Lightscape™

Lightscape Visualization System™ Version 3 for Windows NT™ and Windows® 95

User's Guide

Lightscape Visualization System Version 3
for Windows NT and Windows 95, User's Guide
First Edition, November 1996, Part Number: 0-01-030-01-02008

Lightscape Technologies, Inc.

1054 South DeAnza Blvd., Suite 200
San Jose, California 95129
Phone: +1 408 342-1900
Fax: +1 408 342-1910
e-mail: info@lightscape.com
WWW: http://www.lightscape.com

Contents

Appendix A Import Filters Specifications

Preface

The Lightscape Visualization System manuals are comprehensive documents that contain all the information you need to learn and use Lightscape efficiently and effectively.

There are three Lightscape Visualization System manuals:

- The *Lightscape Visualization System Version 3 Getting Started* manual describes the basic concepts of the Lightscape technology and process and the fundamentals of the user interface.
- The *Lightscape Visualization System Version 3 User's Guide* provides in-depth explanations of the techniques and concepts required to set up, process, and render a Lightscape solution.
- The *Lightscape Visualization System Version 3 Tutorials* manual provides step-by-step examples of the procedures discussed in the user's guide.

The following information is available on-line in the Lightscape On-line Help.

- The Task List provides step by step "How to" instructions for performing the procedures discussed in this manual.
- The Reference Guide provides detailed explanations of each menu item and dialog box in the Lightscape interface.

Conventions Used in This Book

Before you start using your Lightscape application, you should be familiar with the typographical conventions used in this manual.

Convention	Type of Information
Boldface	Used for program commands, such as **lid2cibse** or **lid2ies**.
Italic	Used for emphasis and when a new term is introduced.
`Monospace`	Used for what you see on your screen, such as field syntax and macro listings, and unspecified information, such as suggested syntax and macro samples you need to provide. This typeface is shown in Courier. For example, if the instruction says your batch file should read: `lsray -aa 4 -vf view1.vw -x 1280 -y 1024 solution1.ls image1.tif`, you see that text in your batch file.
`Bold monospace`	Used for specific text you enter. This typeface is shown in Courier bold. For example, if the instruction says "type **`cd projects`**," you would type the letters **`cd`** followed by a space and then type **`projects`**. This typeface is also used for keys that you press on the keyboard, either singly (for example, **`Enter`**, **`Return`**, **`Esc`**) or in combination (for example **`Ctrl+C`**).
`Bold Italic monospace`	Used for place holders for items or filenames you must provide. This typeface is shown in Courier bold italic. As in the previous example, if the instruction says "type **`cd`** ***directory_name***," you would replace ***directory_name*** with the name of the directory. For a directory called "projects", you would type **`cd projects`**.
News Gothic typeface	Used for things that you click, such as buttons and icons, or that you choose from menus and their submenus. For example, click Apply; choose New from the File menu.
>	The > symbol indicates that you are to choose an item from a menu or submenu. For example, File > Parameters > Load means that you are to choose Load from the Parameters submenu of the File menu.

Chapter 1

The Lightscape Model

Lightscape supports a robust model structure for preparing geometry and processing radiosity solutions. It is important to understand this structure as you explore the various operations and processes discussed in the following chapters.

As discussed in the *Lightscape Visualization System Getting Started* guide, creating a simulation in Lightscape requires two stages—the Preparation stage and the Solution stage. Because these stages have specific requirements, Lightscape uses a different model structure in each. During the Preparation stage, when you define the geometry and attributes of the model, the structure is similar to that of many CAD and modeling programs. The model structure used during this stage is the Lightscape Preparation Model and is saved in a Lightscape Preparation file (.lp extension). During the Solution stage, Lightscape alters the model structure to optimize it for radiosity processing. The model structure used during this stage is the Lightscape Solution Model and is saved in a Lightscape Solution file (.ls extension). You can no longer manipulate the geometry in the Solution stage. If you need to make changes to the geometry, you can return to the Preparation file (.lp) and then regenerate a Solution file (.ls) when you are satisfied with your changes.

Preparation File Structure

The Lightscape Model consists of a primary entity database and four associated tables:

- Materials Table
- Layers Table
- Blocks Table
- Luminaires Table

When the program starts up, each of these tables is represented by a list window in the default Lightscape screen. The entity database is displayed graphically in the main Graphic Window. The entity database contains all of the 3D geometric entities in the model. These entities may consist of individual surfaces, blocks, or luminaires.

