NewTek
8200 IH-10 West, Suite 900
San Antonio, TX 78230

LightWave 3D 5.6
Manual Addendum

Software License and Limited Warranty

PLEASE READ CAREFULLY BEFORE INSTALLING THIS SOFTWARE. BY INSTALLING 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 FOR A FULL REFUND.

1. Grant of License

The enclosed computer program(s) (the "Software") is licensed, not sold, to you by 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 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, 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, 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 "© 1998 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 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 AS TO THE SOFTWARE AND ACCOMPANYING WRITTEN MATERIALS AND LIMITED TO 90 DAYS AS TO THE MEDIA. 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. If you are a US Government end-user, this License of the Software conveys only "RESTRICTED RIGHTS," and its use, disclosure, and duplication are subject to Federal Acquisition Regulations, 52.227-7013 (c)(1)(ii). (See the US Government Restricted provision below.)

7. Trademarks

LightWave 3D, LightWave, Video Toaster, Video Toaster Flyer are trademarks of NewTek. Pennello is a trademark of XAOS Tools. SGI is a trademark of Silicon Graphics, Inc. DEC Alpha is a trademark of Digital Equipment Corporation. Pentium is a trademark of Intel. Sun is a trademark of Sun Microsystems, Inc. Wavefront is a trademark of Wavefront Technologies. 3D Studio is a trademark of Autodesk. MetaNURBS is a trademark of Fori Owurowa. 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) (l) (ii) of the Rights in Technical Data and Computer Software clause at 52.227-7013. Manufacturer: NewTek, 8200 IH-10 West, Suite 900, San Antonio, TX 78230.

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.

Table of Contents

INTRODUCTION

This addendum to the LightWave 5.5 User and Reference Manuals includes new features for version 5.6. Also incorporated are changes and new features from interim updates.

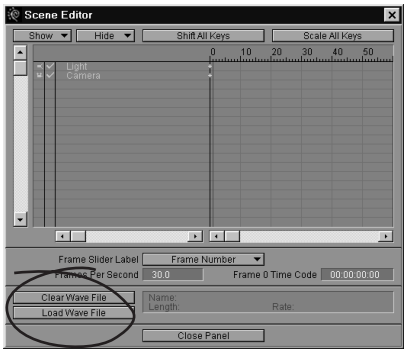
New Features for 5.6

- HyperVoxel plug-in for creating organic liquids and blobs
- Shader plug-ins for snow, rust, and water
- SkyTracer doubler plug-in for faster rendering
- Steamy shader plug-in to reflect and refract selected Steamer items
- SuperJittering plug-in
- Adobe® Photoshop® filter plug-in support
- Jolt! plug-in for physical-impact motion
- Fresnel shaders for altering reflective and refractive properties based on viewing angle
- Color-spectrum shaders based on viewing angle
- Surface attribute changing plug-in based on Z depth
- Halftone shader
- DPI conversion calculator plug-in
- QuickTime saving
- Object fog amount
- True and red/blue stereoscopic rendering
- Direct 3D support
- Selectable graphics library
- Image format saved to config file
- Ability to copy texture panel settings
- Most-recent plug-in shortcut
- Quick access to generic plug-in pop-up menu
- Most-recent channel on motion graphs
- New reflection and refraction options
- The maximum number of rendering threads has been increased to eight
- Data Overlay feature now supports lower-case letters
- The size and position of the main Layout window is now saved in the config file and restored the next time LightWave is started
- The children of items using the Match Goal Orientation option are now oriented as well
- New Loading Image option on all select image pop-up menus
- Much improved multiple-thread rendering
- ScreamerNet II now supports scene names containing spaces
- LScript release 1.3 (Documentation files are included.)

Layout

[NEW 5.6] Pressing the F5 key will bring up the most recently used plug-in panel.

Scene Editor Panel



Press CTRL+SHIFT+F1 to enable experimental features in Layout. This will add an option to the **Scene Editor** panel that allows you to load a .WAV audio file. This will playback when you play your Scene in the main Layout window.

WARNING! Experimental Features are not supported. Use them at your own risk.

Motion Graph

[NEW 5.6] When the motion graph editor is first opened for an item, the current channel will be **X Position**, but from then on it will open showing the most recently graphed channel for that item.

Note For information on interactively manipulating a motion path, see page 3.24 in the User Manual. This is not discussed in the Reference manual.

Objects Panel

Objects with the **Unseen by Camera** option active will still be "seen" by rays when ray-tracing. So, you will still see ray-traced shadows, reflection, and refraction.

Bones

[NEW 5.6] Pressing CTRL+R will toggle the activation status of the current bone. Previously, this could only be done from the **Skeleton** panel. Note that unlike the "r" shortcut, CTRL+R has no effect on the bone's rest position.

Morph Targets

To save memory, morph targets can be points-only, as long as point order is also maintained.

Object Fog

[NEW 5.6] The amount that fog affects each object can be adjusted by pressing "f" in the **Objects** panel. A value of 1.0 is normal, and 0.0 is equivalent to the **Unaffected by Fog** option. Fractional values work as expected, and values greater than 1.0 are allowed.

Surfaces Panel

Texture Reference Object

When using multiple texture layers, individual layers, as well as each surface attribute (i.e. Color, Diffuse, Specularity, etc), can use separate texture reference objects.

Texture Panels

[NEW 5.6] The current texture layer number is now shown in the title bar of all Texture panels. Options for a particular texture can be copied into a buffer by pressing CTRL+C. They can then be "pasted" into other texture panels by pressing CTRL+V. This technique also works in lens flare panels, but currently only static options are copied, not envelopes.

Alpha Channel

[NEW 5.6] **Constant Value Alpha Channel** option for surfaces should now be rendered with the specified alpha value no matter what transparency options they may have. This is true even if they are morphing into surfaces that don't use Constant Value.

Image Panel

[NEW 5.6] All image selection pop-up menus have an option to directly load an image.

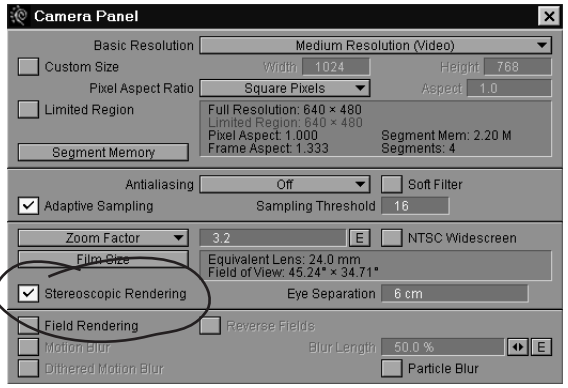
Camera Panel

Limited Region

[NEW 5.6] Pressing CTRL+L in the Layout window, while using the camera view, will toggle the state of the **Limited Region** option (**Camera** panel). Previously, this could only be done by accessing the **Camera** panel. The "l" key continues to activate the option and allow adjustment of the region border).

Stereoscopic Rendering

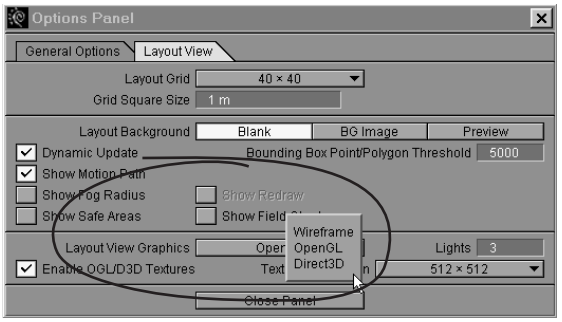
[NEW 5.6] A **Stereoscopic Rendering** option has been added to the **Camera** panel. When active, the **Eye Separation** distance can be specified. The camera should use a target object to make a viewer's eyes converge. Rendering with this option active will result in two images for each frame, which depict the scene as viewed from positions to the left and right of the single camera that would normally be used. The frame numbers in the filenames of the saved images will be followed by either "L" or "R" to distinguish these views. The **Eye Separation** input field defaults to 60mm, which is about the average distance between an average human's eyes.



HINT Use a null object as a Camera target and place it at the point where you want eyes to converge.

Options Panel

[NEW 5.6] Direct3D is now a standard feature in Intel versions. Additionally, the layout view graphics library (e.g., OpenGL, Direct3D, QD3D, etc.) is now selected through a popup menu. Select the **Wireframe** option to deactivate the solid-shaded display mode.



[NEW 5.6] The generic plug-in pop-up menu also appears on the main Layout interface.

Render Panel

[NEW 5.6] The output **RGB** and **Alpha Image Format** settings are now saved in the config file. The maximum number of rendering threads available in the **Multithreading** pop-up menu has been increased to eight. The **Data Overlay Label** field now supports lower-case letters.

Reflection and Refraction

[**New 5.6**] Previously, the colors of reflection and refraction rays were always clipped before being used in surface shading calculations. For example, the reflections of any objects seen in a 50% reflective mirror surface could be no brighter than 50%, no matter how bright the objects might be. This behavior is unrealistic in scenes with a high "dynamic range" of brightness, so ray color clipping is now turned off by default (except when older scenes are loaded). It can be toggled by hitting the "c" key in the **Render** panel.

Inverse Kinematics

[**New 5.6**] The children of items using the **Match Goal Orientation** option are now oriented as well.

Note For correct results, the children should have been added to the scene after the "goaled" item (which is usually the case).

Motion Plug-ins

LW_Jolt! (jolt.p)

[**New 5.6**] See the chapter on Jolt!

LW_SuperJitter (superjitter.p)

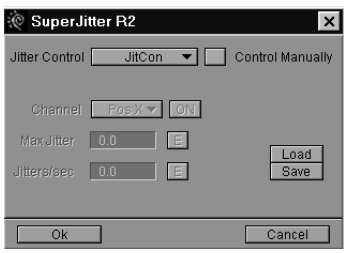
[**New 5.6**] This plug-in adds a random disturbance to the motion of an object, light, bone, or the camera. The range of disturbance added to each animation channel (i.e., position, rotation, and/or scale) of the jittered item comes from a defined control object, generally a Null, or can be set manually.
For example, by moving the control object's Y height to 3 meters and leaving all other animation channels at 0, the jittered object will have a random value between 3 and -3 meters added to its Y position during the animation.

The frequency of the jittering is determined by the control object's pivot point. Increasing the X pivot value increases the frequency of position changes, Y the rotation frequency, and Z the scaling frequency. The disturbance ramps smoothly between the random values.

Note The default size of a Null is 1,1,1. Thus, you should size the Null to 0,0,0 if you do not want LW_SuperJitter (or LW_JitterMotion) to affect the scaling of your object.

To use SuperJitter

1. Clear the Scene and load the Cow object.
2. Add a Null object and rename it by saving as "JitCon".
3. Open the cow's **Motion Graph** panel and select the LW_SuperJitter plug-in.
4. Click **Options** and select JitCon as the **Jitter Control Object**.
5. Return to Layout, select JitCon as the current object, and keyframe it with a Y value of 0.2.
6. Move its **Pivot Point** to an X of 30.
7. Choose the **Side** view and play the Scene. This is the cow in an earthquake.



By animating the **Jitter Control Object**, you can animate the disturbance range—in effect using each of the position, rotation, and scaling channels as an envelope on the target object's jitter.

To set animation channels manually, activate the **Control Manually** option. Select the channel you wish to affect using the pop-up menu and then click the **On** button. Enter the maximum amount of jitter in the **Max Jitter** field (in meters or degrees, as appropriate) Enter the number of jitters per second in the **Jitters/sec** field. Each animation channel can be on or off and have its own independent set of values.

