those bases. In other words, a disc is like a stack of circles with set parameters defined by you.

**Displacement Map** — Can be used to modify the actual mesh (as opposed to the bump map) to create wrinkles, creases, crumples etc. The displacement map will need a more complex mesh to create the same effect as bump mapping, but has the advantage of allowing more thorough close-ups, since the surface is actually deformed and not just simulated as being so.

**Display Types** — Ways of displaying objects in a viewport. Display types are available only for geometry views. The available display types are Bounding Box, Vertices, Wireframe, Front Face Wireframe, Shaded Solid, Textured Shaded Solid and Textured Shaded Solid Wireframe. Display types do not determine the quality of the final render.

**Distant Light** — A light with color, intensity and direction. All rays emitted from a distant light are parallel, and therefore it has no obvious source. Distant lights can be used to simulate point lights from a great distance (whose rays can be approximated to be parallel), for example the sun. The intensity from a distant light does not decay.

**Dithering** — Creating the impression of having more color on the screen than there actually are by plotting pixels (with a limited amount) of different colors next to each other.

**DoF** — See **Depth of Field**.

**Dolly** — To move the camera along its line of sight (in a straight line following the imaginary path between the actual camera and its target point).

**Dongle** — A hardware lock used to prevent the piracy of LightWave.

**Dopesheet, Dopetrack** — Animation tools in LightWave that allow you to better organise keyframes.

**DPI** — Dots per Inch. In a bitmapped image, the number of dots that exist within each inch of the image. This number remains constant, so when you make an image larger, the quality decreases, but when you make the image smaller, it appears to increase.

**Double Buffering** — This is the process of using two frame buffers for smooth animation. While the image of the first buffer is being displayed, the graphics controller can use the second buffer to build or render the next image. Once the second image is completed, the buffers are switched. Thus, the result is the appearance of smooth animation because only complete images are displayed, and the process of drawing is not shown. You can often now see triple buffering in graphics cards to allow an extra buffer for the next image in case there is a problem.

**DP** — See **Director of Photography**.

**DVD** — A high-density compact disc for storing large amounts of data, especially high-resolution audio-visual material. DVDs used solely for computers are commonly referred to as DVD-ROM.



**DWG** — AutoCAD native file format. It can contain 3D data, but is hard to convert to a LightWave-native format because of its construction. A DWG file is parametric, that is to say it does not contain the objects themselves, but rather instructions on how to build the objects. This makes it hard to translate if you do not possess a licence of AutoCAD. The solution is to either get one or get your client to supply you the object in a different format, preferably OBJ.

**Easing** — Reduction in the acceleration or deceleration of motion to present a smoother, more continuous movement. The shape of a function curve can reflect this when using spline interpolation.

**Edge** — Straight line connecting two points on a polygon.

**Edge Loop** — Particular method of modeling organic shapes with the edges of polygons creating a loop or a flow around circular features, like the eyes and the mouth for example.

**Endomorph** — Object containing one or more Morph Maps.

**Envelope** — Way of animating a particular value over time using a graphical input mode.

**Environment Map** — Map often used to simulate (faking) reflection of the surrounding world without using ray tracing.

**Euler Angles** — Euler angles are one of the simplest methods of representing 3D rotations, and generally also the easiest to visualize. An object's rotation can be specified in terms of its yaw, pitch and roll, or rotation around the Y, X and Z axis, respectively. Euler angles suffer from singularities in the form of so-called Gimbal lock, however, and are also difficult to smoothly interpolate for keyframe animation.

**Expression** — Mathematical expressions that allow you to change the animation of an object. You can also create constraints between objects using expressions or create conditional animation. Expressions are very powerful for creating precise animation and to create automated animation of things such as wheels.

**External Attributes** — The position of the camera and the direction it is pointing in.

**Extrude** — Creating a three-dimensional object from a two-dimensional shape by adding a third dimension to it. You can also do this along a motion path or spline.

**Face** — The shape made up by the bounding point making a polygon. Faces can have as many vertices as wanted, but only polygons having a shape of three or four vertices can be made into subdivision surfaces.

**Face Normal** — Also just known as the normal, this is a line perpendicular to the face that also describes which way the face is pointing in a one-sided polygon.

**Falloff** — The volume starting at the outer rim of a spotlight's hotspot, decaying from full intensity at the start to zero intensity at the outermost rim of the spotlight. The less the difference (in angles) between the hotspot/falloff, the crisper the shadows. If the falloff angle is much larger than the hotspot angle, the boundaries of the area lit up by the spotlight will be fuzzy.

**Field of View (FOV)** — The angle of the view frustum. The wider the FOV, the more you see of the scene. Human eyes have a FOV of about 50 degrees, and normally virtual reality application use similar values to resemble real life.

**Field Rendering** — An option that causes the program to render two interlaced fields of information. This is in contrast to rendering only one (full frame) and makes moving objects appear to move more smoothly. Used for projects that will play back on television monitors that display 50 or 60 interlaced frames per second. Fielded animation is not useful for animations designed to be displayed on computer monitors. See **Fields**.

**Field Sequential 3D Video** — The most common format for 3D video. Left and right image data are encoded on alternate fields of the standard video signal.

**Fields** — Interlaced images (video) consist of two fields combined into one frame. Each field contains half the scan lines (either even or odd) and is a separate pass. This is more common to render to for TV broadcast. Moving items horizontally will strobe without rendering to fields.

**Fill Light** — Additional light sources assisting the key light in a scene. Usually they are less intense than the key light and created using point light or spotlight.

**Fillet** — To round off the edges of an object with a round shape. Think "router", use "Rounder" in Modeler to achieve it.

**Fill-rate** — The amount of pixels from a texturemap (texels) that are rendered per time unit. Measured in texels/second.

**Filter Opacity** — Type of opacity that uses a color to simulate object opacity.

**Finger** — A small strip placed in front of a light to cast a discrete shadow.

**FK** — Forward Kinematics. Positioning a .gure by specifying joint angles, like posing a toy action .gure.

**Flag** — A large device placed in front of a light to create a shadow, creating a large shaded area.

**Flat Shading** — Shading technique where all individual faces in a mesh are assigned a single color value based on the orientation of their face normals.

**Flatness** — Flatness is used as a threshold in determining if a polygon is non-planar. A flatness of 0 percent means the polygon is absolutely flat. Flatness is computed as a percentage deviation from a triangle (the "ideal plane") formed from the first two and last vertices of a polygon. All of the other points are measured relative to this plane. The largest deviation is divided by the total size of the polygon to get a percentage that is the flatness value. For example, if a polygon is 1 meter wide, .5% flatness means that no point will be outside the ideal plane of the polygon by more than 5 millimeters. (1 x .005)

**Floating Point (FP) Images** — Refers to images that do not use standard color depth models to represent the colors contained in them, but rather an expression of the floating point value of a color changing from 0 for black up to 1 for the brightest point in the image. A mid-grey in such an image would be represented by R: 0.5, G: 0.5, B: 0.5.



- Focal Length** — The distance between the lens and the light-sensitive surface on the backplane (the film in real-world cameras). The lower the focal length, the more of the scene is visible. Focal lengths less than 50 mm is called wide angle, while lengths greater than 50 mm is referred to as telephoto lenses. The longer the lens, the narrower the field of view. Distant details become more visible. The shorter the lens, the wider the FOV. More of the environment is visible in the rendered image. To simulate the human eye, you can use values of about 50 mm.
- Fog** — Simple yet effective tool often used in real-time graphics to obscure the far plane, thus bounding the viewing distance of the application. There are essentially three types of fog: table-based, vertex-based, and volumetric. Fog values may also follow linear or exponential curves.
- Foreground Image** — The image closest to the camera.
- Foreshortening** — The apparent effect of viewing an object on its long axis that makes it seem shorter. For instance, an arm pointing directly at the camera seems to lose its length as does a road going directly away toward the horizon.
- Formula-Defined Shapes** — Refers to shapes that are defined by using one or more equations. This includes complex shapes such as aesthetic bottles, or simple shapes such as hyperbolic paraboloids, oblate spheroids, prolate spheroids, or ellipsoids.
- FOV** — See **Field of View**.
- FPS** — Frames Per Second. The main unit of measure that is used to describe graphics and video performance.
- Frame** — One image out of many that define an animation. There are 24 frames per second in film, 25 frames per second in PAL video, and approximately 30 frames per second in NTSC video.
- Frame-Buffer** — The memory a computer uses to hold one or more frames for later use.
- Frame-Rate** — The speed at which a frame of animation is shown, usually expressed in frames per second. European TV is at 25 frames per second, US is typically 29.97 frames and movies are projected at 24 frames.
- Freeze** — To convert from vector or interpolated geometry (splines, NURBS, subdivision surfaces) to pure polygons. Even if the renderer supports NURBS or subdivision surfaces, this freezing happens at render time, and is usually definable to the level of polygon creation by you.
- Frustum** — The part of a solid, such as a cone or pyramid, between two parallel planes cutting the solid, especially the section between the base and a plane parallel to the base. See **View Frustum**.
- FX** — Shorthand term for effects.
- GCore (Geometry Core)** — Engine in LightWave that handles all animation and Modeling tools.
- Generic Primitive** — Simple 3D objects that most 3D programs can create easily. These objects typically consist of spheres, cylinders, cubes, and cones.
- Geometry** — Positional layout of points and polygons for an object. These points are usually seen with objects that can be rendered. For example, a cube's geometry is composed of eight points. A curve has geometry since it is composed of one or more points, whereas nulls have no geometry.
- Gimbal-Lock** — What happens when two axes of rotation line up, thereby making three-dimensional rotation impossible. As an example, take any object with neutral rotation (0 degrees on heading, pitch, and bank) and rotate the pitch 90 degrees. Now, try to rotate the bank. This is gimbal-lock.
- Gizmo** — See **Null**.
- Global Illumination** — Unlike local illumination, this method of generating images supports effects not only linked directly to the light sources themselves. In real life, the intensity of a surface depends not only on direct illumination from the light source itself, but also from indirect illumination from surfaces being lit.
- Ray tracing can cast shadows from an object onto a surface, allowing objects to be reflected in shiny surfaces or refracted in transparent materials. Radiosity is the effect of reflected light. If you have spotlights projected at the ceiling in a white room, the light will bounce back and light up the entire room. However, this can only happen if the renderer supports radiosity (as LightWave does) or other similar techniques.
- Glossiness** — Affects how spread out across a surface a lighting highlight is. Low glossiness makes a spread out highlight while high glossiness creates a more central, pinpointed highlight.
- Glossiness Map** — An image to control the glossiness of a surface. Bright values in the image indicate more glossiness, dark values less.
- Glow** — Optical light effect that looks like a fuzzy disc around the center of a light source.
- Goal** — An object used in IK to create a point where an object will always reach for. This is used to make objects appear to have realistic motion.
- Gouraud Shading** — Developed by Henri Gouraud in 1971, this is a fast incremental shading technique using bilinear intensity calculation to create smooth transitions between the vertices in a triangle. It is most often used for lighting purposes by computing the vertex normals in the polygon, calculating the light values for each vertex, and then Gouraud shading the polygon. Even though it has obvious advantages over flat shading, the facets in the mesh can still be discerned. The placement of the highlight depends on the underlying polygons.
- GUI** — Graphical User Interface. The graphical interpreter between man and computer allows a more intuitive interaction with the computer. The window maker in UNIX, and Windows for the PC are both GUIs. This way you don't have to be computer literate to the same extent as if you should have to type all commands you wanted the computer to perform.