A *surface* in Lightscape is any regular planar triangle or convex quadrilateral. A surface has associated with it materials and other attributes. Surfaces and surface attributes are discussed in Chapter 5, “Surfaces.”

A *block* is a group of entities (surfaces and/or other blocks) that has been given a specific name and an insertion (origin) point. A block can be inserted, or *instanced*, repeatedly in the entity database in various positions and orientations. All instances of a block refer to the same geometric description. If you make a change to the geometry or any attribute of a block, every instance of that block in the model inherits the change. The Blocks Table lists all available blocks in a model. You can store individual blocks in libraries and use them repeatedly in different models. Blocks are discussed in Chapter 6, “Blocks and Luminaires.”

Lightscape creates a special class of blocks, called *luminaires,* to represent light fixtures in the model. You create a luminaire by associating photometric data with an existing block. When you create a luminaire, Lightscape removes the block from the Blocks Table and adds it to the Luminaires Table. As with regular blocks, you can store luminaires in libraries and use them repeatedly in different models. Luminaires are discussed in Chapter 6, “Blocks and Luminaires” and Chapter 7, “Lighting.”

Every surface in the entity database has a *material* associated with it. The Materials Table lists all available materials in a model. A material contains information that defines how a surface reflects, absorbs, and transmits light. These material properties and the character of the light striking the surface give the surface its color. If you change the definition of the material, then every surface in the model with that material assigned to it is changed. As with blocks and luminaires, you can store materials in external libraries and use them repeatedly in different models. Materials are discussed in Chapter 3, “Materials.”

Each entity (surface, block, or luminaire) in the database is associated with a specific layer. *Layers* are used to manage the large number of entities that can exist in a model. You can use layers to break models into logical groupings. For example, you can associate all surfaces that make up a particular room with a particular layer. You can also use layers to store multiple versions of the same model. For example, you can store two alternate furniture layouts for a room on separate layers.

When a layer is turned on, the Graphic Window shows all items associated with that layer. When a layer is turned off, the items associated with that layer do not appear in the Graphic Window. The Layers Table shows all layers and their current state. The current layer states can be saved to a Layer State file which can be loaded at any time to set the state of the layers to the saved values. Layers are discussed in Chapter 4, “Layers.”

Solution File Structure

When the system initiates a Lightscape Preparation Model for radiosity processing, it alters the data structures. It reduces all items in the model to a set of surfaces optimized for radiosity processing. At this stage you can no longer move surfaces within a model. The *initiation process* is discussed in Chapter 8, “Solution.”

The main difference in the model structure between the Preparation stage and the Solution stage is the loss of the block and luminaire structures. In the Solution stage, all block and luminaire instances are exploded—that is, they no longer inherit their geometry and attributes from one single definition, but each instance exists as explicit data. If you change the attributes of a surface on one (former) instance, that change does not occur on the corresponding surfaces in other (former) instances of the same block.

Because the block structure is not supported during the Solution stage, the Blocks Table is not available. The Luminaires Table is preserved, however, because during the Solution stage it is still possible to edit the photometric characteristics of all luminaires of a particular name (even though their geometric properties cannot be altered).

During the Solution stage, the Materials and Layers tables function in the same way as they do during the Preparation stage. If you change the definition of a material in the Materials Table, every surface in the model that has that material assigned to it changes. You can also define new materials or load them from an external library during the Solution stage, so that you can explore various material options using a single Solution file.

Chapter 2

Importing Geometry

The first step in using Lightscape is to import a geometric model. Lightscape can import models from a wide variety of CAD and modeling applications.

Lightscape is not a modeling application. Although it does support some limited modeling operations, generally you create models in specialized CAD or modeling systems and then import them into Lightscape for lighting and rendering. To this end, Lightscape supports specialized translators for converting popular formats to the Lightscape Preparation Model structure.

This chapter describes the common operations and options supported by all translators. Individual translators may offer additional tools or options. For information on specific formats and translators, see Appendix A, “Import Filters Specifications.”

There are four general options you need to consider when importing geometry using any of the translators:

- Units of measurement
- Coordinate system translations
- Block Creation
- Overwrite/append
- Maximum lighting intensity

Units of Measurement

When importing files into Lightscape, you must indicate the units of measurement of the values stored in the incoming file. Because Lightscape is a physically based lighting renderer, physical measurements are an integral part of the solution process. It is therefore essential that you import the originating model into Lightscape with dimensional accuracy to obtain accurate lighting results.