Object Displacement Filter Plug-ins

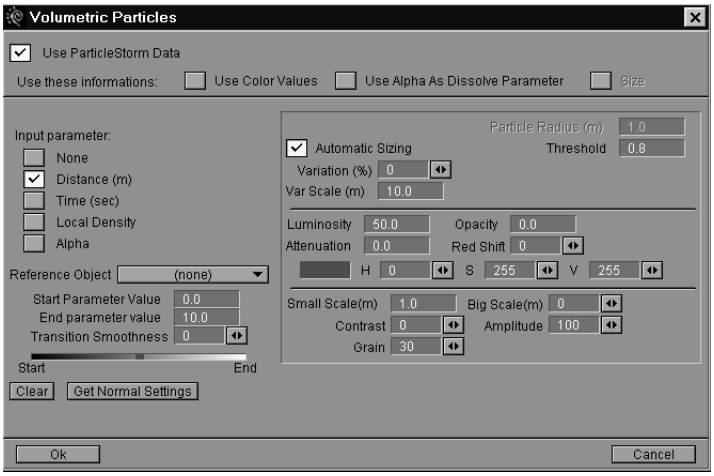
LW_HVRealFlow_Import (HyperVoxels.p)

[**NEW 5.6**] This plug-in allows you to use RealFlow particle displacement data for an object.

LW_HyperVoxel_Particles (HyperVoxels.p)

[**NEW 5.6**] See the chapter on HyperVoxels.

LW_Steamy_Particles (Steamer.p)



You no longer need to specify the (Particle Storm) PSM file. Steamy Particles gets the information about the object it's applied on. However, make sure that the PSPlay displacement plug-in precedes the Steamy Particles plug-in.

The (Particle Storm) Color option has been renamed **Use color values**. The (Particle Storm) Transparency option has been renamed **Use Alpha as dissolve parameter**. It still uses the alpha channel information from Particle Storm to dissolve each particle individually. The (Particle Storm) Rotation option has been removed.

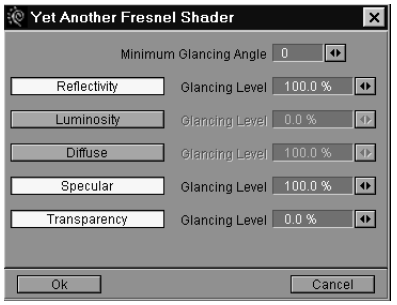
A new Input Parameter option has been added called **Alpha**. It allows the user to use the (Particle Storm) alpha channel information for each particle as an input parameter. For example, if you create a color ramp in Particle Storm with an alpha channel varying from 0 to 255 depending on age, you will be able to use the information with Steamer. You can also access and use other information, like speed or acceleration.

The **Transition Smoothness** control has a color ramp that shows the speed of the progression between the start parameter value and the end parameter value. The red line shows the position of the average values interpolated between Steamer's values and Steamy Particle's values. The ParticleStorm option will be activated by default, if the ParticleStorm displacement map plug-in is active.

Note You can now bring this panel up directly from the Steamer pixel filter plug-in panel by clicking the **Params** button.

Surface Filter Plug-ins

LW_FastFresnel (RealFresnel.p)



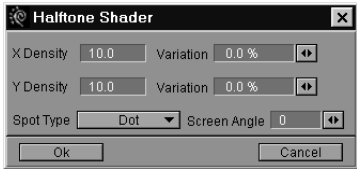
[NEW 5.6] The French physicist Augustin Jean Fresnel (pronounced “fra nel”) found that the angle between the viewer and a surface affects the amount of light that is reflected and refracted. **LW_FastFresnel** works in combination with the basic surface parameters set by the user and then modifies those settings based on slope angles and the **LW_FastFresnel** parameters.

Minimum Glancing Angle sets the range of slope angles that determine the strength of the surface attributes. If this is set to 0, the shader will vary the surface strengths over a range of 0 to 90 degrees. The angle (of incidence) is measured from the surface normal, thus 0 degrees is any surface normal that points directly at the camera and 90 degrees is where any normal is perpendicular to the camera.

The remaining options are alternative surface settings that will determine the range of values for the Fresnel shader to pass through while varying these numbers based on incidence angle from the camera’s perspective. Essentially, the LightWave surface settings change to the **LW_FastFresnel** settings as the angle of incidence goes from 0 to the **Minimum Glancing Angle**.

For example, if you had set the basic surface value for transparency to 100% and the **LW_FastFresnel Transparency** to 0%, the shader would create a transparency gradient from 100% to 0%, as the surface angles reached 90 degrees, assuming the **Minimum Glancing Angle** was set at 90. This will give the effect of the surface becoming less transparent as you reach the edge. The other attributes will go in the opposite direction, becoming more intense as you near the edge—reflections and specular highlights will naturally get stronger as you reach the higher angles of incidence.

LW_HalftoneShader (Halftone.p)



[NEW 5.6] This plug-in applies a series of patterns on a surface. The **X Density** determines how many shapes/marks should be placed per default (LightWave) unit (generally a meter) along the X axis. If the units are meters and the value is 10, there will be 10 shapes/marks per meter on the surface. **Y Density** works identically, except along the Y axis.

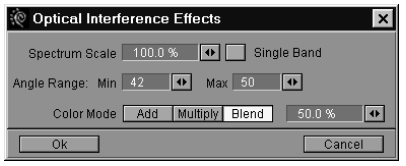
The **Spot Function** pop-up menu determines what shape or mark will be used: **Dot**, **Soft Dot**, **Block**, **Cross**, **Scale**, **Chex**, **Line**, or **CrossHatch**. **Screen Angle** allows you to set a rotation offset for pattern.

Note You may also apply this effect on a global basis using the **LW_Halftone** Pixel Filter plug-in.

LW_HyperVoxels_Shader (HyperVoxels.p)

[NEW 5.6] See the chapter on HyperVoxels.

LW_Interference (Interfere.p)



[NEW 5.6] **LW_Interfere** is a shader plug-in that adds the sort of distortion seen on an oil slick. This interference pattern is caused by the light reflecting between two layers of various thickness. In the case of an oil slick, it is the light reflecting between the water and the oil. This will often appear rainbow-like, where all spectral colors can be viewed swirling about. This plug-in adds all spectral colors in a banding fashion and uses incidence angles to vary which colors are visible.

Spectrum Scale determines how far through the color spectrum the shader will travel across the slope of the surface. This is dependent on the **Min** and **Max Angle Range** settings. For example, the default settings of **Spectrum Scale**=100%, **Min**=42, and **Max**=50 tells the shader to travel through the entire spectrum (100%) as the angle of incidence changes from 42 degrees to 50 degrees, or a delta of 8 degrees. The spectral range colors are red, orange, yellow, green, blue, indigo, and violet. If you were only to change the **Spectrum Scale** to 50%, the surface would only travel through red, orange, yellow, and green, across the same angle.

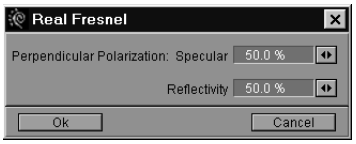
Activating the **Single Band** option will restrict the spectral change to a single ring between the **Min** and **Max** angles. In effect, this keeps the texture from repeating across the entire surface.

The **Add Color Mode** adds the color values of the interference pattern to the original surface colors. If a pixel was originally shaded at 100, 100, 100 and the interference color for that pixel was to be 0, 0, 100, the resulting pixel value will be 100, 100, 200. The **Multiply** option multiplies the original pixel values by some number between 0 and 1. If the original pixel value was 255 it is multiplied by one. If it was 0, the new pixel value is multiplied by zero. Intermediate values are altered on a sliding scale.

The **Blend Color Mode** blends the pattern with the original surface attributes using the **Blending** percentage. At 50%, the default, the interference pattern will be seen on top of the original surface with a strength of 50% of its own color values.

LW_RealFresnel (RealFresnel.p)

[**NEW 5.6**] The French physicist Augustin Jean Fresnel (pronounced “fra nel”) found that the angle between the viewer and a surface affects the amount of light that is reflected and refracted. This shader is based on real physics and thus features few user-definable controls. It is essentially set up to create a transparent item that utilizes the Fresnel equation to calculate falloff for the transparency value.



Specular and **Reflectivity** are the values for those surface attributes that will be used when the camera is perpendicular to the surface.

LW_Rust (NaturalShaders.p)

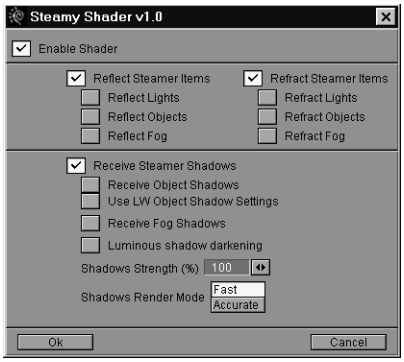
[**NEW 5.6**] See the chapter on Natural Shaders.

LW_Snow (NaturalShaders.p)

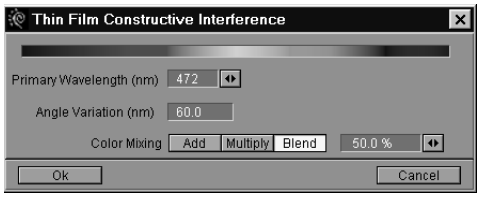
[**NEW 5.6**] See the chapter on Natural Shaders.

LW_Steamy_Shader (Steamer.p)

[**NEW 5.6**] This surface shader will allow the surface to which it is applied to reflect and refract selected Steamer items. You can also make the surface receive shadows from Steamer objects. The **Accurate Render Mode** will yield better looking results at the expense of rendering time.



LW_ThinFilm (Interfere.p)



[**New 5.6**] Similar in effect to **LW_Interference**, this plug-in also changes the color spectrum based on the surface's angle to the camera. It can be used for effects like an oil film on water, etc. **Primary Wavelength** is the color in the spectrum that the shader will use as it's base color. You may either enter the wavelength value or simply click on a color in the spectrum. **Angle Variation** is the angle at which the colors will start to shift.

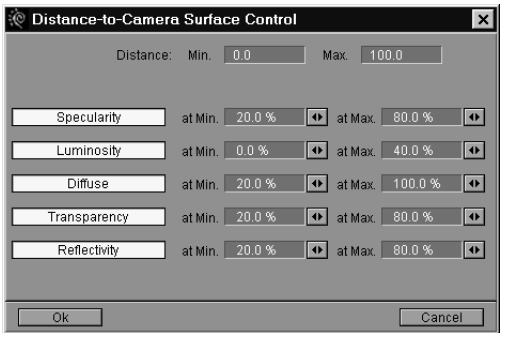
The **Add Color Mode** adds the color values of the interference pattern to the original surface colors. If a pixel was originally shaded at 100, 100, 100 and the interference color for that pixel was to be 0, 0, 100, the resulting pixel value will be 100, 100, 200. The **Multiply** option multiplies the original pixel values by some number between 0 and 1. If the original pixel value was 255 it is multiplied by one. If it was 0, the new pixel value is multiplied by zero. Intermediate values are altered on a sliding scale.

The **Blend Color Mode** blends the pattern with the original surface attributes using the **Blending** percentage. At 50%, the default, the interference pattern will be seen on top of the original surface with a strength of 50% of its own color values.

LW_Water (NaturalShaders.p)

[**New 5.6**] See the chapter on Natural Shaders.

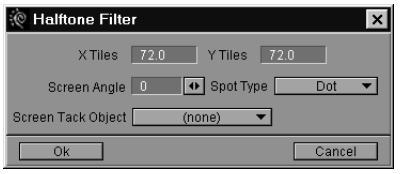
LW_ZShader (ZShader.p)



[**New 5.6**] This plug-in will vary selected surface attributes based on the Camera's distance from the surface. Set the **Min** and **Max Distance** fields to the Camera distance range you want the effect to happen over. (The units are measured in LightWave default units.) Select the desired surface attributes you want to change by clicking on their respective button(s). Set the corresponding **Min** and **Max** values you want to correspond with the **Min** and **Max Distance** settings at the top of the panel. Intermediate values will be interpolated linearly.