**Greeblies** — English slang. This describes the non-such little details on objects, usually mechanical objects. Those details which can be found on spaceships, in engine rooms, etc. You can also use the words “didges”, “nurnies” and “doohickies”.

**Halo** — Optical light effect that forms concentric circles around the center of a lightsource. Often clearly visible around street lights after a rainy day.

**Hidden** — Any element that is not shown in the current rendering of the scene but that still exists.

**Hidden Surface Removal** — Algorithm for removing obscured polygons in a three-dimensional view space. As opposed to the faster algorithm backface culling, the hidden surface removal algorithm is able to sort out those polygons that are obscured by another object. Another way of finding an obscured polygon is the z-buffer.

**Hierarchy** — A way of defining objects in relationship to each other (using a parent-child or tree analogy). This relationship means that transformations, deformations, and any other property of the parent object affect all child objects. This allows separately modelled objects to be used in a scene as a single functional unit. The movement of a parent affects the movement of the child, but you can move the child without affecting the parent.

**HDRI** — High Dynamic Range Image. An image with a wide intensity range between the brightest and darkest pixels. In typical 8/24-bit images, the maximum possible intensity range is 255 times brighter than the darkest grey pixel (with a value of 1). Natural scenes and images rendered with radiosity can have dynamic ranges from 10 to 10,000 times greater than this. Recording this information requires use of an image format with higher precision - such as LightWave's native .FLX format.

**Highlight** — Reflection of a light source on an object's surface. The size of the highlight (the area that shows the light source reflection) depends on the angle. Consequently, multiple light sources result in multiple highlights. This is also the Specularity.

**HLS Color Model** — Hue, Lightness and Saturation: the three components of the HLS color model. Hue refers to the position of the color in the spectrum, such as red, yellow, or green. Lightness is the amount of white mixed in a color, such as the difference between a pure red and pink. Saturation is the purity of the color, such as the difference between a pure red and a dusty rose - low saturation means that there is grey in the color.

**Hotspot** — The inner intense cone of light emanating from a spotlight.

**HSV Color Model** — Hue, Saturation, Value: the three components of the HSV color model. This color model defines the hue and saturation similar to the HLS model. Value is similar to lightness, as in HLS; however, a value of 1 represents a pure color when saturation is 1, while a lightness of 1 yields white no matter what the saturation. In both systems, 0 is black.

**Hub, The** — Module in LightWave that allows the Layout and Modeler modules to synchronise information. It uses the TCP/IP protocol to transfer information between modules.

**Hue** — The position of the color in the spectrum that describes the tone or tint of a color, such as red, yellow, or blue.

**HyperVoxel** — Voxels are volumetric rendering effects. HyperVoxels are voxels that are applied to nulls, points, or objects.

**IK** — Inverse Kinematics. The process of determining the motion of joints in a hierarchical 3D object given the desired start and end points, all the while obeying the laws of kinematics. Think of it like the strings on a marionette puppet.

**Image instance** — A copy or instance of a source image. Each time you use a source image, an instance of it is created. You can have as many instances of the same source as you need. You can then edit, crop, or even blur the instance without affecting the original source image.

**Image Map** — An image that is applied to an object's surface.

**Incandescence** — The emission of visible light by a hot object. In LightWave, this is the luminosity channel.

**Incremental Shading** — See Interpolative Shading.

**Indirect Illumination** — Light that bounces off one surface and illuminates another surface. This can only happen if the renderer supports radiosity. The LightWave renderer supports radiosity.

**Intelligentities** — Refers to LightWave's object format. The object format can contain morphs, multiple layers, and independent pivot points on a per layer basis.

**Intensity** — The strength at which the light source illuminates objects in the scene.

**Interference Checking** — The process of identifying if and where two or more pieces of geometry (usually solids) intersect. When moving parts are involved, a kinematics analysis is used to detect interferences.

**Internal attributes** — Properties of the camera such as depth of field and line-of-sight - compare with External Attributes.

**Interpolation** — Process used to estimate an unknown value between two or more known values. In animation, interpolation is the process used to calculate values at frames between two keyframes in a sequence.

**Interreflection** — When a reflective object reflects another reflective object. For example, if you place two mirrors in front of each other, the first one will display the second one, who, in turn, shows the first one. In real-life, there is virtually no upper limit of how many interreflections may occur, whereas in 3D rendering, one must set an upper limit to be able to render the scene. The default value for LightWave is 16, but it can be lowered to 0, if desired, or raised to 24 at a cost in increased rendering time.

**IR Transmitter** — A device that sends synchronisation signals to wireless shutter glasses.

**Isometric view** — Standard view in a 3D design where the top, front, and right side faces of a cube are equally inclined to the screen surface.





- Item** — An item in Layout refers to an object, bone, light, or camera.
- JPEG** (Joint Photographic Experts Group) — A widely accepted, international standard for compression of color images.
- JPS** — Stereoscopic JPEG file. JPS refers to a stereoscopic image file format that is based on JPEG compression. Used by DepthCharge & other stereoscopic imaging applications.
- Junkyard** — A special directory used by some studios to hold mechanical and non-organic pre-modelled parts.
- Keyframe** — (Key) A frame for which you define an animation channel(s) (e.g., position or rotation) for an item in Layout. Animations are composed of a beginning keyframe, an ending keyframe and usually some number of keyframes in between. See also **Tween**.
- Key-Light** — Dominant light source in a scene, normally created with a spotlight.
- Kinematics** — The properties of each 3D object that control its transformations. These transformation properties are used to modify the selected object's scaling (size), rotation (orientation), and translation (position) in X, Y, and Z in either local and global space. Although related, kinematics are not to be confused with inverse and forward kinematics for animation.
- Kit-Bashing** — An expression taken from model making. The practice of using model kits to give detailing to a larger project. This is still in use. It refers to the taking of models that you have already made, to use in the creation of another, perhaps even basically unrelated model.
- Lasso** — One way to perform a selection of points or polygons. This method involves drawing a loop that encircles all of the objects that need to be selected.
- Latent Surfaces** — Surfaces that are no longer visible after a Boolean or intersection operation because they lie inside or outside the solid.
- Lathe** — Creating a 3D object from a 2D shape by rotating it around an axis.
- Lattice** — Either a way of deforming object using a lattice or a way of creating outlined geometry.
- Layer** — A portion of a scene. Each layer consists of an object or multiple objects that can be edited separately from the rest of the objects in a scene. A layer is basically a building block for a scene and each layer contains separate blocks for a final model.
- Left-Handed Coordinate System** — Coordinate system where the positive part of the Z-axis goes away from the observer (from the screen).
- Lens** — Part of the camera determining the optical characteristics of the image, such as wide angle, fish eye, and depth of field.
- Lens Flare** — Optical light effect made up from a number of bright discs. If the rays from a light source reflect off the surface of a compound lens in a camera, it can generate star-like patterns on the image. Lens flares tend to be a cliché of bad CG imagery, probably because of their short rendering time and flashy appearance.
- LoD** — Level of Detail. This is a term which refers to varying the amount of detail in an object depending on the distance from the object to the camera. Example: A car for a close-up would need to have every little detail modelled into it. Chrome, bumpers, body seams, door handles, etc. But that same car, as seen from a helicopter flying over a highway, might be able to be a simple cube with an image map applied to it.
- Level-of-Detail Control** — The ability to vary the amount of details displayed in a graphics image to improve performance. For instance, at a distance, models can appear as simple 3D figures, but as users zoom in, a more detailed representation is presented.
- Light** — In LightWave, a light is generally used just like a light in real life. Lights illuminate a scene and allow you to see the objects within it. Different types of lights are distinguished: ambient light, diffuse light, point light, spotlight, etc. There are also different terms used to simulate the way material properties are illuminated: ambient component, diffuse component, specular component. Incident light at a surface = reflected + scattered + absorbed + transmitted. Light has a major impact on the photorealism of a rendered scene, but can be hard to recreate.
- Light Source** — There are several different sorts of light sources used in 3D graphics to simulate light: ambient, distant, linear, area, and spotlight. Special light effects can be recreated such as volumetric light and glow. With radiosity, an object with a high luminosity value can cast light as well.
- Lighting a Scene** — One of the ingredients of a nice rendering is realistic lighting. It is often good to use one single light source (the key light) doing most of the work, helped out by some additional, less intense lights (fill lights) which illuminate the background of the rendered object to create a smoother look. Try to avoid shadows with edges that are too crisp, since this is unusual in real life due to radiosity.
- Lighting Model** — This is a model that uses a mathematical formula to decide what will happen when light strikes an object's surface.
- Light-Map** — Luminance map generated (normally rendered) individually for each polygon and then blended with the texture map to give the impression of light and shadows falling onto the polygon without having to draw the effect on the texture itself. The advantage of separating the light-map from the texture map is that if you should want to create a new "mood" for a scene you can set up new lighting conditions for the scene, re-render the light-maps and apply them to the mesh again, without having to redraw all texture maps.
- Linear Patterning** — The repetitive placement of the active pattern cell along a line, line string, shape, arc, circle, ellipse, or curve element.



**Line-of-Sight (LoS)** — Has become quite important in modern real-time interactive simulators, especially for military purposes. To cut down on the polygon count and increase rendering performance, programmers are often forced to employ schemes to simplify terrain at large distances. This, however, has the unfortunate drawback of warping the terrain, something that may make a difference for long distance targeting purposes. Because of this, modern terrain rendering algorithms such as ROAM tend to not simplify along the primary LOS.

**Local Coordinate System** — As opposed to the world coordinate system, the Local Coordinate System is tied to a specific object. LCS are used, among other reasons, to simplify the representation of complex objects by using several, different LCSes as reference points for the object's vertices. It is also easier to transform the object if you for instance can rotate it around its own "center of gravity" instead of the origin of the World Coordinate System.

**Local Coordinates** — Every object has its own origin, which is subordinate to the world coordinate system (or other objects that are higher in the hierarchy). Local coordinates are useful for determining positions of subordinate objects.

**Local Illumination** — A mathematical model capable of creating imagery where only direct illumination is considered. Depending on the distance from the lightsource, etc, each surface in the model can be given a color and intensity. This does not include shadows, reflections and radiosity.

**Loop** — A continuous playback of an animation sequence.

**Low-Poly Modeling** — To model using as few polygons as possible, to speed up rendering and processing time. Common style for games, but as game processor engines get better, and computers faster, this is losing ground as an art form.

**LScript** — LightWave's built-in scripting language. Can be installed and used just like plugins.

**Lumel** — Short for LUMinance ELEment, the lumel is a pixel in a lightmap which constitutes the color level in a specific area of the texture map it is superimposed upon.

**Luminance** — The black and white information (brightness, sharpness, and contrast) encoded in a color. The amount of luminance contained in a color is directly proportional to the amount of light intensity.

**Luminance Map** — An image to control the luminance of a surface. Bright values in the image indicate more light intensity, dark values less.

**Luminosity** — Much like glow, luminosity is a measure of how much light a surface gives off before any light strikes it. This effect can be used to create an object that gives off its own light.

**Magnet** — This tool allows you to move points in an object as if he or she was using a magnet. It has an area of falloff where the strength of the magnet decreases gradually to 0 giving a soft selection effect.

**Map** — An attribute that can be added to an object's surface to give it a certain look. Projecting an image so that it covers the surface of an object or images that affect the way an object looks. There are a variety of different maps used to create specific effects: diffuse maps, bump maps, opacity maps, etc. Maps can be divided into bitmap-dependent texture maps and procedural maps. The latter categories can, in turn be divided into 2D maps and 3D maps.

**Mapping** — Process of making one image conform to the size, shape, and/or texture of another.

**Material** — Even if it is hidden beneath another texture map, there is an underlying material in any surfacing. The material is applied to the whole object, and can be made to look like wood, plastic, glass, metal etc, (hence the name), by modifying its properties. Also referred to as Surface.

**Material Properties** — The different properties of a material such as the ambient component, diffuse component and specular component in the reflection-illumination model.

**Matrix** — Matrices form the core of linear algebra and are important tools in most engineering disciplines. In essence a two-dimensional array of numbers, matrices are often used in transforms of different properties, such as rotation, scaling, translation, deformation, and much more.

**Memory Swapping** — The transferring of data back and forth between active RAM memory and disk. When this happens, it can considerably slow down computing tasks such as rendering.

**Mesh** — Object made up from a number of triangular faces. Also, slang used to refer to objects.

**Mesh Complexity** — Describes the amount of information (number of vertices, normals, triangles etc) used to create objects. More complex meshes need more memory and are slower to process

**Meta-primitive** — A Metaball, Metaedge or Metaface object.

**Metaform** — Option used with the Subdivide tools. It does not simply divide the individual polygons of an object, but rather renders the edges of the polygons to be smooth, making the object seem less faceted and cleaner.

**MIP-Mapping** — Using a pyramid structure of a predefined fixed amount of differently sized bitmaps (original size, original size/2, original size/4, etc) to speed up rendering time by using less detailed textures for distant objects (represented by only a few pixels on the screen), and the full-sized version of the bitmap when the objects are closer to the observer. This way, moiré-pattern can be avoided.

**Mirror** — Creates an exact mirror image of the selected object. This tool is very useful for any symmetrical object, including faces, cars, and airplanes. This tool literally cuts the Modeling time of this sort of object in half.





- Modal/Non-modal** — A modal panel must be closed before you can continue working with the rest of the application. A non-modal panel lets you shift the focus between it and another part of the application without having to close the panel—you can continue to work elsewhere in the current application while the panel is displayed. Modeler's Numeric Panel is non-modal because you can do other things while it is open. In contrast, Modeler's Display Options Panel is modal because you must close it before you can continue working.
- Modeling** — The process of creating, or recreating, an object inside your 3D software.
- Moiré Pattern** — Optical pattern created due to aliasing. Usually appears as a swirling pattern along a distant edge.
- Morgue** — A special directory, used by some studios, to hold already modelled organic body parts for other modellers to draw from. If you have modelled a good head, hands, ears, feet, etc. there is no reason to model them again.
- Motion Blur** — The blurring of objects that move while the camera shutter is open, creating the illusion of movement. Motion blur also prevents strobing caused by too-rapid movement.
- Motion Capture** — Method used to input live movements into a computer from an external source.
- Motion Channel** — Generally the same as Animation Channel, but refers only to position, rotation, and scale (i.e., not light intensity.).
- Motion Path** — The line an object follows while in motion.
- Multiplex** — The process of taking a right and left image and combining them with a multiplexing software tool or with a multiplexer to make one stereo 3D image.
- Multi-Texturing** — Applying two (or more) textures on the same face. For example, a polygon can have a texture map resembling a brick wall and then be multi-textured with a light-map to give the illusion of being lit.
- Natural Light** — Light that exists in nature, such as sunlight or moonlight, depends on the time of day, season and location on the Earth. The sunlight on a clear day has an RGB value of about R:250 G:255 B:175. For simulating overcast it might be a good idea to add the blue component, whereas a sunset could be a little more orange. As opposed to artificial light, the natural light has only one source (the sun) and can most effectively be recreated using a distant light.
- Node** — The basic graph element used to represent distinct items (vertices, faces, etc.). A signal coordinate in a grid, or finite element grid point used to describe the structure. A node will lie on each vertex of a finite element, and additional nodes may lie along element edges to define curved element topology.
- Non-Planar** — Generally refers to a polygon where all points do not reside in the same plane and can occur only with polygons using more than three points. Non-planar polygons can cause erratic rendering errors. As an example, a square piece of cardboard sitting upon a tabletop will become non-planar on all vertices when lifted by a corner. Inherent in manipulation and deformation of a model, non-planar "holes" can appear in the surface consistency of models. Solutions include "tripling" (halving the quads diagonally) or tessellating the polygons into triangles. As an example, a triangular piece of cardboard sitting upon a tabletop will remain planar on one vertex when lifted by any corner. Thus, when joined on their vertices, a group of triangles are more robust when deformed.
- Normal** — A polygon normal is the imaginary line projecting out perpendicular to a surface at any point indicating the direction of the polygon. A polygon surface normal is represented as dashed lines on selected polygons in Modeler. LightWave sees polygons or faces of an object only from the surface normal side. A single-sided polygon (like a piece of paper) with its normal facing away from the camera will be invisible to the camera from that viewpoint (unless the surface is using the Double Sided option). A vertex normal's direction is the average of the polygon normals it is connected to.
- NTSC** — National Television Standard Committee. The most common video standard in the United States and Japan. It has a frame-rate of roughly 30 fps. 60 times per second every other scan line is changed, resulting in smoother transitions. Its pixel resolution is 720x486 with a pixel aspect of .9
- Null** — Non-renderable helper-object used in modeling programs to simplify the manipulation of 3D-objects and texture mapping.
- NURBS** — Abbreviation for Non-Uniform Rational B-Splines.
- Nurnies** — American slang. See greeblies.
- Object** — A model or construction that when placed in a scene will render what it represents from the real world. An object is composed of points and faces. Points connected together to form a polygon define a face. Faces joined together form an object.
- Object Oriented Graphics** — Different from bitmap format, this image type consists of objects that have definite mathematical formulas behind them. These images always print at the maximum quality specified by the printer, unlike bitmapped images that always print at the same quality level. They can also be referred to as "vector graphics".
- Omni-Directional Light** — Same as a point light.
- Opacity** — The opposite of transparency.
- Opacity Map (or Transparency Map)** — Makes the surface more or less transparent depending on the pixel intensity (color value) of the opacity map where normally black is transparent and white is opaque.
- OpenGL** — A 3D graphics API that includes capabilities for 2D imaging. Basically, OpenGL is a set of instructions that can be used by a program to interpret images and display them on the screen. LightWave uses OpenGL for all its displays.



**Optical Light Effect** — If the observer (or camera) looks directly at a bright light source, it may appear to glow. If the light is refracted through a lens or even your own eyelashes (try squinting towards a spotlight!), the light will appear to form star-like patterns.