Pixel Filter Plug-ins

LW_Halftone (Halftone.p)



[**New 5.6**] This plug-in applies a series of patterns to an entire rendered image to give it a "halftone" look. The **X Density** determines how many shapes/marks should be placed per default (LightWave) unit (generally a meter) along the X axis. If the units are meters and the value is 10, there will be 10 shapes/marks per meter on the surface. **Y Density** works identically, except along the Y axis.

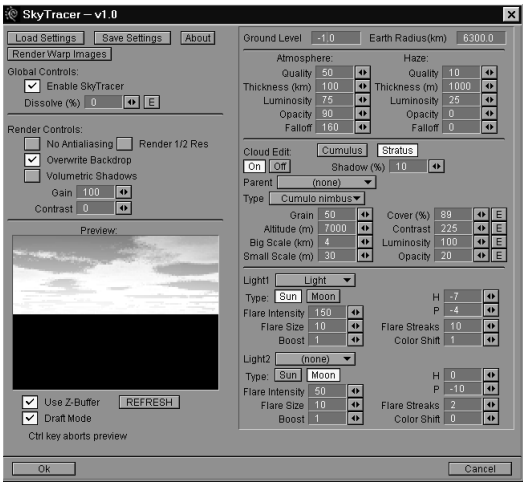
The **Spot Function** pop-up menu determines what shape or mark will be used: **Dot**, **Soft Dot**, **Block**, **Cross**, **Scale**, **Chex**, **Line**, or **CrossHatch**. **Screen Angle** allows you to set a rotation offset for pattern. Use the **Screen Tack Object** to attach the pattern's position relative to a specific object.

Note You may also apply this effect on a surface-by-surface basis using the **LW_HalftoneShader** Surface Shader plug-in.

LW_HyperVoxels_Particles and LW_HyperVoxels_Doubler (HyperVoxels.p)

[**NEW 5.6**] See the chapter on HyperVoxels.

LW_SkyTracer (SkyTracer.p)



[**NEW 5.6**] **Render Warp Images** renders a cubic image map that you can map onto a cube and place the camera in the center. You can use that method for your sky instead of having Skytracer render every frame.

You must now select a light before SkyTracer will render anything.

Ground Level is the location of the ground so that the plug-in can keep track of the Camera's elevation. **Earth Radius** can be changed from the default 6,300 km to create sky effects for different planets. This will affect how the sky is rendered. **OverWrite Backdrop** prevents the SkyTracer image from being additive to the current background.

LW_SkyTracer_Doubler (Steamer.p)

[**NEW 5.6**] The **LW_SkyTracer_Doubler** pixel filter plug-in must be loaded after **LW_SkyTracer**. (There may be plug-ins loaded in-between, but they should not be active.) It performs the second pass after SkyTracer's rendering pass when **Render ½ Res** is active. The default values give a good balance between quality and render time.

When **Adaptive Sampling** is inactive, the Doubler performs a straight interpolation between pixels. When active, the Doubler determines if each pixel needs to be recalculated or not, depending on its boundary pixels. This option is very similar to Lightwave's **Adaptive Sampling** option (**Camera** panel).

Sampling Threshold controls the edge detection that decides which pixel should be recalculated. Pixels below the threshold level will be recalculated, those above will be simply interpolated between neighboring pixels. Low values of threshold will make higher quality images with finer details but with higher render times. If Doubler creates aliasing problems on objects boundaries, the threshold should be set to low values (below 4).

Note If the Doubler is installed and **Render ½ Res** is inactive, the Doubler will be bypassed.

LW_Steamer

LW_Steamer features faster light cone rendering. The new sampling method is more accurate and reduces aliasing problems during animation. You can now get the same quality as before with four-times less samples.

It is now possible to work with large and complex particle objects, with up to 20-times faster rendering on large objects.

ParticleStorm rendering will be faster and require less memory. The ParticleStorm option will be activated by default, if the ParticleStorm displacement map plug-in is active.

Steamer now works internally using a two-pass process. To utilize **Render ½ Res**, you must now also activate the new **LW_Steamer_Doubler** plug-in. This plug-in will perform the

interpolation after Steamer's rendering pass. This implementation provides better quality and corrects aliasing problems. It also gives additional control over the interpolation quality.

When **Alpha Output** is active, Steamer saves the opacity values for each pixel to LightWave's internal alpha channel. The luminosity values are not saved, however.

The new **Render Mode** option allows you to determine the density of the Steamer effect on objects. In prior versions, a high number of particles in roughly the same position, overlapping each other, would result in overexposed "hot spots". This was because Steamer was adding the densities of each particle at each sampling point. This effect is still available by using the default **Additive** mode. The new modes were designed to keep density to more reasonable values preventing oversaturation.

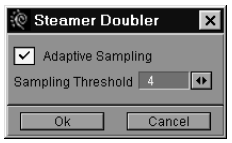
The **Maximum** mode uses the highest (i.e., maximum) density value from all of the overlapping particles at the sampling point. **FuzzyBlobby** also limits the density but slowly fades the effect towards its edge. This makes soft edges, good for simulating vaporous objects. **Fat&Blobby** works similarly, but has a sharper edge, which is good when trying to simulate a dense smoke.

Shadow Strength controls the contrast between light and shadow. Reducing this value will lower the contrast between dark and bright areas. Values higher than 100 will make more contrasted shadows. This option is activated when **Inner Shadows** is active. The **Inner Shadows** calculation continues to take into account the Opacity setting.

Local Coordinates is the opposite of the **World Coordinates** option in a standard Surface **T**exture panel. It attaches the fractal noise to the object being rendered, so that it moves with the object. This is especially useful for particles, since it attaches a piece of fractal noise to each particle. When animating, the noise will be calculated locally on each particle, which gives the best visual effects when trying to simulate smoke effects.

Noise Filter allows you to specify a fractal pattern, like **Bumpy**, **Fibers**, and **Smokey**. **Params** provides direct access to the **LW_Steamy_Particles** panel, if that plug-in has already been added.

LW_Steamer_Doubler (Steamer.p)



The **LW_Steamer_Doubler** pixel filter plug-in must be loaded after Steamer. (There may be plug-ins loaded in-between, but they should not be active.) It performs the second pass after Steamer's rendering pass when **Render ½ Res** is active. The default values give a good balance between quality and render time.

When **Adaptive Sampling** is inactive, The Doubler performs a straight interpolation between pixels. When active, the Doubler determines if each pixel needs to be recalculated or not, depending on its boundary pixels. This option is very similar to Lightwave's **Adaptive Sampling** option (**Camera** panel).

Sampling Threshold controls the edge detection that decides which pixel should be recalculated. Pixels below the threshold level will be recalculated, those above will be simply interpolated between neighboring pixels. Low values of threshold will make higher quality images with finer details but with higher render times. If the Doubler creates aliasing problems on objects boundaries, the threshold should be set to low values (below 4).

Note If the Doubler is installed and **Render ½ Res** is inactive, the Doubler will be bypassed.

Image Filter Plug-ins

LW_AnaglyphStereoCompose (StereoComposer.p)

[**NEW 5.6**] When this image filter plug-in is used in conjunction with the new **Stereoscopic Rendering** option (**Camera** panel). StereoComposer will save out the left and right channels "stitched" together as the right channel, and the normal RGB image as the left channel.

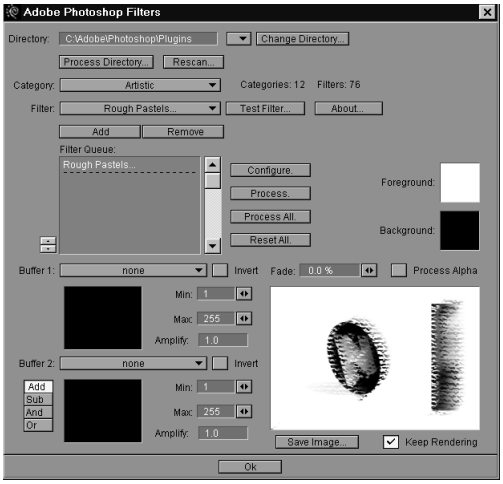
This can also be used with the **QuickTime-Stereo Animation Type** (**Render** panel) that is added by QTSaver.p (mac and intel only) to save out a stereo quicktime movie.

LW_EffectsTile (LW_Tiler.p)

LW_EffectsTile is an Image Filter plug-in. When used, it will tile thumbnail images of all of the special internal LightWave buffers used in the current scene over the rendered image. This includes the full-color, alpha channel, flat color, diffuse, specularity, transparency, texture, shadow, and Z buffers, but will vary from scene to scene depending on which options are being used.

LW_Photoshop_Filter (8bf.p)

[**NEW 5.6**] LW_Photoshop_Filter allows you to use your Adobe® Photoshop® filter plug-ins on rendered images.



Click the **Process Directory** button to create a catalog of filters. This list is saved as a file in the plugins directory. If the filter catalog file already exists, it is loaded and the directory is not reprocessed. To recreate the list, click the **Rescan** button. This refreshes the catalog file.

Use the **Category** pop-up menu to select the filter category (i.e., type). Then, select the desired filter from the **Filter** pop-up menu.

Clicking the **Test Filter** button allows you to test the filter without adding it to the rendering queue. If the filter has a dialog box, it will appear allowing you to adjust its settings. The results will appear in the preview window. The **About** button will display info about the plug-in.

Note Perform a quick render before entering this panel (press F9) to place a picture in the preview window. Otherwise, a default image will be used. The **Keep Rendering** option must also be active.

Click the **Add** button to add the selected filter to the Filter Queue window. You can add the same filter more than once and give each unique settings. To remove a filter from the window, first select it by clicking on it, then, click the Remove button.

The parameters for a filter can be set by selecting it and then clicking the **Configure** button. Its dialog box will then appear.

Note You must configure all filters with settings, even if you will be using the default values. If a filter has not been configured, it will request the information whenever it is trying to process the image.

Clicking the **Process** button applies the selected filter (only) and shows the results in the preview window. To show the results of all filters in the window, click the **Process All** button.

Filters are applied in order, top to bottom. To rearrange the order, select a filter from the list and then click the up or down arrow buttons located to the left of the **Filter Queue** window. This will move the filter up or down in the list, respectively.

The **Reset All** button will reset all parameters and preview image.

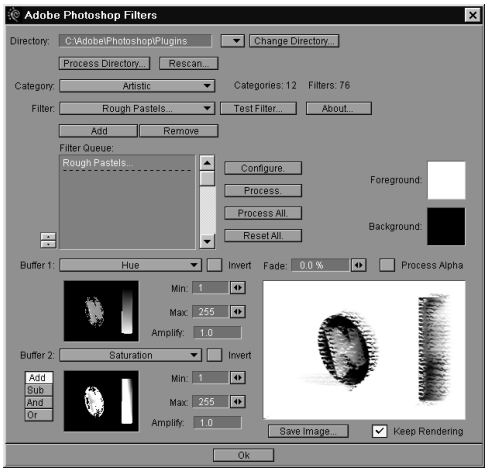
The **Fade** percentage specifies the percentage of the original image that should be composited over the image being processed by the filter.

The **Foreground** and **Background** colors are set individually for each filter in the **Filter Queue**. These are the same as their counterparts in Photoshop® and may be required for some filters. Click on either color swatch to display a color selection requester. These are foreground and background colors selected in Photoshop®.

The **Process Alpha** option makes the filter use LightWave's alpha channel as a Photoshop® transparency mask.