**Orbit** — To travel around a target - more commonly circular, but a comet's orbit can be elliptical.

**Origin** — The world Origin is the absolute center of the LightWave universe. A local Origin is the center of an object. Both are defined by the XYZ coordinates of 0, 0, 0.

**Orthogonal** — A view that displays a parallel projection along one of the major axes. In an orthogonal view, the camera is oriented to be perpendicular (orthogonal) to specific planes: the Top view faces the XZ plane, the Front view faces the XY plane, and the Right view faces the YZ plane. An orthogonal view eliminates the effect of distance from a viewpoint, which provides a useful means of locating points and objects in 3D space and is particularly helpful when modeling objects in wireframe mode. An orthogonal view is in contrast to a perspective view.

**Orthogonal Direction** — There are six different orthogonal directions in a three-dimensional space: up, down, back, forward, left and right.

**Orthographic Projection** — Viewing system where the projectors are parallel and therefore don't create a perspective with foreshortening.

**PAL** — Phase Alternating Line. The most common video standard in Europe. It has a frame-rate of 25 fps. It is interlaced, which means that 50 times per second every other scan line is changed, resulting in smoother transitions. The resolution is 720x576 pixels and the pixel aspect ratio is 1.0667.

**Pan** — To rotate the camera horizontally. As opposed to the orbit movement, pan rotates the camera around a single axis, as if it were mounted on a tripod.

**Panel** — In a 3D program, a screen that serves many functions such as informing you of errors, asking for user input, or informing you of the state a program is currently in. Otherwise known as a dialog, window or requester.

**Parabola** — A plane curve formed by the intersection of a right circular cone and a plane parallel to an element of the cone or by the locus of points equidistant from a fixed line and a fixed point not on the line.

**Parametrisation** — Technique for assigning values to the edit points as they are spaced along the length of a curve. Can be either uniform parametrisation or non-uniform parametrisation (chord length). The first edit point on the curve has the value 0.0 (regardless of whether it is uniform or non-uniform) and the following edit points are assigned greater values the closer they lie to the other end.

**Parameters** — Also generally known as properties, parameters are the "atomic" elements of a property set whose values determine the behavior of something. A parameter is one degree of freedom. You can set parameters in property editors.

**Parent** — An object that influences the motion of another object in a hierarchy, called the "child".

**Parenting** — The process of creating a hierarchical organization of objects in a scene. In parenting, an object (called the parent object) is "parented" to another object (called the child object). Parenting relationships can be nested to any degree, so that one or more objects are the children of another object, which is in turn the child of another.

**Particles** — 2-dimensional objects typically used in large quantities to create effects like rain and explosions.

**Passive Polarised 3D glasses** — 3D glasses made with polarising filters. Used in conjunction with a view screen that preserves polarised light.

**Penumbra** — A partial shadow, as in an eclipse, between regions of complete shadow and complete illumination, a fringe region of partial shadow around an umbra.

**Perspective** — A traditional art method of creating the illusion of three-dimensional form and distance on a two-dimensional surface. Perspective provides a three-dimensional view of the scene that indicates depth. In a perspective view, objects appear to converge toward a central vanishing point, and objects closer to the camera appear larger than those farther away. A perspective view is in contrast to an orthogonal view.

**Perspective Projection** — Simulating three-dimensionality by using foreshortening that triggers the human perception to interpret a two-dimensional image as if it was three-dimensional. An object is drawn smaller and smaller the further it is from the observer. This is achieved by using a center of projection to which all projectors converge, as opposed to where the projectors are parallel.

**Phong Shading** — The most frequently used interpolative shading technique used today. It uses a linear combination of three components - the ambient component, the diffuse component and the specular component. The placement of the highlight is less dependent on the underlying polygons as Gouraud shading since Phong shading interpolates the normals on a per-pixel basis instead of interpolating the light intensity based on the distance to the three vertices.

**Photorealism** — The process of generating computer images that mimic photographs.

**Pitch** — The amount that the camera or an object in the scene is tilted up or down. If you nod your head "yes", you are rotating your head in the pitch axis.

**Pivot Point** — A single point, usually in the geo-center of an object that is used for many functions. It is the point that is addressed to locate an object's position in 3D space. It is also the point around which all rotational moves are made and the reference point for transformations and scaling.

**Pixel** — Short for Picture Element, the smallest element of computer or video display.

**Plane** — Refers to a two-dimensional (i.e., flat and level) surface. Imagine a plane as a piece of glass that is infinitely large, but has no depth.

**Plugin** — A program that works with and extends the functionality of LightWave.



- Point** — A fundamental building element of an object in 3D space with an XYZ location. Point coordinates are the minimum information from which the geometry of an object can be calculated.
- Point Light** — Light source emitting light in all directions (omni-directionally) from a single point in space, think “light bulb”. It takes six shadow calculations (one in each orthogonal direction) to render shadows generated by a point light, which means that inserting multiple point lights into a scene might slow down rendering time considerably.
- Polygon** — Geometric shape in one or many planes. Polygonal Modeling consists of using many faces to create the shape. Since polygons in most cases are faceted simplifications of a much smoother shape, they are more or less inaccurate, as opposed to the more organic NURBS. The more the tessellation, the higher and the closer the accuracy compared to the desired shape.
- Poly-Line** — A geometric entity composed of one or more connected segments that are treated as a single entity.
- POV** — Abbreviation for Point of View.
- Primary Colors** — There are three primary colors of light: red, green and blue (RGB). Light colors are additive, which means that if these three colors are combined equally, the result is a white light. Black is thus the absence of light.
- Primitive** — Basic geometric shape used in Modeling. Some primitives consist of a combination of different primitives. Cone, box, sphere, tube, torus, and disc are common primitives.
- Procedural Map** — A map (often three-dimensional) generated mathematically using a set of user-customised variables instead of using an image. The procedural map does not need texture coordinates.
- Procedural Textures** — Mathematically generated textures (2D and 3D). Their advantage is that they are largely independent of the projection type.
- Projected shadow** — A shadow that falls from an object and projects on a surface.
- Projection Map** — A mapping procedure that allows you to apply the map to multiple objects as if they were one.
- Quad** — A polygon with four sides, short for quadrilateral.
- Quantise** — This tool causes points to snap to a specific (X, Y, Z) coordinate. This tool is generally used when a lot of precision is required.
- Quaternion** — Quaternions are mathematical objects consisting of a scalar and a vector which together form a four-dimensional vector space. Although having interesting uses in mathematics, their main use in computer graphics resides in their capability of easily representing 3D rotations. Although impossible to visualise, they suffer from no singularities like Euler angles, and are also easy to smoothly interpolate for keyframe animation (using a mathematical operation called SLERP for Spherical LinEar INTERPolation).
- Radiosity** — A more physically correct approach (developed in 1984 by Siegel and Howell) to simulate propagation of light in a virtual environment. It takes into account the fact that light bounces off a surface and creates diffused lighting on the surrounding objects. The scene is divided into a certain amount of triangles that are used to represent the original scene (which speeds up the time-consuming process), and then light interaction is calculated using these triangles. As far as visual quality is concerned, the more crucial the part of the scene the denser the triangles must be. This technique creates much more realistically lit environments, however it takes much longer to render due to the massive amount of calculations.
- Rail Clone** — This tool creates multiple copies of an object that are evenly spaced along one or more user-defined curves.
- Rail Extrude** — Used to extrude polygons along a specified line or combination of lines. This allows you to create a shape other than that created from a normal, linear extrude.
- Rapid Prototyping** — The process by which a computer model of a 3D object is turned into a real object. Rapid prototyping methods vary but often involve laying down strata of base material which is then bonded together using a substance like cyanoacrylate (superglue).
- Rasterisation** — The process of, on a per pixel basis, determining what value to assign to the pixels on the screen from a vector-based image.
- Ray Traced Shadow** — Shadow created by tracing the light rays from a light source. The ray traced shadows are more accurate than those created by shadow maps, but take more time to render and always have crisp edges.
- Ray Tracing** — An advanced rendering technique capable of calculating reflections, refractions and shadows. Ray Traced renderings take more time to generate, but have a more photorealistic quality than simple scanline rendering.
- Ray Tracing Depth (Ray Recursion Limit)** — Number of times the light bounces off a surface when ray tracing. Used to create reflections and/or refractions. For example, ray tracing two mirrors facing each other with the ray tracing depth set to 3 will allow the image of the reflected mirror to show the first mirror in it.
- Reflection** — Light that bounces off a surface. A mirror is highly reflective, whereas the reflection of a matte rubber surface is insignificant.
- Reflection Map** — Simulates reflections in a surface using the reflection map instead of actually ray tracing the reflected image. This speeds up rendering time, but can also be a give-away if the scene is animated.
- Reflection-Illumination Model** — Model used when creating two-dimensional images from three-dimensional meshes. To produce more realistic and convincing images, the reflection model imitates attributes of real-life objects.
- Refraction** — When light passes through a transparent material and into a denser, or less dense, medium the light rays are refracted and change direction. Each material has its own refraction and, depending on the density of the material, the refraction is more or less evident. Refractions are calculated similarly to reflections using ray tracing.



**Refraction Index** — A value describing the amount of refraction that takes place in a specific transparent material. For vacuum the refraction index is 1.0000, for air 1.0003, for glass approximately 1.5 and for water 1.3333.

**Refraction Map** — An image to control the level of refraction across a surface where dark values indicate a low refractive index and bright ones a high refractive index.

**Render** — To mathematically generate geometries, algorithms, reflections, etc. Our work would be meaningless without the ability to render. Creating a final image of a model that shows all of the surface properties which have been applied to an object. This process involves adding all colors; bump maps; shading; and other elements that add realism. In a normal 3D program, you can view the wireframe of the created image. When an image is rendered, the wireframe is covered with the specified colors and properties.