Note This option can cause problems with some plug-ins; however, if the filter supports alpha channel, this option should be used as it may speed up processing time.

Buffers



You can use two buffers (**Buffer 1** and **Buffer 2** pop-up menus) to setup a particular region in the image you want to be processed by the filter. You can select one buffer or two buffers. If two buffers are selected, you can specify how you want them combined.

The **Invert** option will reverse the data in the source buffer.

The **Min** and **Max** values lock buffer input to this range. For instance, if we are looking at the depth buffer, selecting values 0-128 will lock the buffer and only affect the closest objects. If the image contains areas where colors are oversaturated, setting the range to 128-255 will process only highly saturated areas.

The **Amplify** buffer output is used to correct buffer input before processing. For example, if you are using the Red channel of the image, the brighter areas will indicate the areas where the filter processing will be stronger. If you want the filter to process all red areas of the image evenly, amplify the buffer until output becomes bright white.

Buffer Compositing Options

With **ADD**, regions are overlapped producing a region which includes areas of both source buffers. With **SUB**, the second region is subtracted from the first one. If second region is present in areas of the first region, these areas will no longer be covered by the first region.

With **AND**, regions are combined producing a region which covers regions only common to both source regions. With **OR**, regions are combined producing a region which covers regions of the source buffers excluding areas where these regions are overlapped.
Save Image allows you to save the processed image while still working in the plugin preview.

Activate the Keep Rendering option if you want to utilize the last rendered image (if available) for use in the preview window. This is active by default.

Note Some filters may not operate properly outside of the Photoshop® environment.

Note Only Photoshop® plug-in filters will be a available. You will not be able to access built-in filters (e.g., Gaussian Blur, etc.)

Note Many filters (mostly 3rd party) do not provide their information in such way that it can be saved in a scene file.

ScreamerNet

Generally, this filter will not work with network rendering. You will have to use a trial and error approach to determine which filters will work. Make sure that the (.8bf) filter files are by all nodes. This can be accomplished by mapping a network drive using the same drive letter on all nodes, copying filters to the same location on both computers, or using the DOS SUBST command.

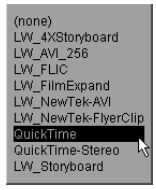
LW_StereoAnaglyph (StereoRB.p)

[NEW 5.6] This image filter plug-in "fakes" having the **Stereoscopic Rendering** option turned on in the **Camera** panel by making the image a stereo image. Use 60mm in the **Options** dialog box for eye separation, which is about the average distance between an average human's eyes.

Note You must render in one segment to use this plug-in, so you may have to adjust your **Segment Memory** (**Camera** panel) accordingly.

Render Panel Plug-ins

QuickTime (QTSaver.p)



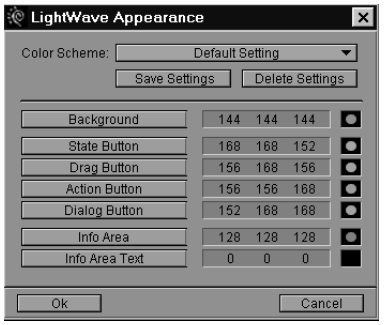
[NEW 5.6] This adds a **QuickTime** option to the list of **Animation Types** (**Render** panel) for Windows machines. When you save your animation, a standard QuickTime dialog box will appear, where you set your compression and quality settings.

Note This plug-in requires QuickTime 3.0 or later.

Other Layout Plug-ins

LW_ColorChanger Plug-in (colorchange.p)

The **LW_ColorChanger** plug-in, located in the Layout **Generic Plug-ins** pop-up menu (**Options** panel, **General Options**), allows the user to change the color scheme used by Layout and Modeler. Simply run the plug-in, and select the colors you wish to use.



The **Color Scheme** pop-up will display the available Color Schemes, which are files saved in the current Content Directory.

Save Settings saves the **Color Scheme** to a file. Choosing **Overwrite existing file** will write over the settings file selected in the **Color Scheme** pop-up menu. Choosing **Save as a new file** will display a requester allowing you to enter a unique name for your color settings. The file will be saved as LWCOM<NAME>.CFG in the current Content Directory.

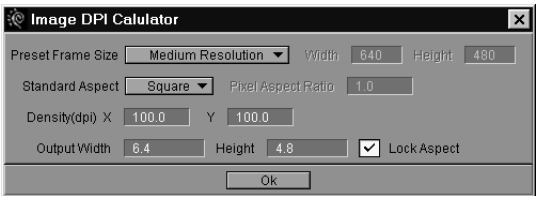
You will have to manually copy the **Color Scheme** file from the current Content Directory to the NEWTEK\PROGRAMS folder and rename it LWCOM.CFG, if it isn't already named that.

Delete Settings will delete the Color Scheme file selected in the **Color Scheme** pop-up menu from your hard drive.

Click on one of the following buttons to display an RGB color requester to change the appropriate color. **Background** is the general shade of the background (default: gray). **State Button**s that are either on or off (default: yellow). **Drag Button**s are for pop-up menu, slider, and drag buttons (default: green). **Action Button**s perform an instant action, like Clear, Add, and value reset (default: red). **Dialog Button**s call up a requester or dialog box (default: blue). **Info Area** is the background color for information fields (default: gray). **Info Area Text** is the text color for information fields (default: black).

Note The actual color used in the interface may be different than the one selected. LightWave will internally limit certain color settings.

DPI_Calc (DPI_Calc.p)



[**NEW 5.6**] DPI_Calc is a generic plug-in that will calculate the **Camera** panel settings necessary to achieve a specific DPI (or the vice versa). This is a helpful tool for people who use LightWave for print work.

To calculate the required **Camera** panel settings, set **Standard Aspect** to **Square** and make sure **Lock Aspect** is not active. Then, enter the desired **X** and **Y DPI** values, and **Output Width** and **Height** (in inches). The necessary resolution will appear in the upper half of the panel. If you have the **Lock Aspect** option activated, the **Height** will automatically change when you change the **Width** value, in order to maintain the same image proportions.

To calculate the print output values for any **Camera** panel settings, set the desired **X** and **Y DPI** values. Then, set the **Standard Aspect** and **Preset Frame Size**. The resulting output values will appear in inches.

Note The **Standard Aspect** actually only affects the **Preset Frame Size Width** and **Height values** for settings other than **Custom**.

Motion Capture Load (Mocap.p)

This plug-in allows you to load Acclaim or Wavefront motion capture files into LightWave. First, load the Acclaim .ASF or Wavefront .CAP file using the normal Load Scene requester. A special requester will then appear. In the requester, specify the associated Acclaim .AMC file or Wavefront .MOV (motion data). You will also specify a target LightWave scene filename to use.

You may also specify a BOD file. This is a LightWave scene file containing the object with bones and hierarchy already set up. If you do not specify a BOD file, a skeleton will be created from the .asf file data using Null objects. You will then need to replace the Nulls with the appropriate object files.

Next, an object association requester will appear. Here, you should select which motions go with which objects.

LW_Morph_Gizmo Plug-in (Gizmo.p)

Typing: "wave" when the interface comes up, will activate the experimental feature that allows you to load a WAV audio file. This will playback when you play your Morph Gizmo animation. Note that there is not a particular input field for you to type the word 'wave' into, just type the word with the interface appearing.

WARNING! Experimental Features are not supported. Use them at your own risk.

System Plug-ins

System plug-ins are plug-ins that operate at a system level and are typically not directly accessible by a user. They work in conjunction with other plug-ins and should, therefore, always be added.

Note System plug-ins are added in the same manner any plug-in is added.)

EnvAccess.p

[**NEW 5.6**] This system plug-in provides envelope editor services to plug-ins using lightwave panels.

gColorPicker.p

[**NEW 5.6**] This system plug-in provides color picker services to plugins.

Gradient.p

[**NEW 5.6**] This system plug-in provides shader editing services to plugins.

Preview.p

[**NEW 5.6**] This system plug-in provides render preview services to plugins.

Modeler

Deselecting

To deselect points or polygons press the / (backslash) key. This is listed as **Drop Tool** in the 5.5 documentation and **Edit Keyboard Mapping** panel. When tools are active the result will vary. For example, if a primitive has been outlined, pressing the key will reset the drawing. Also, if an area of influence is set using, say, magnet, the area will be reset as well.

Keypad Assignments

We have included default keypad assignments for Modeler: 1= face view; 2= patch polygons off; 3= side view; 4= face/top view; 5= default view; 6= side/preview view; 7= top view; 8= patch polygons on; and, 9= preview view. Of course, these can be changed by the user.

Exporting Multiple layers

If multiple foreground layers are selected, all geometry in those layers will be exported when using the **Put** function.

Modeler Plug-ins

LW_Julienne

Do not use this plug-in on a 2D object.

LW_PathToMotion (Motions.p)

If no points are selected, the motion path is generated in the order the points were created. You can also select the points in the order in which you want the path to go.

Configuration Files with WinNT

When a user runs LightWave for the first time, the .cfg files from the Newtek\Programs folder will be used. However, new configuration files will be written to the user's WinNT Profiles directory, which will be utilized the next time the user runs LightWave. This allows multiple users on an WinNT system to have independent LightWave configuration files.

Manual Typos

Reference Manual:

Page 3.12, paragraph one: “The size of a bone’s influence **are** determined...” should read “The size of a bone’s influence **is** determined...”.

Page 3.20: The images for Self Shadow On and Self Shadow Off are reversed.

Page 4.12: The correct value for Dougbrainium is 3.038.

Page 5.4, paragraph two: “Texture Falloff determines the rate at which a texture falls off and **become** non-existent.” should read “Texture Falloff determines the rate at which a texture falls off and **becomes** non-existent.”

Page 5.5, paragraph five: “This is useful if a camera is going get close to the mapped surface.” should read “This is useful if a camera is going **to** get close to the mapped surface.”

Page 15.20, paragraph two: “Metamation **metaforms the cage object and uses the new set of points to reshape** the meta object.” should read “Metamation **replaces the cage object with** the meta object.”

Page 15.29, paragraph four: “The images are **a** 256-level grayscale representations...” should read “The images are 256-level grayscale representations...”.

Page 17.2, paragraph one: “...the color of sky comes from the scattering and absorption of light **if** the different layers of the atmosphere” should read “...the color of sky comes from the scattering and absorption **of** light in the different layers of the atmosphere”.

Page 17.8, paragraph one: “This value should be set with regard the speed of particles.” should read “This value should be set with regard **to** the speed of particles.”

Page 17.8, paragraph six: “The falloff direction is different for different **item**.” should read “The falloff direction is different for different **items**.”

Page 17.11, paragraph three: “Values from 3 to 5 **area** usually adequate.” should read “Values from 3 to 5 **are** usually adequate.”

User Guide

Page 3.11, step 9: “At a **-1.0** setting, the Camera goes directly to frame 15. At **1.0**, it takes the long route” should read “At a **1.0** setting, the Camera goes directly to frame 15. At **-1.0**, it takes the long route”

Page 4.31, paragraph one: “Also, **are** series of parented Null objects...” should read “Also, **a** series of parented Null objects...”.

Page 5.14, paragraph one: “A Linear light can be best thought of as a sizable straight **string of Point** lights. An Area Light is like a rectangular **array of point lights**. **As such,** these lights send out light **equally in all directions**.” should read “A Linear light can be best thought of as a sizable straight **tube light**. An Area Light is like a **rectangular-shaped light**. These lights send out light **consistent with their shape**.”

Page 11.9, paragraph six: “Explode07**8**9” should read “Explode07**6**9”.

Page 13.7, bottom note: “Moving objects between Layout and Modeler automatically **saves** the object file.” should read “Moving objects between Layout and Modeler **does not** automatically **save** the object file.”

Tech 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 Aura doesn't seem to be functioning as it should, please contact technical support in one of the following ways:

- By email: ltech@newtek.com
- By fax: (210) 370-8030
- By telephone: (210) 341-8444. Technicians are available to answer questions from 8:00 AM to 8:00 PM Central Time, Monday through Thursday, and Friday from 8:00 AM to 5:30 PM Central Time.

Please supply in your communication or have the following information handy when calling:

- Your computer's operating system and version (e.g., Windows NT v4.0, Windows 95, etc.)
- The version of LightWave 3D you are using
- The amount of RAM in your computer
- Any relevant specifics about your system (display card type, memory managers, accelerator type, etc.)
- Your product serial number

Note Your product must be registered before you can receive support.

Feature Requests and Bug Reports

For bug reports, send e-mail to: lwbugs@newtek.com
For feature requests, send e-mail to: lwfeatures@newtek.com

HyperVoxels

With HyperVoxels you can create “organic globs” around points. These globs will attach to one another when they get close, much like the substance in lava lamps. You can apply various surface attributes to a HyperVoxel texture, much like a normal object surface; however, since there is no actual geometry, it is all handled through the plug-in.

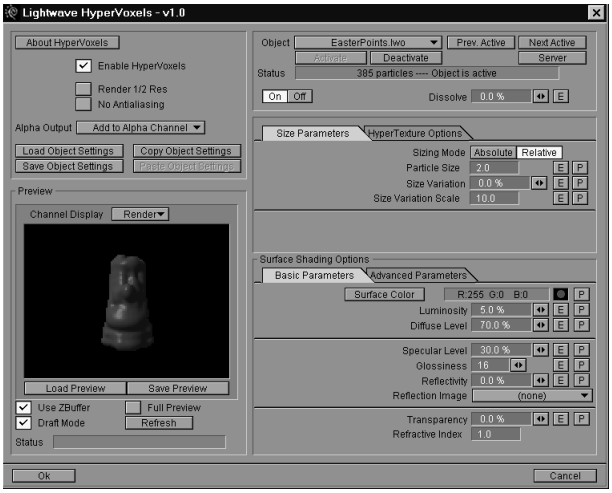
Even though there is no actual geometry, HyperVoxel textures are three-dimensional algorithmic textures, unlike normal surface bump maps, which only appear to have depth. This means you can actually get real close to the surface and they will look 3D no matter what angle you view them at. HyperVoxels feature subpixel displacement, which results in surface details no matter how close you get.

HyperVoxels consists of four plug-ins which work as a team. LW_HyperVoxels_Particles is a displacement map plug-in that provides information about objects to the main LW_HyperVoxels Pixel Filter plug-in. LW_HyperVoxels_Shader can be applied to object surfaces to have them catch shadows from HyperVoxels, as well as reflect and refract them. LW_HyperVoxels_Doubler works with the main plug-in to reduce rendering time.

LW_HyperVoxels (Pixel Filter plug-in)

This is the main HyperVoxels plug-in.

Global Controls



Enable HyperVoxels activates the plug-in. This allows you to turn it off without losing any settings.

Render 1/2 Res makes the plug-in render in half resolution and then interpolates to create the full-resolution effect. This will substantially increase rendering speed, at the cost of some accuracy.

Note The HyperVoxels_Doubler pixel filter must be loaded after HyperVoxels.

No Antialiasing keeps HyperVoxels from calculating during LightWave’s antialiasing passes. This will improve render speeds at the expense of image quality, motion blur and depth of field.

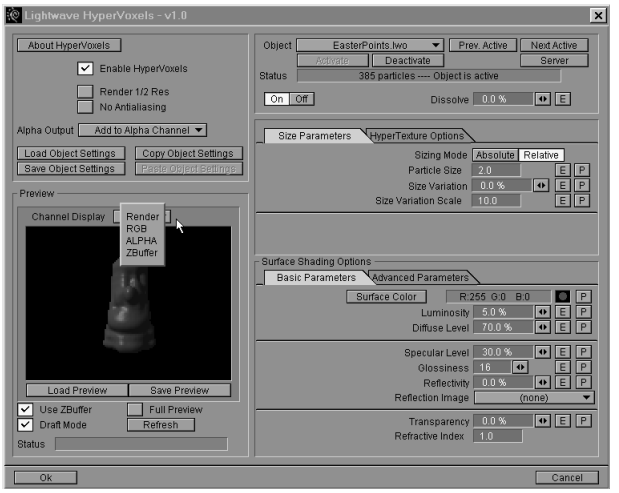
The **Alpha Output** pop-up menu lets you affect LightWave’s alpha channel. **None** will not affect the alpha channel at all. **Add to Alpha Channel** simply adds the HyperVoxels data to the alpha channel along with the LightWave generated alpha channel. **Replace Alpha Channel** will result in an alpha channel that only contains data for the HyperVoxels alpha—this can be used to create a mask for the HyperVoxels effect only.

Load/Save Object Settings allows you to retrieve and store HyperVoxels setting files. This can be used to create a library of attributes.

Note These settings are for the objects only and do not include global controls.

The **Copy/Paste Object Settings** buttons allow you to retrieve and store HyperVoxels parameters from a RAM buffer (without saving to disk).

Preview Controls



The **Channel Display** pop up menu allows you to view any of the following channels; **Render**, **RGB**, **Alpha**, and **Z Buffer**. **Render** is the combination of any previously rendered LightWave material with the HyperVoxels surfaces mixed in. The **RGB** option shows only what was rendered directly inside Layout in its color representation. **Alpha** displays any alpha channel for a previously rendered frame inside of Layout—this will only show the alpha for geometry and not the HyperVoxels effect. **Z-Buffer** displays the depth information for the previously rendered LightWave image.

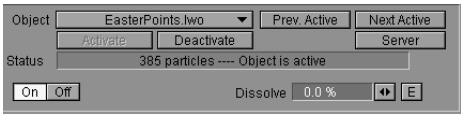
Note The last three options will not display any result until the user has rendered a frame from Layout.

The **Load/Save Preview** buttons allow the user to retrieve and store preview images, as a reference for certain settings. For example, you could make a library of settings by saving the **Object** settings and then save an image to match each saved parameter to use as a catalog. These previews can then be loaded for display and selection.

The **Use Zbuffer** option will allow HyperVoxels to display rendered LightWave material in the preview window.

Draft Mode renders half-resolution in the preview window. **Full Preview** will show all HyperVoxels items in the preview window. When this is inactive, only the item currently being edited is previewed. Click the **Refresh** button to preview the HyperVoxels effect.

Object Controls



The **Object** pop up menu selects the object to receive the HyperVoxels surface. The object must also be "activated", by clicking the **Activate** button. Clicking the **Deactivate** button will deactivate the object whose name appears on the **Object** pop-up menu. The **Previous Active** and **Next Active** will cycle you through all activated objects.

The **Server** button is a quick access option to get to the **LW_HyperVoxels_Particles** plug-in panel for the selected object.

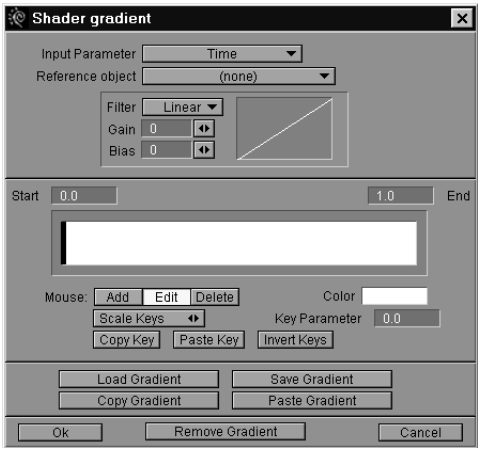
Dissolve allows you to set an opacity value for each individual item that is receiving a HyperVoxels surface: 0% is totally opaque and 100% is completely transparent.

Envelopes and Parameter Controllers

The **E** buttons are merely an envelope controller, which operate just like their counterparts in LightWave Layout. These allow you to vary values over time with spline based results.

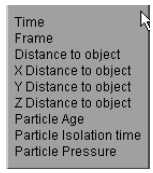
The **P** buttons are parameters controllers. These allow you to create "gradient ramps" that can either be used as color gradients for shading or for grayscale ramps to create value changes. The gradient editor is similar in some respects to the standard LightWave envelope editor. It allows the user to control a value depending on an external input parameter. For envelopes, the parameter is the frame number, for gradients, the parameter can be things like distance to an object, slope, bump height, etc. Like with envelopes, you have the ability to create, edit and delete keys. (A "key" is characterized by two numbers: its value and its parameter.). The input parameter can also be filtered to change the overall behavior of the gradient.

The **Input Parameter** defines what item will dynamically control the parameter. This allows you to control a setting beyond merely enveloping static values. Selecting **Time** changes the value over a specific time interval and the gradient bar unit is seconds. When set to **Frame**, the units are frames (essentially making this operate like a normal envelope).



Note The list of parameters varies depending on the particular setting the gradient is being applied to.

Distance to object uses the **Default Unit** (**Options** panel, **General Options**) as the gradient bar unit and changes the parameter independently based on the distance from each HyperVoxel point to the selected **Reference Object**'s pivot point. **X-**, **Y-**, and **Z Distance** to object work similarly but measured only along the respective axis. The **Reference Object** is only used for input parameters involving distance to an object.

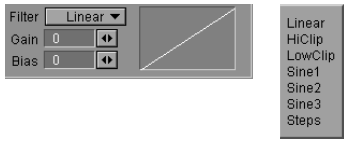


Note Since distance is measured from the point location to the **Reference Object**'s pivot point, you can use the same object as the **Reference Object**.

Slope angle changes the parameter based on the angle of the HyperVoxel surface in degrees. **Bump** uses the height of surface using the LightWave **Default Unit**. **Incident angle** uses the angle of the surface relative to the Camera in degrees.

Note Other input parameters may be available depending on how the related Displacement Map panel is set. For example, you can use data from external plug-ins like ParticleStorm and RealFlow.

The **Filter** modulates the selected **Input Parameter** and allows you to have it applied at a rate other than linearly, the default. For example, a linear curve is straight starting at the bottom left corner of the graph and ending at the top right. The input parameter is applied evenly over the gradient bar. If you used a setting that bowed the curve up, the input parameter would change faster at the steeper areas and slower where it is less inclined. If you were using **Time** as the **Input Parameter**, you would be scaling time to be faster at the beginning and slower at the end of the gradient. (If you had a curve that was declining, the input parameter would be reversing.)



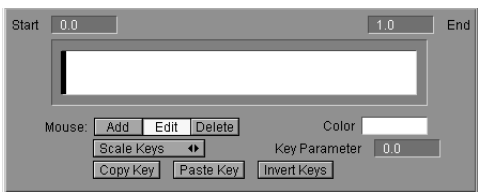
Note Since the Filter is only modifying the input, it has no (visual) effect on the gradient bar.

Adjusting **Gain** will make the curve(s) more or less pronounced S-shaped—the slope of the curve(s) is adjusted at the beginning and end in opposite ways. **Bias** bows the curve(s). These settings can be used together and negative values are allowed.

The various **Filter** options (**Linear**, **HiClip**, **LowClip**, **Sine1**, etc.) provide different basic curve shapes. **LoClip** clips off low values and stretches the high values. **HiClip** clips off high values and stretches the low values. The **Sine** options create oscillating or undulating values.

The Gradient Bar

The Gradient Bar graphically displays the key values set by the user. If the parameter is a color, the bar will be color and you will see color transitions from one key to the next. If the parameter relates to numeric values, the bar will appear in grayscale. The lowest set key value will be black and the highest will be white.



The gradient colors will automatically adjust if you add or change a key value that exceeds the previous minimum or maximum.