**Render Pass** — Division of a scene according to different aspects (such as highlight, mattes, or shadows) for the purposes of applying specific rendering options. Passes can then be composited during post-production. The default pass is the beauty pass, which includes all objects in the scene. Preset passes include matte, shadow, and highlight passes. You can also define your own passes to include any object you want to be affected by specific rendering properties. Render passes are further divided into partitions.

**Rendering Pipeline** — Description given to the process of creating the rendered images. Some studios have a process by which all the images go through. Some render in passes, one for the base, then the shadows, then the reflections, etc. This process is the pipeline.

**Resolution** — The number of picture elements in an image.

**Revolution** — A Modeling term defining a surface made by rotating a curve around the axis of another curve.

**RGB Color Model** — A color model that mixes the three primary colors to produce colors. To create yellow, red and green are mixed without any blue component. The higher the value of the red, green and blue, the clearer the color. Lower RGB values give darker colors, while higher RGB values give lighter colors.

**Rigging** — The process of making an object ready for animation. This does not have to be just characters; it is the same for all objects. Rigging involves creation and implementation of bones, hierarchies, clamps, weight maps and sliders.

**Right-Handed Coordinate System** — A coordinate system (frequently used in 3D-graphics applications) whose positive Z-axis emerges from the screen towards you, just like the one used in mathematics, as opposed to the left-handed coordinate system.

**Roll** — The amount that a camera is tilted to the left or right. Also known as the Bank Angle.

**Rotoscoping** — A technique in which video or film images are placed in the background of a scene, one frame at a time. You can use these reference images to create your own animation by tracing objects from the images or matching your objects with the images' motion. You can zoom and pan the scene while maintaining a perfect registration with the imported background.

**Row Interleaved** — A format to create 3D video or images in which each row or line of video alternates between the left eye and the right eye (from top to bottom).

**Rule-Based Design** — The definition of relationships between objects in the design. Another name used to describe Knowledge-Based Design.

**Scalar** — A quantity, such as mass, length, or speed, that is completely specified by its magnitude and has no direction, a one dimensional value.

**Scanner** — Device for reading images (from books, photos etc.) into the computer. This is useful for creating realistic textures. With a 3D scanner it is even possible to capture three-dimensional objects and convert them into models.

**Scene** — A Scene is a LightWave project defining the objects loaded and their motions, the number of lights and their values/motions, the resolution of the final image, special effects, Camera settings, and so on. This ASCII text file is generally saved from Layout.

**Scrub** — The process of manually dragging the frame advance control slider on the timeline to see or hear its effect on video/audio.

**S-Drill** — Refers to Solid Drill. Acts just as a drill would, using a 3D object as the drill bit. This tool can be used to take sections out of objects or perform other functions that a drill might perform.

**Seamless** — A seamless texture can be tiled without visible transitions where the bitmap begins and ends. This means that the upper part of the bitmap can be placed next to the lower part of the bitmap, or the right can be placed next to the left, forming a pattern that is apparently coherent.

**SECAM** — Séquentiel Couleur à Mémoire. The television broadcast standard for France, the former USSR, and various eastern European countries. Like PAL, SECAM is based on a 50 Hz power system, but it uses a different encoding process and displays 819 lines interlaced at 50 fields per second. SECAM is not compatible with NTSC or PAL, but conversion between the standards is possible.

**Secondary Animation** — Once the main movements of animation have been applied, this refers to the detail animation step. Hoses bouncing when a robot walks and flab wiggling when a heavyset character moves are examples of secondary animation.

**Sector** — Convex volume used to speed up rendering time.

**Self-Shadow** — Object property that allows one part of a complex object to cast a shadow onto another part of that same object (example: the branches of a tree casting shadows onto its trunk).





- Self-Illumination (or Luminosity)** — Allows non-homogeneous self-illumination of the surface. Some parts can be self-illuminated, some partially self-illuminated, and some not at all, based on the pixel intensity of the self-illumination map (normally black=left unchanged, white=self-illuminated).
- Session** — A session is a single use of an application. A session begins when you first start the application and ends when you exit.
- Shaded Mode** — Shaded mode generally refers to a viewport that has its Rendering Style (Display Options panel or viewport title bar) set to something other than wireframe. These modes show polygon surfaces with some level of shading.
- Shading** — Simulating that an object is lit by a light source.
- Shadow** — An area that is not or is only partially lit because of the interception of light by an opaque object between the area and the light source.
- Shadow Map** — Bitmap generated by the rendering engine during a pre-render pass of the lit scene. Generally a shadow map is less precise than a raytraced shadow, but takes less time to render. As opposed to a ray-traced shadow, a shadow map can create shadows with smooth edges. Furthermore, the shadow map is unable to show the color cast by a transparent object. The quality of the shadows in the rendered image depends on the size of the shadow map. The bigger the map the nicer the shadows. A shadow map that is too small might result in aliased or stairstepped edges. For example, a 256x256 shadow map (65k) is normally sufficient for resolutions of 320x200 and less. If an object is far away from the light source, the shadow map will have to be increased in order to maintain visual quality. If the final rendering is in high-resolution, the shadow map also needs to be hi-res.
- Skew** — Modifying an object by tilting it.
- Skin** — Creating three-dimensional object from two or more two-dimensional shapes and then extruding them along a path.
- Smoothing** — Technique that, when rendering or shading, smoothes out the edges between segments making objects appear smoother than their geometry really is.
- Soft Shadow** — does not have hard edges. Traditionally, ray traced shadows always have hard, crisp edges, whereas shadow-mapped are considered soft shadows. With global illumination methods, physically accurate, soft-edged shadows are achievable at a cost in rendering time.
- Space** — A set of elements or points satisfying specified geometric postulates: non-Euclidean space. The infinite extension of the three-dimensional region in which all matter exists.
- Specular** — This property determines how shiny (and sometimes wet) an object appears. It represents the highlight that the light creates when shining on an object.
- Specular Component** — Part of the reflection-illumination model. Specular surfaces are capable of reflecting light like a mirror.
- Specular Map** — Replaces the specular component of the reflection-illumination model, thus only visible in an object's surface's highlights.
- Specular Reflection** — The brightest area on a surface, reflecting surrounding light sources, creating highlights.
- Spherical Image Map** — One of the many methods of applying a texture to a surface. This method applies the texture to an object as it would apply the texture to a sphere. There are many other methods of texturing objects, such as Cubic and Planar image mapping.
- Spinning Light Trick** — Trick to create soft shadows with ray tracing. It involves parenting multiple lights to a null and spinning the null.
- Spline (Curves)** — Layout uses splines or curved paths between keys while moving items about. When Modeling, splines refer to open or closed curves.
- Spline Cage** — A spline cage is usually a three-dimensional object made up of connected spline curves.
- Spline Patching** — The process of adding polygons to fill in areas outlined by splines.
- Spot** — A small opaque circle placed in front of a light, usually to remove the specular hot spot from an object.
- Spotlight** — A lightsource emanating light in one direction only, in the shape of a cone.
- Staircasing (or Stairstepping)** — A graphical flaw caused by insufficient resolution. When rendering an object its contours might stand out too crisply from the background and the pixels might be obviously "zig-zagged", or look like stairs. To prevent this, pixels can be blended into their neighbours' colors by antialiasing.
- Stencil** — When using the drill tool, the stencil option adds the details of the drilling polygon to the polygon being drilled. This creates new geometry on a shape.
- Stereoscopic 3D** — Two separate photographs taken from slightly different angles that, when compiled, appear three-dimensional.
- Stretch Tool** — Allows you to change the size of an object along a particular axis.
- Subdivide** — Divides any selected polygons with three or four sides into smaller polygons. This makes an object appear smoother, but also makes the model more complex.
- Subdivision Surfaces** — Subdivision surfaces are a technique to create a smooth curved surface from a coarse polygon mesh. Several different subdivision schemes have been developed since Ed Catmull and Jim Clark first introduced the idea back in 1978. The most well known schemes are the triangle-based Loop scheme and Catmull & Clark's original scheme, which is based on quad polygons. Subdivision surfaces were introduced to the public in the Pixar movies, *Toy Story 2* and *Geri's Game*.
- SubPatch** — Refers to a Modeling mode wherein polygons become a cage that controls an underlying mesh of subdivision surfaces.



**Subtractive Opacity** — Type of opacity that subtracts the background color from the transparent object's material color.

**Super Pixel** — They are created in a supersampling image. Groups of the super pixels are filtered into the one single pixel that is displayed on the output display.

**Supersampling** — Generating images at a resolution  $n$  times  $n$  larger than the display resolution and then filtering the so-called super pixels into the smaller resolution image, creating smooth images with antialiasing.

**Surface** — Essentially, the surface is the skin of an object. The surface attributes can be changed using the Surface panel. Many features, such as the name and color attributes, affect the appearance of an object. A single object can have multiple surface names, each with its own independent attributes (e.g., color), and multiple objects can share the same surface name(s).

**Tangent** — A straight line that makes contact with a single point along a curve.

**Taper** — Modifying an object by progressively narrowing it along an axis.

**Target** — In aiming the camera, the target is the object that is selected for the camera to point toward. The target is kept in the center of the camera's view.

**TD** — Technical Director. A job in a studio that mainly concerns making rigs, and to help out the other departments wherever possible. They are the problem solvers.

**Tessellation** — Increasing the detail level of a polygonal 3D model by increasing its number of polygons, usually triangles. The more triangles, the smoother the shape and subsequently the larger the model. The tessellation can be performed by dividing one triangle into two (or more) smaller ones. By doing this the new, more faceted model can be modified without losing too much of its smoothness.

**Texture** — Normally texture describes the attributes of a surface, for example if it's coarse, smooth, wrinkled or rough, but it is also used with the meaning of texture map. There are textures made from bitmaps (texture map), and textures generated mathematically (procedural map). Textures specify how the surface of an object will look, and can be anything from simple, solid colors to complex images representing the surface of the object. The simplest example of a texture is placing a picture on a flat plane. The picture is the texture being applied to the plane.