The **Start** and **End** fields set the beginning and end of the visible area of the gradient. There can still be keys set outside this range, however. The **Filter** is applied to this range.

The Mouse buttons determine what happens then you click on the gradient bar. If **Add** is active, a key will be created at the point to click. If **Edit** is active, you can grab an existing key and drag it left or right, but not past any other existing key. If **Delete** is active, a key will be deleted when clicked on.

The current key is shown in the gradient bar in orange. All other keys will be blue. Its value will be shown in the **Key Value** field, which can be edited. If the parameter is a color, this will be a color selector button instead. The **Key Parameter** field indicates the numerical position of the current key. Note that this value is not limited to integers.

Note You can use the left and right arrow keys to select the preceding or next key as the current key. However, make sure that you do not also have an input field active.

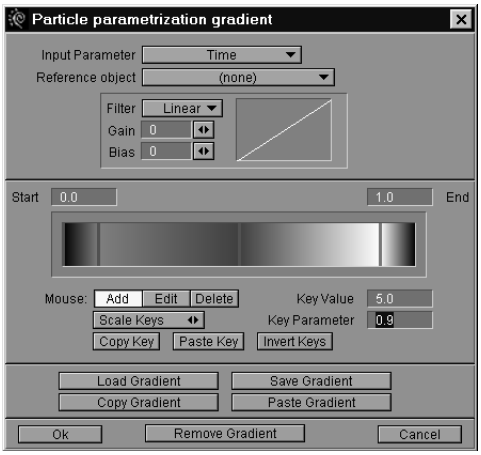
If you drag on the **Scale Keys** quick drag button, you will scale all keys in the gradient bar.

The **Copy Key** button copies the value/color of the current key to a memory buffer. The **Paste Key** button pastes the buffer into the current key.

Use the **Load Gradient** and **Save Gradient** buttons to load and save the gradient bar keys as a file. The **Copy Gradient** button copies the keys to a memory buffer. The **Paste Gradient** pastes the buffer into the current key. Note that this will overwrite any existing keys.

Clicking **OK** closes the panel and saves any settings. **Cancel** forgets any changes made to the panel. **Remove Gradient** will turn this option off for the parameter.

Using Gradients



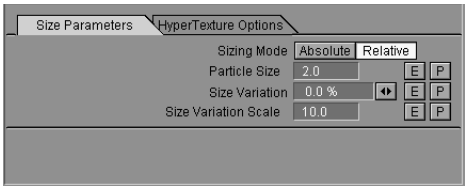
To change the particle size over time using a parameters controller, first create some keyframes. Then, change the values of the keyframes. You will see the gradient bar change. This is a reflection of the numeric values changing, but visualizing them in the form of color gradients—similar to using a grayscale image as a bump or displacement map.

If you were creating a fountain, for example, you would want the particles near the emitter to bond together and as they separate become smaller. Use a gradient Parameter to adjust particle size based on the distance to the base object (emitter). When the particles are emitted (0 distance from the emitter) they will have a user defined size and as they move away from the emitter the particles will shrink to the second user defined size. This would be accomplished by creating two gradient keys, one at 0 distance and one at 1 meter (if the fountain spanned a 1 meter distance). The first key value is set to the larger size (1.2 for example) and a second key is created at distance of 1 and set to a key value of 0. As the particles move further away they shrink.

Parameters can also be very useful for surfacing an item. To create a rusty surface, you can use the gradient parameters to vary surface color based on bump. This will allow different levels of the bump to have different colors with a nice blend between. For example, a reflective surface might change to burnt orange as it approaches the tip of the fractal

bumps. This is accomplished by adding values to the gradient that start one color and move to the burnt orange.

Size Parameters



The **Absolute Sizing Mode** sets a specific radius for each particle (**Particle Size**) in the default unit (usually meters). The **Relative** mode computes an average particle size based on the shape of the entire object. In this mode, a **Particle Size** of 1 means the exact equal to an internally calculated "average" size—HyperVoxels surfaces will not intersect at this setting. A value of 2 would mean twice the average and .5 would be half. A new value is dynamically computed for every frame, thus if the particles are going to move, the computed particle size will likely change from frame to frame.

Switching from **Relative** to **Absolute** will display the last calculated particle size, so this is a handy way to find a starting point when using the **Absolute** mode.

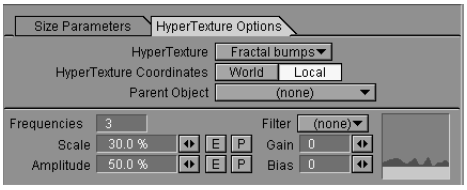
HINT The **Relative** mode simplifies a lot of the work and should be the preferred mode for less advanced users and when setting up HyperVoxel settings. Rendering time is related to how much the particles intersect each other—the more you have intersections, the longer the rendering time. As such, begin with the **Relative** mode, and tweak it until you achieve the desired results. Then, (in most instances) switch to the **Absolute** mode for your final rendering.

Size Variation sets the maximum percent that the particle size can vary. For example, a particle size of 1 with a variation of 100% can be as small as 1 and as large as 2. Using a variation of 50% would yield particles ranging from 1 to 1.5 in size. Use this to create random disturbances in the surface.

Size Variation Scale randomly varies the **Size Variation** as particles move. Think of this as a large invisible 3D field. Using a value of 10 would allow a particle to vary from its original size to the Size Variance over 10 meters as it moves through this field. A setting of 1 would vary the size much more quickly.

Note Changing either size variation setting from 1 is only useful when the particles are animated. Thus, you will not see much difference in the preview window.

HyperTexture Options



The **HyperTexture** pop-up menu determines what the texture will look like. If you look at the small graph located just to the right of the **Filter** pop-up menu, you will see a general cross-section view of the selected texture.

For **HyperTexture Coordinates**, select **Local** to have the surface attached to the object. Using **World** will cause the object to "move through" the texture.

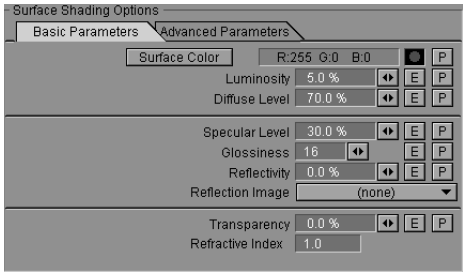
You may select a **Parent Object** using the pop-up menu. This works just like a normal surface reference object, so you can move, scale, and rotate the parent object to animate the texture.

Increase **Frequencies** to increase the detail in the texture pattern. **Scale** sets the relative size of texture and **Amplitude** sets the intensity (i.e., height) of texture. You will notice the cross-section graph change as you adjust the later two parameters.

On the **Filter** pop-up menu you have several options that affect the smoothness of the transitions between the texture's peaks and valleys. The cross-section graph will change as you adjust these parameters. The various **Filter** options (**HiClip**, **LowClip**, **Sine**, etc.) provide different basic curve shapes. **LoClip** clips off low values and stretches the high values creating low harsh valleys. This produces a non-uniform texture with spots on an otherwise smooth surface. **HiClip** clips off high values and stretches the low values, creating harsh plateaus. The **Sine** option creates undulating values for individual curves.

Adjusting **Gain** will make the curves of the texture more or less pronounced S-shaped—the slope of each curve is adjusted at the beginning and end in opposite ways. This tends to erode the peaks and fill in the valleys of the texture, if **Gain** is increased, and emphasizes peaks and valleys (eventually flattening them out), if **Gain** is decreased. Decreasing **Bias** lowers everything—faster for peaks—ultimately to the point where everything is flattened at the minimum value. These settings can be used together and negative values are allowed.

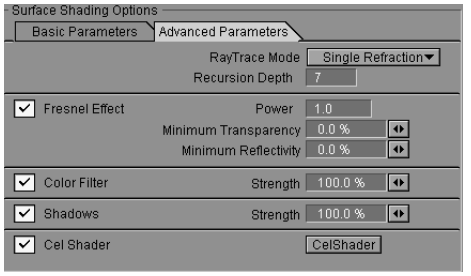
Basic Parameters



These parameters work just like their counterparts in the normal **Surfaces** panel.

Advanced Parameters

The **Full Refraction RayTrace** mode forces HyperVoxels to bounce rays each time a new refracting (i.e., transparent) surface is found, until it reaches the limit number set as the **Recursion Depth**. **Single refraction** mode only renders one refraction, which is enough most of the time. **Full Refraction** will, of course, increase render time, but will result in more sophisticated effects.



The **Fresnel Effect** option (pronounced "fra-nel") affects textures with some level of transparency and/or reflectivity. Essentially, these surfaces are more opaque and reflective when viewed from the side and less so when viewed perpendicular (i.e., head on) to the surface. As such, this option varies transparency and reflection based on the angle of incidence.

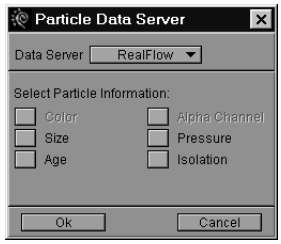
The **Power** setting sets the overall amount of effect you want. The **Minimum Transparency** and **Minimum Reflectivity** set the minimum amounts of transparency and reflectivity when the surface is viewed head on.

Color Filter works just like its counterpart on the normal Surfaces panel and tints objects seen thorough a transparent surface.

The **Shadows** option lets HyperVoxel objects cast self-shadows. To have them cast shadows on other objects, you must use the **LW_HyperVoxel_Shader** surface shader plug-in.

The **Cel Shader** options lets you use the LW_SuperCelShader on HyperVoxel objects. Note that you do not need to load the LW_SuperCelShader plug-in separately.

LW_HyperVoxel_Particles (Object Displacement plug-in)



LW_HyperVoxel_Particles allows you to choose different particle controllers. A server is add-on software which provides particle information on a per frame basis. Currently, HyperVoxels supports Particle Storm and RealFlow.

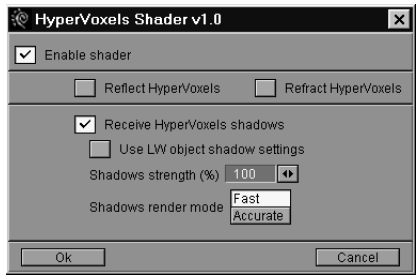
Selecting one of these particle controllers as the **Data Server** will let the HyperVoxels pixel filter plug-in extract selected information, such as **Color**, **Size**, **Speed**, **Alpha Channel**, **Pressure**, and **Force**. These attributes can be used to change parameters in the HyperVoxels surface creation and texturing.

This could allow for HyperVoxels color to support a Particle Storm color path, fade a HyperVoxels surface out based on the Particle Storm death rate, or even change HyperVoxels particle size based on the speed of a Real Flow particle. Many great effects can be created with this tool.

For Particle Storm, selecting **Color** will force particles to get their color fixed by Particle Storm, no matter what the color setting is in the HyperVoxels panel. Selecting **Alpha Channel** will resize the particles depending on the alpha value, as this is usually used to dissolve particles when they “die”. For example, particles with a value of 255 would have a 0 size. Using **Size** will resize the particles using the incoming value from Particle Storm.

For RealFlow, **Size**, **Speed**, **Pressure**, and **Force** are provided to certain gradients as input parameters, then the user may use them in order to change a parameter (color, diffuse, etc.). The items available in the **Input Parameter** pop-up menu will vary. Note that if you use the RealFlow **Size** parameter, the RealFlow particle size value overrides any HyperVoxels **Particle Size** settings. Any effects from **Size Variation** and **Variation Scale** will be applied, however. Also, when using **Pressure**, the values are normalized between 0 (lowest) and 1 (highest).