**Texture Coordinates** — Coordinates used to describe how to map a texture map onto an object. There are different kinds of techniques to apply the texture: planar, cylindrical, spherical, cubic, front, and UV. Their names indicate how the texture is projected onto the object the mapping coordinates are applied to. Procedural maps do not need texture coordinates.

**Texture Map** — Map wrapped over the surface of an object. The texture map needs to be spaced correctly in U and V direction over the object.

**Texture Mapping** — The process of projecting a (usually) two-dimensional image onto a three-dimensional face such as a triangle or a quad, texture mapping is a relatively cheap way of adding tremendous detail to a scene without resorting to extremely detailed meshes that take an inordinate amount of memory and time to render.

**Tiling** — Repeatedly placing the same texture next to itself on the same surface, creating a pattern from one image. This is achieved by increasing the texture coordinates on a polygon to a value greater than 1. Normally, the entire bitmap is tiled from 0.0 to 1.0 in  $u$ - ( $=x$ ) and  $v$  ( $=y$ ).

**Timeline** — The slider below the Layout viewport representing time in animation.

**Transformation** — The act or an instance of transforming. The state of being transformed. A marked change, as in appearance or character, usually, /hopefully/ for the better.

**Truck** — To move the camera in the viewing plane.

**Twist** — Modifying a mesh by rotating its vertices non-uniformly along an axis.

**U-Direction** — Represents a grid line in one direction (normally that of the original curve) of a UV texture map.

**Umbra** — A dark area, especially the blackest part of a shadow from which all light is cut off. The completely dark portion of the shadow cast by the earth, moon, or other body during an eclipse.

**Unify** — This command creates single-sided polygons according to the properties of their surface normals. Basically, this tool transforms polygons that share points into a single polygon.

**Union** — One of the options in the Boolean tool. This option makes an object that is a combination of the two objects.

**UV-grid** — A grid system for identifying points on a surface. The U-direction and V-direction are for the surface, what the X-axis and Y-axis are for the coordinate system.

**V Map** — V Map is an abbreviation for vertex maps. V Map provide additional information associated with object points (vertices), like weight, UV and morph maps.

**V-Direction** — Represents a grid line in one direction (normally "up-down") on the surface of an object.

**Vector** — Entity with both magnitude and direction. A three-dimensional vector is written as:  $V=(v1, v2, v3)$  where each component is a scalar.

**Vertex** — (pl. vertices) Point at which the sides of a polygon intersect. In 3D graphics, vertex may also refer to a point at a specified location in three-dimensional space, and such a point is the smallest component in a 3D mesh.

**Vertex Count** — The number of vertices in a scene. Remember, the higher the mesh complexity the longer the rendering time.





**Vertex Normal** — Even though it is a single point in three dimensional space, its normal can be calculated based on the normal of the face they are describing. The three vertex normals of a single triangle without any neighboring triangles are set to be the same as the polygon's normal. For triangles surrounded by other triangles, the vertex normals are the average of the surrounding face normals.

**View Frustum** — Representing the field of view of the camera, the view frustum is a pyramid volume with the top sheared off. The top of the pyramid represents the viewport of the camera (usually the screen), and is often called the near (or hither) plane, while the bottom is called the far (or yon) plane.

**View Frustum Culling** — Removing faces that lie outside the observer's view. Only the face that is within the view frustum is kept for rendering — speeding up rendering time and helping to maintain a high framerate.

**Viewport** — Window area displaying orthogonal or perspective projection in a 3D application. The screen can either contain one big viewport or several smaller, tiled viewports. By simultaneously using several viewports displaying a three-dimensional object from different sides (e.g. top, front, left, perspective), modeling in a virtual 3D environment is made possible.

**VIPER** — Versatile Interactive Preview Render. A window that provides you with an interactive previewing system.

**Volume** — When selecting, a volume of an object is a 3D representation of the area to be edited. When editing, all of the parts of objects contained within this 3D selection can be edited without changing what lies outside of the selection.

**Volumetric Fog** — Fog that, opposed to ordinary fog, is restricted to fit within a containing volume.

**Volumetric Light** — Light simulating illumination of particles floating in mid-air, thereby making the light cone itself visible.

**Vortex** — A tool that rotates an object more in the center than in the outer edge. This tool can be easily related to a tornado, where the wind in the center moves faster than the wind in the outer part of the cone.

**Voxel** — Short for VOLUME ELEMENT, this term refers to a specific rendering technique common in medical visualisation as well as some interactive media. In essence, a voxel is a three-dimensional pixel, that is, a cube, with a specific color.

**Weights** — The strength of influence on a particular vertex of an assigned deformer, such as a bone. See V Maps.

**Weld** — This command takes the selected points and combines them into one point, a single point that is specified by the last point that is selected.

**WIP** — Short for Work In Progress.

**Wireframe** — A way of visualising geometry by drawing lines between its vertices but not shading the surfaces within.

**World Coordinate System** — The coordinate system, normally in three dimensions, used to describe the location in space of a specific point called vertex.

**X-Axis** — Usually is the axis that is left-right/side-side.

**Yaw** — To turn about the vertical axis, also known as heading.

**Y-Axis** — Usually is the axis that is up-down/top-bottom.

**Y-up** — Coordinate system with the Y-axis pointing upwards.

**Z-Axis** — Usually is the axis that is in-out/front-back.

**Z-Buffer** — Also called depth buffer, the z-buffer is a two-dimensional matrix of 16 or 32-bit integers with the same dimensions as the screen (or viewport). Whenever a polygon is drawn to the screen, the rasteriser checks the corresponding z-buffer value for each screen coordinate and skips drawing the current pixel if the z value is marked as being closer. This allows for some nice effects such as overlapping 3D models, and completely solves the rendering-order problem.

However, this comes at the price of slower performance and greater memory usage, two factors that have become more or less moot with the proliferation of modern 3D accelerators that tend to support z-buffers in hardware.

**Z-Up** — Coordinate system with the Z-axis pointing upwards.





# Index

## Symbols

3D Object 24

## A

Absolute Size 129, 136

Action Center Control 57, 27

Airbrush 228, 230

Color Maps 230, 232

Morph Maps 230, 232

Radius 229, 231

Strength 229, 231

Weight Value 229, 231

Aligner Tool 113, 116

Angle Tool 219, 220

Apply Morph 281,

Array Tool 166

Aspect and Video images 49, 55

Auto Patch 203,

## B

Background Conform 129, 136

Ball Tool 65, 66

Bandglue 195, 196

Band Saw Pro Tool 178

Band Saw Tool 183

Bend Tool 117, 121

Range 118, 122

Bevel Tool 140

Bezier Tool 98,

Bone

Moving a Bone 243, 249

Scaling a Bone 243, 250

Bones

Choosing Bones 254, 263

Displaying Weight 254, 264

Renaming Bones 255, 265

Rotating a Bone 242, 249

Bone Weight 235, 239

Boolean Command 195, 197

Box Tool 63

Bridge Tool 201, 204

Bubbles 71, 75

## C

Camera

Camera Presets Customizing 14

Capsule Tool 69, 73

Center 1D 114, 117

Center Data 110, 111

Center Scale 134,

Center Stretch 134,

Clone to Layer 169

Clone Tool 167

Closing

Object Files 36, 39

Collapse Polygons 194, 196

Color Map 228, 229

Commands 296

Cone Tool 68, 70

Configuring Divide 183

Construct

Collapse Polygons 194, 196

Combine 195, 197

Combine:Boolean 195, 197

Combine:Bridge 201, 204

Combine:Drill 199, 201

Combine:Solid Drill 201, 204

Convert 203,

Convert:Edit Metaballs 205, 189

Convert:Freeze 205, 189

Convert:Metaballs 205, 189

Convert:Metaedges 206, 190

Convert:Metafaces 206, 190

Convert:Subpatch 203,

Convert:Toggle Metamesh 206, 191

Faster and Better Drills and Booleans 195, 197

Patches 202, 205

Patches:Auto Patch 203,

Patches:Patch 202, 205

Reduce 190

Reduce:Bandglue 195, 196

Reduce:Points 190

Reduce:Polygons 190

Reduce:Polys + (Qemloss2) 191

Remove 193, 195

Remove Polygons 194, 195

Remove:Vertices from Polygons 193, 195

Construct Tab 190

Content Directory 39, 44

Create Part 309, 326,

Create Skin 163

Create Tab (Modeler) 62

Creating Points 88, 99

Creating Text 83, 91

Cruves

Smoothing Two Overlapping Curves 101,

Curve Divisions 40, 44

Curves 97, , 216

Bezier Tool 98,

Control Points 100, , 216

Curve Direction 100,

Curves into Polygons 101,

Modeling Tools 102,

Plot 1D Tool 103,

Sketching Curves 97,

Sketch Tool 97,

Smooth 216

Spline Cage 102,

Spline Draw Tool 99,

Customizing the Color Palette 252, 261

Custom Preferences 39, 43

Cut Tool 181



## D

Dangle 119, 123  
 Default Polygon 39, 44  
 Default Surface Name 39, 44  
 Default Unit 51, 58  
 Detail  
   Curves 216  
   Curves:Control Points 216  
   Curves:Smooth 216  
   Edges 217  
   Edges:Add Edges 217  
   Measure 219, 220  
   Measurement 219, 220  
   Points 208  
   Points:Merge Points 208  
   Points:Set Value 210  
   Points:Unweld 209, 210  
   Points:Weld 209  
   Points:Weld Average 210  
   Polygons:Aligning Polygons 212  
   Polygons:Double Sided 215  
   Polygons:Fix 3-5 214, 215  
   Polygons:Fix Poles 214  
   Polygons:Flip Polygons 211  
   Polygons:Merge Polygons 211, 210  
   Polygons:Sketch Color 213, 212  
   Polygons:Spin Quads 212, 211  
   Polygons:Unify Polygons 215  
 Detail Tab 208  
 Disc Tool 66, 68  
 Display Options (Modeler) 42, 45  
 Display Options Panel  
   Backdrop Tab 48, 54  
   Interface Tab 49, 55  
   Layout Tab 42, 45  
   Units Tab 51, 58  
   Viewports Tab 45, 50  
 Divide Tool 182  
 DragNet Tool 111, 113  
 Drag Tool 109, 110  
 Drills and Booleans 195, 197  
 Drill Tools 199, 201  
 Duplicate 165