LW_HyperVoxels_Shader (Surface Shader plug-in)

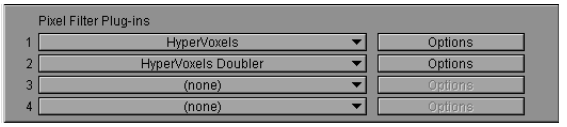


This surface shader allows object surfaces to receive shadows from HyperVoxel, as well as reflect and refract them. If you elect to have the surface receive HyperVoxels shadows, you have the option of using the object’s shadow settings (i.e., **Self-**, **Cast-**, **Receive Shadow** (**Object** panel)).

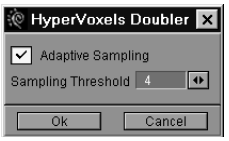
Shadow strength makes shadows more or less opaque. The **Fast Shadows render mode** does not account for hypertextures in the shadows. **Accurate** renders the shadows including the effect of hypertextures.

To use the reflect and refract options, the surface must be reflective or transparent; however, the **Reflection Option** (**Surface** panel) setting is not considered.

LW_HyperVoxels_Doubler (Pixel Filter plug-in)



The **LW_HyperVoxels_Doubler** pixel filter plug-in must be loaded after **LW_HyperVoxels**. (There may be plug-ins loaded in-between, but they should not be active.) It performs the second pass after HyperVoxel's rendering pass when **Render ½ Res** is active. The default values give a good balance between quality and render time.



When **Adaptive Sampling** is inactive, the Doubler performs a straight interpolation between pixels. When active, the Doubler determines if each pixel needs to be recalculated or not, depending on its boundary pixels. This option is very similar to Lightwave's **Adaptive Sampling** option (**Camera** panel).

Sampling Threshold controls the edge detection that decides which pixel should be recalculated. Pixels below the threshold level will be recalculated, those above will be simply interpolated between neighboring pixels. Low values of threshold will make higher quality images with finer details but with higher render times. If Doubler creates aliasing problems on objects boundaries, the threshold should be set to low values (below 4).

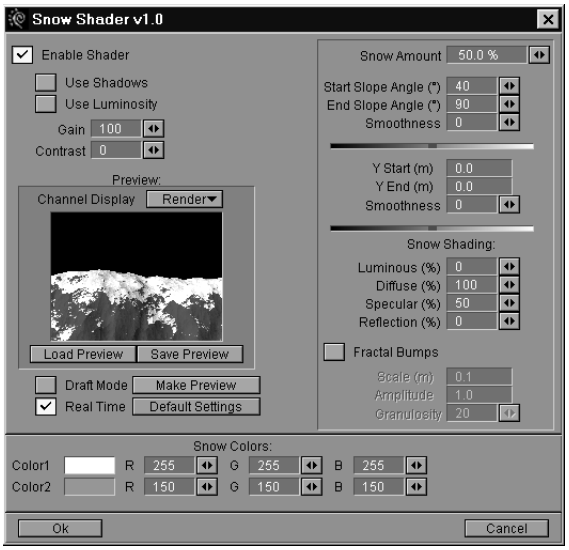
Note If the Doubler is installed and **Render ½ Res** is inactive, the Doubler will be bypassed.

THE NATURAL SHADERS

TheNaturalShaders.p plug-ins are tools designed to modify the shading of surfaces based on natural parameters. You will note that many of the parameters are somewhat arbitrary by nature and thus you will have to use trial-and-error to achieve satisfactory results.

Snow Shader

LW_Snow is a surface shader which controls the surface and shading based on the slope of the geometry. This provides the ability to have a white color on areas creating a snow effect. The shader reacts to geometry as well as bump mapping. The amount of snow can also be controlled by altitude, this way it's possible to have the snow starting at a given altitude and a transition to the maximum amount of snow at another given altitude.



Global Controls

Enable Shader activates the plug-in. This allows you to turn it off without losing any settings.

Use Shadows adds snow only where there are shadows, as if the snow would not melt in those areas because it was protected from the sun. **Use Luminosity** uses the surface luminosity channel to vary the amount of snow, and sets the luminosity value to 0 after the process. Use this to control where snow should and shouldn't appear. A 100% value of luminosity removes any snow, and a 0% value keeps the snow. **Gain** and **Contrast** control the overall luminosity and contrast, respectively, of the shader.

Preview

The Preview area displays the last rendering using the shader. To have a preview the user must do a “test” render (F9) in Layout before opening the shader’s panel. You can also save and load Preview displays using the **Save Preview** and **Load Preview** buttons.

The **Channel Display** pop-up menu currently only provides one option, **Render**. Additional features will be added later.

Activate **Draft Mode** to render a faster low-resolution Preview. Activate **Real Time** to have the Preview redraw anytime a value is modified. When this is off, click the **Make Preview** button to update the display manually.

Click the **Default Settings** button to restore default values to all parameters.

Snow Settings

The **Snow Amount** is the overall strength of the effect, the higher the value, the more visible the snow will be.

Start Slope Angle is the angle at which the snow begins to accumulate. It is measured from the XZ plane towards the surface normal. Thus, a value of 0 means a flat surface with no incline. The **End Slope Angle** is the angle where the snow accumulation reaches its maximum density. Below this angle value, the snow amount is the same. **Smoothness** is the smoothness of the transition between the start and the end slope angle values.

Y Start is the altitude where the snow starts to accumulate. **Y End** is the altitude where the snow accumulation reaches its maximum value. **Smoothness** is the smoothness of the transition between the start and the end values. Higher values create sharper transitions. A value of 0.0 for both **Y Start** and **Y End** means that altitude is not taken into account (the default).

Note The effects of slope and altitude are combined.

Snow Shading

Luminosity, **Diffuse**, **Specular**, and **Reflection** work just like their counterparts on the LightWave **Surfaces** panel.

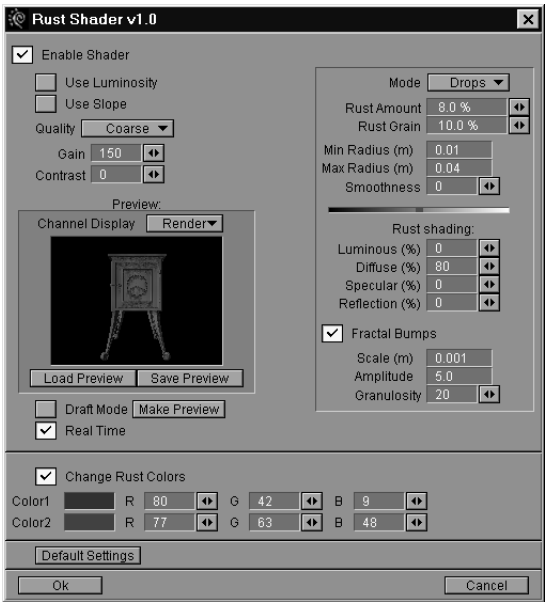
Fractal Bumps applies a texture to the snow. **Scale** is the size of the texture, **Amplitude** is the texture's height and **Granulosity** sets the amount of "grittiness".

Snow Colors

Color1 is the snow color where the effect is at its maximum. **Color2** is the snow color for the surface when it is between snow and original color. This provides a melted snow border between snowy and non-snowy areas.

Rust Shader

Basically, **LW_Rust** evaluates the accessibility of polygons to be shaded. This is accomplished by determining when a defined polygon axis intersects (i.e., hits) another polygon within a defined radius. The direction is set by the mode you use. You can use this plug-in to add rust as well as dust to objects.



Note This shader works very well on objects with relatively straight geometry, like manufactured objects. Smooth surfaces may not fair as well, since intersections will be found more often in places where it's not really needed. Gaps between polygons, inconsistent facing normals, and non-planars may cause unwanted results, due to the miscalculation of intersections.

Global Controls

Enable Shader activates the plug-in. This allows you to turn it off without losing any settings.

Use Luminosity uses the surface luminosity channel to vary the amount of effect, and sets the luminosity value to 0 after the process. Use this to control where the effect should and shouldn't appear. A 100% value of luminosity removes any effect, and a 0% value keeps the effect. **Use Slope** takes into account the effect of a surface's slope to add more rust on flat areas and less on vertical ones.

The **Quality** pop-up menu lets you set the detail level of the effect. Higher settings provide mode detail at the cost of longer rendering times. **Gain** and **Contrast** control the overall luminosity and contrast, respectively, of the shader.

Preview

The Preview area displays the last rendering using the shader. To have a preview the user must do a "test" render (F9) in Layout before opening the shader's panel. You can also save and load Preview displays using the **Save Preview** and **Load Preview** buttons.

The **Channel Display** pop-up menu currently only provides one option, **Render**. Additional features will be added later.

Activate **Draft Mode** to render a faster low-resolution Preview. Activate **Real Time** to have the Preview redraw anytime a value is modified. When this is off, click the **Make Preview** button to update the display manually. Click the **Default Settings** button to restore default values to all parameters.

Rust Settings

The **Access** mode "looks" in the direction of the surface normal. If a surface is within the **Max Radius** value, the effect is applied. The **Min Radius** values sets the distance the effect reaches at its minimum. With low radius values, this mode will only affect small inlay areas, that is, areas that are not easily accessible. The **Grooves** mode looks in the plane that is perpendicular to the surface normal. The **Drops** mode also looks in the plane that is perpendicular to the surface normal, but in the most upwards direction. This mode often results in the most realistic results. **Smoothness** is the smoothness of the transition between the rusted and non-rusted areas. Increasing it will tend to erode the effect from the edges.

Note The **Max Radius** value not only determines where the effect will occur, but it also determines the length of the effect on the surface.

The **Dust Amount** is the overall strength of the effect, the higher the value, the more visible the effect will be. The **Dust Grain** settings will add some grittiness to the effect.

Rust Shading

Luminosity, **Diffuse**, **Specular**, and **Reflection** work just like their counterparts on the LightWave **Surfaces** panel.

Fractal Bumps applies a texture to the dust. **Scale** is the size of the texture, **Amplitude** is the texture's height and **Granulosity** sets the amount of "grittiness".

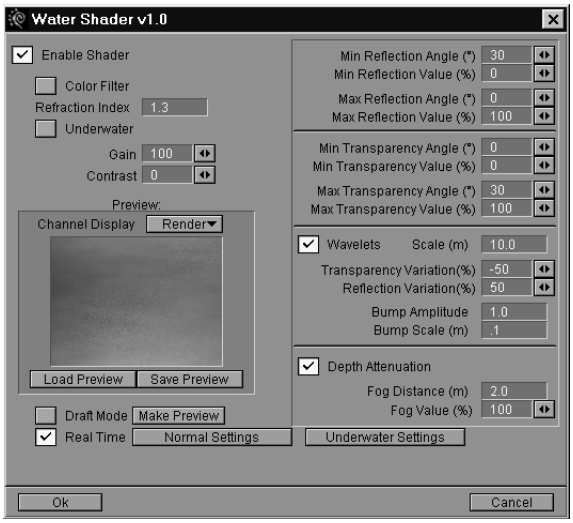
Rust Colors

Color1 is the dust color where the effect is at its maximum. **Color2** is the color for the surface when it is between the dust and original color. This provides a transition border between affected and non-affected areas.

Water Shader

LW_Water lets you quickly add realistic water surfacing.

Global Controls



Enable Shader activates the plug-in. This allows you to turn it off without losing any settings.

Color Filter works like the standard Surface setting with the same name and tints surfaces seen behind transparent surfaces. The **Refraction Index** value can be set using the save values that you would use in the standard **Surfaces** panel. The Underwater option should be used if the camera is immersed in the water. **Gain** and **Contrast** control the overall luminosity and contrast, respectively, of the shader.