## E

Edge Bevel 142  
 Edges 217  
 Edit  
   Normalize:Normalizing Weights 256, 266  
 Edit Metaballs 205, 189  
 Encapsulated Postscript  
   Export Encapsulated Postscript 36, 39  
 Encapsulated PostScript Loader 33, 35  
 Endomorph 280,  
 Equilateral Triangle 81, 88  
 Exporting Objects 36, 39  
 Extender Plus 152  
 Extender Tool 154  
 Extrude 147

## F

Fast Triple Fan Tool 185  
 Fast Triple Traverse Tool 188  
 Flatness Limit 39, 44  
 Fractalize Tool 186  
 Freeze Command 205, 189  
 Frequently Asked Questions 263, 275

## G

Gears 75, 80  
 Gear Tool 75, 80  
 Gemstone Tool 78, 84  
 General Options 39, 44  
 Grid Info Field 57, 27  
 Grid (Modeler)  
   Snap 51, 58  
   Units 51, 58

## H

Helix 174  
 High Dynamic Range Images  
   Internal Compositing 18  
 High Dynamic Range Images (HDRI) 17  
   Import/Export 18  
 Hub 18

## I

IMAGE AND ANIMATION TYPES 16  
 Images and Memory 17  
 Independent Options 48, 54  
 Independent Visibility 48, 54  
 Info Display (Modeler) 57, 27

## J

Jitter Command 128, 135  
 Julienne Tool 185

## K

Keyboard Shortcuts 37, 41  
 Knife Tool 175

## L

Lathe Tool 148  
 Layer Browser Panel 53,  
 Layer Navigation 52, 60  
 Layers 316, 315  
 Layout  
   Layout Viewport Navigation 14  
 Layout Communication 52, 60  
 Lights 247, 254  
   Choosing Light 247, 255  
   Distant Light 247, 255  
   Light Type 247, 255  
   Point Light 248, 255  
   Positioning the Light 247, 255  
   Setting the Light Source 247, 254  
   Spotlight 248, 256  
 Lightwave  
   Installation 4



- Registering 5
- Working With The Interface 8
- Lock UVs to Polygon 271,
- Logo 86, 96
- LSCompiler 298
- LScript 298
  - Execute Command 298
- LScript/RT 298
- Luxigon 292, 294
- LW\_ColrPikr 10

## M

- Magic Bevel Tool 155
- Magnet Tool 111, 112
- Maintaining Key Mapping Sets 38, 42
- Maintaining User Command Sets 296
- Make Command 58, 28
- Make Pole Tool 185
- Manage Fonts 82, 90
- Map
  - Color 237, 241
  - Color:New Color Map 237, 241
  - Color:Point Color 238, 242
  - Color:Textured Point 239, 244
  - General 227, 228
  - General:Airbrush 228, 230
  - General:Clear Map 230, 233
  - General>Edit Maps 231, 233
  - General:More 232, 234
  - General:Normalize Map 232, 235
  - General:Set Map Value 227, 228
  - Morph 280,
  - Morph:Apply Morph 281,
  - Morph:Bkg to Morph 281,
  - Morph:Create Joint Morph 282,
  - Morph:Endomorph 280,
  - Morph:New Endomorph 280,
  - Per-polygon Maps 226, 228
  - Texture 263, 276
  - Texture:Flip UVs 269, 283
  - Texture:Guess Viewport UV Image 276,
  - Texture:Make UVs 268, 282
  - Texture:New UV Map 268, 281
  - Texture:Per-polygon Mapping 263, 276
  - Texture:Point Map 272,
  - Texture:Point Maps:Quantize UVs 272,
  - Texture:Point Maps:Spread UVs 272,
  - Texture:Polygon Normal UVs 273,
  - Texture:Poly Map 270,
  - Texture:Poly Map:Adjust Polygon Map 270,
  - Texture:Poly Map:Clear Polygon Map 271,
  - Texture:Poly Map:Lock UV's to Polygon 271,
  - Texture:Poly Map:New Quad Polygon Map 271,
  - Texture:Poly Map:Poly Map to UVs 271,
  - Texture:Texture Guide 276,
  - Texture:Transform UV 269, 284
  - Texture:UV Mapping 263, 276
  - Texture:UV Spider 274,
  - UV Maps 225
  - Vertex Maps 224
  - Weight 233, 237

- Weight:Bone Weight 235, 239
- Weight:New Weight Map 233, 237
- Weight:PatchWeight2Weight 236, 241
- Weight:UV to Weight 235, 240
- Weight:Weights 234, 237
- Maps 305, 322, 324
- Map Tab 224
- Measure Tool 220, 221
- Menu Layout (Modeler) 38, 42
  - Edit 38, 42
- Merge Points 208
- Metaball 228, 230
  - Appearance 94,
- Metaball Resolution 40, 45
- Metaballs 205, 189. *See* Meta-primitive objects
- Metaedges 206, 190
- Meta Edges. *See* Meta-primitives
- Metafaces 206, 190
- Metaform Plus Tool 184
- Metaform Subdivide 177
- Meta-primitives
  - Animating 94,
  - Editing 94,
- Mirror Tool 165
- Modeler Edit Menu 37, 40
  - Basic Point and Polygon Maintenance 37, 40
  - Customizing Keyboard Shortcuts 37, 41
  - Undoing Operations 37, 40
- Modeler File Menu 33, 35
  - Importing an Object 33, 35
  - Loading an Existing Object from Disk 33, 35
  - Starting a New Object 33, 35
- Modeler Help Menu 59
- Modeler Interface 30
- Modeler Tools
  - Edit 39, 43
- Modeler Windows Menu 52, 59
- Modeling 24
  - Components of a 3D Object 24
  - Editing Objects 25
  - Modeling in 3D 24
  - Points and Polygons:Selection Mode 25
- Modeling in 3D 24
- Modeling Tools 102,
- Modes
  - Action Center Control 57, 27
- Modify Tab (Modeler) 106
- Morph 280,
- Joint Morph 282,
- Morph Map 227, 229
- Morph Polygons Tool 164
- Motion Path Clone Tool 170
- Motion Path Extrude Tool 160
- Move Plus 112, 114
- Move Tool 109, 110
  - Falloff Mode 106
- Multi-document Environment 52, 60
- Multiply
  - Duplicate:Array 166
  - Duplicate:Clone 167
  - Duplicate:Clone to Layer 169
  - Duplicate:Helix 174



- Duplicate:Mirror 165
- Duplicate:Motion Path Clone 170
- Duplicate:Particle Clone 168
- Duplicate:Point-Clone-Plus 173
- Duplicate:Radial Array 174
- Duplicate:Rail Clone 171
- Duplicate:Spin it 174
- Duplicate:Symmetrize 173
- Extend 140
- Extend:Bevel 140
- Extend:Create Skin 163
- Extend:Edge Bevel 142
- Extend:Extender 154
- Extend:Extender Plus 152
- Extend:Extrude 147
- Extend:Lathe 148
- Extend:Magic Bevel 155
- Extend:Morph Polygons 164
- Extend:Motion Path Extrude 160
- Extend:Rail Bevel 157
- Extend:Rail Extrude 152
- Extend:Rail Extrude:Multiple Curves 153
- Extend:Rail Extrude:Single Curve 152
- Extend:Rounder 143
- Extend:Router 159
- Extend:Seashell 161
- Extend:Spikey 162
- Extend:Super Shift 150
- Subdivide 175
- Subdivide:Add Points 175
- Subdivide:Band Saw 183
- Subdivide:Band Saw Pro 178
- Subdivide:Cut 181
- Subdivide:Divide 182
- Subdivide:Fast Triple Fan 185
- Subdivide:Fast Triple Traverse 188
- Subdivide:Julienne 185
- Subdivide:Knife 175
- Subdivide:Make Pole 185
- Subdivide:Metaform Plus 184
- Subdivide:Quick Cut 187
- Subdivide:Quick Saw 187
- Subdivide:Random Pricks 186
- Subdivide:Split Polygons 180
- Subdivide:Subdivide 176
- Subdivide:Triple 180
- Multiply Tab 140
- Multitext 84, 93
- Multiply
  - Subdivide:Fractalize 186

## N

- New Color Map 237, 241
- Normal Baker 233, 237
- Numeric Panel 57, 27

## O

- Object
  - Importing 33, 35
- Object Import/Export 16
  - Exported Formats 16
  - Imported Formats 16
- Objects
  - Editing 25
- OpenGL Display Clipping 313

## P

- Parametric Surface Object 79, 87
- Particle Clone 168
- Patch Divisions 40, 45
- PatchWeight2Weight 236, 241
- Pdbreaderv3- 34, 37
- Per-polygon Maps 226, 228
- Pivot Point 316, 315
- Platonic Solid Tool 73, 77
- Plot 1D Tool 103,
- Plot 2D Tool 80, 87
- Plug-ins 299
- Point-Clone-Plus 173
- Point Color 238, 242
- Point Map 272,
- Point Normal Move 116, 119
- Points 88, 98
- Points and Polygons
  - Selection Mode 25
- Point Tool 88, 98
  - Alternative Method 88, 99
  - Creating Points 88, 99
  - Random Points 91, 101
  - Spray Points 89, 99
  - Stipple 90, 101
- Pole Evenly Tool 130, 137
- Pole Tool 131, 138
- Polygon Map 271,
- Polygons 91, 211, 210
  - Aligning Polygons 212
  - Double Sided 215
  - Flip Polygons 211
  - Make Polygon 92, 103
  - Make Triangle Fan 96,
  - Make Triangle Strip 96,
  - Merge Polygons 211, 210
  - Pen Tool 91, 102
  - Points to Polys 93, 104
  - Unify Polygons 215
- Powergons 307, 324, 328
- Primitive
  - Platonic Solid Tool 73, 77
  - Toroid Function 72, 77
- Primitives 62
  - Ball Tool 65, 66
  - Box Tool 63
  - Bubbles 71, 75
  - Capsule Tool 69, 73
  - Cone Tool 68, 70
  - Creating in Perspective 62
  - Disc Tool 66, 68





Equilateral Triangle 81, 88  
 Gears 75, 80  
 Gear Tool 75, 80  
 Gemstone Tool 78, 84  
 Make UVs Option 62, 63  
 Numeric Panels 62  
 Parametric Surface Object 79, 87  
 Plot 2D Tool 80, 87  
 StarSphere Tool 82, 89  
 SuperQuadric Tool 79, 85  
 Teapot 81, 88  
 Tools and the Mouse Buttons 62  
 Using the Arrow Keys 62, 63  
 Wedge Tool 74, 79

## Q

Quantize Tool 135,  
 Quick Cut 1;2 and 3 Tools 187  
 Quick Saw Tools 187  
 Quit 36, 40

## R

Radial Array 174  
 Radial Select 307, 324, 328  
 Rail Bevel Tool 157  
 Rail Clone Tool 171  
 Rail Extrude Tool 152  
 Random Points 91, 101  
 Random Pricks 186  
 Reduce Points 190  
 Reduce Polygons 190  
 Resetting Tools 56, 26  
 Rest On Ground 116, 119  
 Rotate About Normal 121, 125  
 Rotate Any Axis 120, 125  
 Rotate Arbitrary Axis 122, 126  
 Rotate HPB 121, 125  
 Rotate Morph Function 123, 128  
 Rotate to Ground 122, 127  
 Rotate-to-Object 123, 128  
 Rotate Tool 117, 120  
 Rounder Tool 143  
 Router Tool 159  
 Rove Tool 115, 118

## S

Save View Presets 56, 26  
 Saving  
   Object Files 35, 38  
 Saving the Data 241, 247  
 Scale Morph Tool 133,  
 Seashell Tool 161  
 Sections 36, 39  
 Segment Scale 125, 130  
 Selection Sets 308, 325,  
 Selection Switch 308, 325, 328  
 Select Outline 307, 324, 327  
 Select Points 306, 323, 326  
 Select Polygons 306, 323, 327  
 Select Ring 307, 324, 328  
 Select UV Seam 305, 322, 325

Setup 286  
   Layout 292, 294  
   Layout:Add Luxigon 292, 294  
   Layout:Add Powergon 292,  
   Layout:Clear Powergons 293,  
   Skelegons:Convert Skelegons 290, 292  
   Skelegons>Create Skelegons 286  
   Skelegons>Edit Skelegons 288  
   Skelegons:Mocap Skelegons 291, 293  
   Skelegons:Rename Skelegon 288  
   Skelegons:Rotate Skelegons 288, 289  
   Skelegons:Selection 290, 291  
   Skelegons:Skelegon Tree Panel 289, 291  
   Skelegons:Skelegon Weight 290, 292  
   Skelegons:Split Skelegon 288  
 Shear Tool 110, 111  
 Size Tool 124, 129  
 Skelegons. *See* Setup Tab  
   Convert Skelegons 290, 292  
   Edit Skelegons 288  
   Mocap Skelegons 291, 293  
   Rename Skelegon 288  
   Rotate Skelegons 288, 289  
   Selection 290, 291  
   Skelegon Tree Panel 289, 291  
   Skelegon Weight 290, 292  
   Split Skelegon 288  
 Skelgons  
   Editing the Bank Rotation Axis of Skelegons 261, 273  
   multiple skelegons share the same weight map name  
     240, 246  
 Sketch Color 213, 212  
 Smooth 135,  
 Smooth Scaling 134,  
 Snap Drag Tool 114, 117  
 Spherize 136,  
 Spikey Tool 162  
 Spin it 174  
 Spline Cage 102,  
 Spline Draw Tool 99,  
 Spline Guide Tool 132,  
 Spline Patching 202, 205  
 Split Polygons 180  
 Spray Points 89, 99  
 Squarize 136,  
 StarSphere Tool 82, 89  
 Stipple 90, 101  
 Stretch Tool 124, 128  
 Subdivide 175, 176  
 SuperQuadric Tool 79, 85  
 Super Shift Tool 150  
 Surface  
   Change Surface 58, 29  
 Surface to Parts 310, 327, 311  
 Symmetrize 173  
 Symmetry Mode 58, 28



## T

Taper Constrain 127, 134  
 Taper Tool 126, 133  
 Teapot 81, 88  
 Text Numeric Panel 84, 92  
 Text Tool 83, 91  
 Text Tools 82, 90  
     Interactively Creating Text 83, 91  
     Logo 86, 96  
     Manage Fonts 82, 90  
     Multitext 84, 93  
     Text 83, 91  
     Text Numeric Panel 84, 92  
 Textured Point 239, 244  
 Texture Guide 276,  
 Texture Map 227, 229  
 Titlebar Shortcuts 47, 54  
 Toggle Metamesh 206, 191  
 Toroid Function 72, 77  
 Translate (Modeler) 109, 110  
 Translate Plus Tool 112, 114  
 Triple Tool 180  
 Twist Tool 118, 122

## U

Upright Rotation 47, 54  
 User Commands 296  
 Utilities  
     Additional 299  
     Add Plug-ins 299  
     Commands 296  
     Commands:Save Command List 297  
     Edit Plug-ins 299  
     Edit Startup 297  
     Plug-ins 299  
     Plug-ins:File Grouping Method 299  
 Utilities Tab 296  
 UV Map 235, 240  
 UV Map Jitter 273,  
 UV Maps 225  
 UV Texture Maps 225, 226

## V

Vdvertex Paint  
     Vertex Color Painting 250, 258  
 Vertex Loader. *See* Weight Maps  
 Vertex Map Panel 54,  
 Vertex Maps 224  
     Create Vertex Color Map 248, 256  
     Delete Vertex Color Map 248, 256  
     Rename Vertex Color Map 248, 256  
     Shortcuts 55, 24  
     Vertex Color Map Type 248, 256  
 Vertex Paint 240, 245  
     Bone Weight Paint 254, 263  
     Creating a Morph Target 261, 273  
     Customizing the Color Palette 252, 261  
     Light Setting 247, 254  
     Misc 261, 273  
     Showing / Hiding Polygons 244  
     User Interface 241, 247  
     Vertex Color Paint 248, 256  
 View  
     Layers 316, 315  
     Layers:All BG Layers 318  
     Layers>Delete Layer 316  
     Layers:Flatten Layers 318  
     Layers:Insert Layer 316  
     Layers:Layer Settings 317  
     Layers:Next Empty Layer 318, 319  
     Layers:Object Collapser 317, 316  
     Layers:Pivot Point 316, 315

Layers:Shift Layers Up/Down 318, 319  
 Layers:Swap Front and Back 317, 318  
 Selection 302, 309, 319  
 Selection:Contract Selection 304, 321, 323  
 Selection:Drop Selection 303, 320, 321  
 Selection:Expand Selection 304, 321, 322  
 Selection:Invert Connected 303, 320, 322  
 Selection:Invert Selection 303, 320, 321  
 Selection:Maps:Polygon Map 305, 322, 324  
 Selection:Maps:Select by Map 305, 322, 324  
 Selection:Maps:Select By Map Influence 305, 322, 325  
 Selection:Radial Select 307, 324, 328  
 Selection:Select Connected 303, 320, 321  
 Selection:Selection Switch 308, 325, 328  
 Selection:Select Loop 306, 323, 326  
 Selection:Select Outline 307, 324, 327  
 Selection:Select Points 306, 323, 326  
 Selection:Select Polygons 306, 323, 327  
 Selection:Select Ring 307, 324, 328  
 Selection Sets 308, 325,  
 Selection Sets:Surface to Parts 310, 327, 311  
 Selections:Select Powergons 307, 324, 328  
 Selection:UV Seam:Select UV Seam 305, 322, 325  
 View:Hide Invert 327, 311  
 View:Hide Selected 327, 311  
 View:Hide Unselected 327, 311  
 Viewports 312  
 Viewports:Fit All 314, 313  
 Viewports:Fit Selected 314  
 Viewports:Magnify 313, 312  
 Viewports:Pan 312  
 Viewports:Single View 315  
 Viewports:Zoom 313  
 Viewports:Zoom In/Zoom Out 315  
 View:Unhide 327, 311  
 Volume Selection 302, 309, 319  
 Viewports 312  
     Rendering Style:Resetting Tools 56, 26  
     Save View Presets 56, 26  
 Viewports, Modeler 55, 25  
 Viewport Titlebar 32  
 View Tab 312  
 Visual Browser 50, 57  
 VMap 232, 236  
     Renaming Weight Maps 255, 265  
 VMap Name 239, 244  
 VMap Selection 55, 24  
 Vmap Shortcut 59, 30  
 Vmap Type 239, 244  
 Volume Selection 302, 309, 319  
 Vortex Tool 120, 124

## W

Wedge Tool 74, 79  
 Weight Map 227, 233, 229, 237  
 Windows  
     Hide Floating Windows On/Off 55, 25  
     Hide Toolbar On/Off 55, 25  
     Layer Browser Panel 53,  
     Layer Navigation 52, 60  
     Layout Communication 52, 60  
     Multi-document Environment 52, 60  
     Multi-Layer Object Standard 52, 59  
     Vertex Map Panel 54,  
     VMap Selection. *See* Vertex Maps  
 Wrap Sphere 136,