Preview

The Preview area displays the last rendering using the shader. To have a preview the user must do a "test" render (F9) in Layout before opening the shader's panel. You can also save and load Preview displays using the **Save Preview** and **Load Preview** buttons.

The **Channel Display** pop-up menu currently only provides one option, **Render**. Additional features will be added later.

Activate **Draft Mode** to render a faster low-resolution Preview. Activate **Real Time** to have the Preview redraw anytime a value is modified. When this is off, click the **Make Preview** button to update the display manually.

Reflection and Transparency

Min Reflection Angle and **Min Reflection Value** are the angle and reflective value, respectively, at which reflection is at its lowest. **Max Reflection Angle** and **Max Reflection Value** are the angle and reflective value, respectively, at which reflection is at its highest. The angle is measured relative to the surface, so looking straight down at the surface would be 90 degrees and looking at it directly from the side would be 0 degrees. The transparency settings work identically, but affect the opacity of the surface.

Rises

The **Rises** option simulates the effect of wind on the water surface. It creates bumpy spots with varying transparency and reflection values. **Scale** is the size of the spots. **Transparency Variation** lowers the transparency in the spots, while **Reflectivity Variation** increases the reflection in them. **Bump Amplitude** is the height of small ripples inside the spots and **Bump Scale** is the size of these ripples. **Bump Scale** should generally be set to low values.

Depth Attenuation

The **Depth Attenuation** option simulates the absorption of light underwater. Basically, it adds a pseudo-fog for any object below the surface, but viewed from above the surface. **Fog Distance** is the depth where the fog reaches its maximum value. **Fog Value** is the maximum amount of fog to be applied. The surface color determines the color of the fog.

The **Normal Settings** and **Underwater Settings** provide default settings for above-water and below-water situations (i.e., looking up to the water surface).

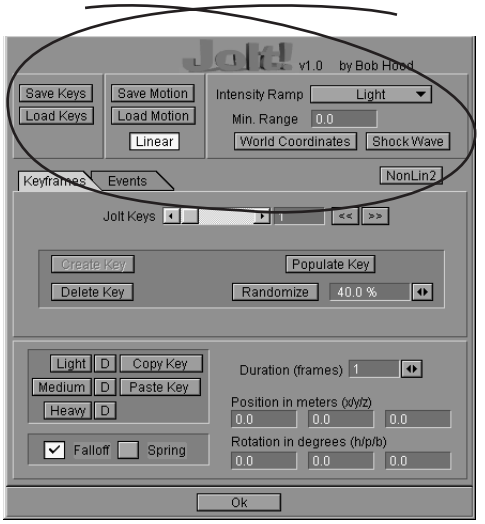
Jolt!

Jolt! is a motion plug-in that simulates the chaotic and uncontrolled movements that would occur when physical impacts occur with enough weight and velocity to cause jarring vibrations. For example, the collision of a celestial body striking another, the impact of a robot's foot with the earth, the collision of a laser beam with a passing spaceship, etc. All of these events would be good candidates for Jolt!

Jolt! also allows you to specify vibration events using keyframes that indicate when a vibration will begin—and be at its most intense—as well as the duration of the vibration and its initial intensity. Intensities can be set on the position of the object (X, Y, and Z axis) as well as its rotational values (heading, pitch, and bank).

Note Jolt! does not provide the basic motion of an item, but rather modifies an existing motion. Jolt! does not modify existing motion except to radically deviate from it temporarily at specified intervals.

Global Options



The **Save Keys** and **Load Keys** buttons allow you to save and restore Jolt! settings. The **Save Motion** and **Load Motion** buttons save and load, respectively, standard LightWave motion files based on the keyframe data. The **Save Motion** button will be ghosted if there are no keyframes set and both buttons will be ghosted when the **Events** tab is selected. Motion files saved with the **Linear** button active will have each of their keyframes set to the (spline control) linear mode.

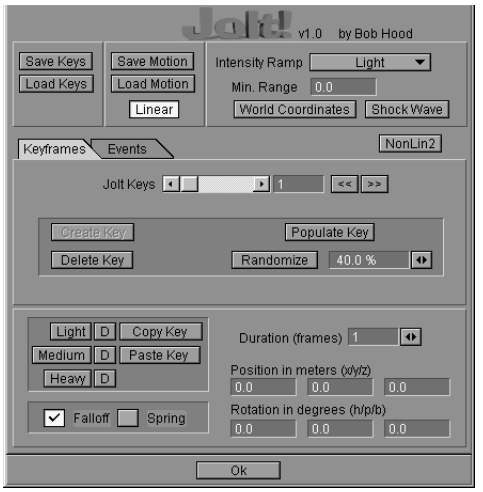
Activating **Intensity Ramp** tells Jolt! to track the movement of a LightWave item (i.e., Camera, light, object, or bone) and based upon its distance from the Jolt! object, increase or decrease the effect of the vibration proportionally. **Minimum Range** is the radius in meters of the effect range. LightWave items outside of this range will not create a visible effect. If you forget to set this value when you leave the Jolt! interface panel, you will be reminded and Jolt! will disable Intensity Ramping (with a minimum range of 0.0, it is ineffective anyway).

If you activate **Shock Wave**, vibration keyframes will be adjusted internally to offset for the distance of the ramp object. In effect, the further away the ramp item is, the longer it will take for the vibration event to actually trigger, and the later it will trigger from its indicated start time. This option would be useful if you were using Jolt! on items in your scene other than the Camera. Items using Jolt! that are closer to the ramp item will begin their vibration events sooner than items that are further away. With items positioned correctly, this option can produce a visible shock wave effect from the ramp item's location.

If you select an item that is part of a parental hierarchy, Jolt! will make you aware of this condition and—if it isn't selected already—suggest that you turn on the **World Coordinates** option to ensure that you get proper movement information from the child item.

The **NonLin2** button activates an alternate calculation for Jolt!'s motion calculations. It creates motions that are less harsh.

Keyframes Tab



The **Keyframes** section houses the Jolt! key controls. When you want the jolting to occur on specific keyframes, you will utilize this tab. (If you want the effect to occur based on the position, rotation, and/or scale of an item, this is set on the **Events** tab. You can set both.) The **Jolt Keys** slider selects the current frame. The range of the slider will exactly coincide with the number of render frames that have been specified on LightWave's **Render** panel—not necessarily the same value that is used in the Preview settings. The << and >> buttons will jump to the previous or next keyframe, if any.

Clicking the **Create Key** button makes a keyframe at the current frame. Use **Delete Key** to remove an existing keyframe. Note that the current frame must be a keyframe to do this.

Clicking the **Populate** key will take the settings for the current keyframe and copy them to every existing keyframe. In other words, it “populates” all keyframes with the current settings and saves you from the tedious task of copying and pasting settings, frame by frame.

Randomizing Keys

Randomizing (or "jittering", if you prefer) provides a means of "breaking up" potentially monotonous key settings. Although Jolt! will internally randomize settings to some degree as it applies them to the item, unless you are using intensity ramping, the actual key values themselves will not be altered. By using the **Randomize** button, you can generate variances such that the motion of subsequent keys do not look so much alike.

The randomizing process requires two or more keys in order to function. The settings of the first key are never altered (nor is there any reason to alter them because no other key will look exactly like them after the effect is applied). The degree of variance can be altered by using the Threshold control input field to the immediate right of the **Randomize** button. This control allows you to specify, as a percentage, the maximum amount that each altered key will deviate from its current value.

Jolting Effect

The bottom half of this tab sets the actual jolting effect. Clicking **Light**, **Medium** or **He**avy will update a keyframe's **Position** and **Rotation** values to reflect preset values for a light, medium, or heavy vibration. This feature can be used as starting points or final settings.

WARNING! Please be aware that when you press any of these buttons, any existing keyframe settings will be destroyed.

Clicking the **Copy Key** button will copy all of the Jolting effect settings to an internal memory buffer. Clicking the **Paste Key** button will paste the settings into the fields currently visible. Note that this can affect either the **Keyframe** or **Events** tab interchangeably.

Using Preset Values

Preset values can be altered by the user. If the **D** button to the immediate right of any of the presets is pressed, then the current settings for the key are stored as the default values for that preset. These new default values will persist between sessions with LightWave 3D. (Jolt! stores its preset defaults in NEWTEK/PROGRAMS/TEMP/JOLT.PRE. You can restore the built-in Jolt! preset values at anytime by deleting this file.)

Applying Turbulence

Jolt! can apply turbulence to your item's motion path in several ways. When you select **Falloff**, the turbulence applied will gradually decrease throughout the duration of the event. In other words, at the first frame of the event, the position and rotation values you have entered for the event will be at their strongest, while at the last frame of the event they will be at their weakest. If **Falloff** is not selected, then the values for the event will be applied at their full strength at each frame throughout the event duration.

If **Spring** is selected, you will get a uniform application of the turbulence, making it appear as though your item were supported and buffered by springs. Without **Spring**, the application of the turbulence is more chaotic or random, producing more of a true vibration effect.

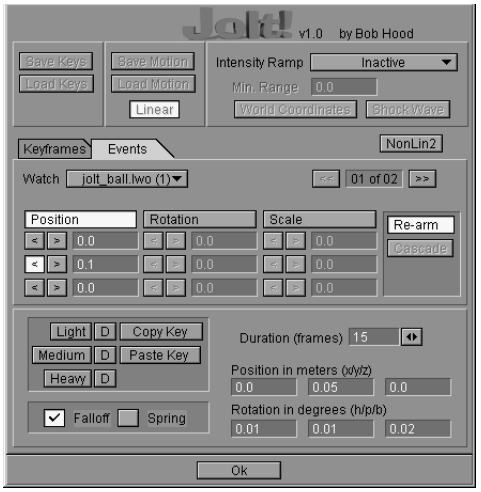
Key Settings

A **Duration** value must be specified, in terms of the number of frames, within which Jolt! must complete the effect. Because the user is allowed to specify duration and location of vibration events, the possibility exists that keyframe data will overlap. Jolt! handles this situation by simply warning the user about the overlap condition. The overlapping key will start before the preceding key ends. If a keyframe is deleted, Jolt! will recalculate all keys to ensure that any overlapped keys are corrected.

The **Position** controls allow you to indicate the maximum deviation on each of the three axes for the item at the current frame. These values are all specified in meters. The higher the number, the more dramatic the initial movement in that direction. A value of zero (0.0) in any position will prevent the item from deviating in that direction. The **Rotation** controls work similarly, but set deviation values for heading, pitch and bank (in degrees).

Events Tab

On the **Events** tab you can cause the jolting to occur based on the position, rotation, or scale of items. (If you want the effect to occur on specific keyframes, this is set on the **Keyframes** tab. You can set both.) The **Watch** pop-up menu will provide you with a list of all the items currently in your LightWave scene. You may select any available item from this list as the “watched” item, that is, the item that triggers the event.



With the watched item selected, click on the **Position**, **Rotation**, and/or **Scale** button to activate the watched attributes. Click either the < (less than) or > (greater than) button next to the input field you wish to set. The **Position** and **Scale** fields correspond to X, Y, and Z, from top to bottom and the **Rotation** fields correspond to H, P, and B.

The << and >> buttons allow you to watch more than one item. To add another watched item, simply click the >> button. The informational display to the left will tell you what item is currently selected and how many there are in total (e.g., 01 of 03). The << and >> buttons are also used to navigate through existing watched items. If you add a watched item by mistake, set **Watch** to **(none)**. You also cannot add another watched item if the last existing item is set to **(none)**.

Activate the **Re-arm** button if the item repeats its motion and you want it to trigger the event again. If **Cascade** is active, Jolt! will ignore the event (i.e., not evaluate it) until the event immediately preceding it has occurred at least once.

The settings on the lower half of the tab work as described for the **Keyframes** tab (see **Jolting Effect**, above).