By default, 1 unit in model space is represented as 1 meter in Lightscape. You can also choose millimeters, centimeters, kilometers, inches, feet, or miles. For situations in which the units in the incoming model do not represent whole physical units, you can set a scaling factor. For example, if 1 unit represents 1/10 foot, you select a unit of measurement of feet and a scaling factor of 0.1. Or if the model has a scale of 1 unit to 500 meters, you select meters as the unit and 500 as the scaling factor.

When you import a file, Lightscape asks if the units are correct. If you click No to say the units are not correct, Lightscape aborts the import process and gives you the chance to adjust the import file units. If you click Yes to say the units are correct, Lightscape imports the file and displays the extents of the model in the Graphic Window.

If you incorrectly set the units of measurement when you import your file, you may not obtain the lighting results you expect. For example, if you import a model whose units are inches but you use the default of meters, instead of a room being 120 inches high, Lightscape interprets it as being 120 meters high. Obviously, a lighting fixture at the top of this room will not illuminate it as you expect.

If you aren't sure what units to use when you import the file, it's a good idea to confirm the scale of the model before you start to work on it. One way of testing this is to use the Measure Distance option in the Tools menu to measure the size of a surface whose measurements you know. You can use the Snap to Nearest Vertex option to make sure you are accurately picking the edges of the surface. Pick the points corresponding to the edge of the surface you wish to measure and check the distance between those points. Do these measurements make sense? If not, then you can usually determine which setting you should have selected instead. For example, if you selected inches when you imported and the query tells you that a 10-foot wall is 10 inches, then you know you should have used feet. Import the file again using the correct setting.

Note that after you have correctly imported the model into Lightscape you can change the units you want to work in by choosing Edit > Properties and setting the units in the Document Properties dialog box. This operation has no effect on the physical size of the model. It simply converts the existing dimensions to the new units selected.

Coordinate System Translations

Lightscape uses a right-handed X Y Z Cartesian coordinate system. As shown in Figure 2-1, when viewed from the front, the X coordinate sets the position from right to left (right is positive), the Y coordinate sets the position from front to back (toward the back is positive), and the Z coordinate sets the position up and down (up is positive).

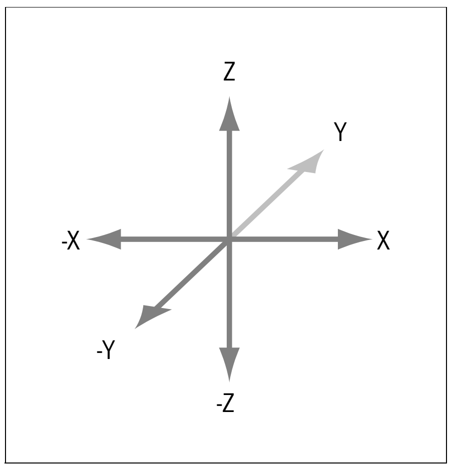


Figure 2-1. *The Lightscape Coordinate System*

Other applications may not follow this coordinate system convention. Many programs use a left-handed coordinate system, as shown in Figure 2-2.

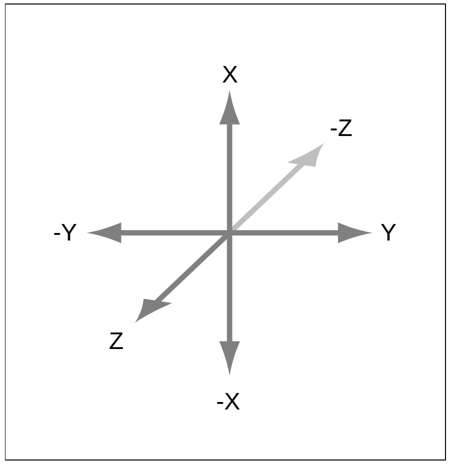


Figure 2-2. *Left-Handed Coordinate System*

The Lightscape translators described in Appendix A, "Import Filters Specifications," use a default conversion that automatically converts the coordinate systems of external file formats to the Lightscape coordinate system. If this is not correct, however, it is also possible to provide an explicit axis conversion string to a translator. With the following procedure, you can determine the string that describes the desired conversion.

To determine the string that describes the desired conversion:

1. Draw the positive axes of the two systems, as shown in Figure 2-3.

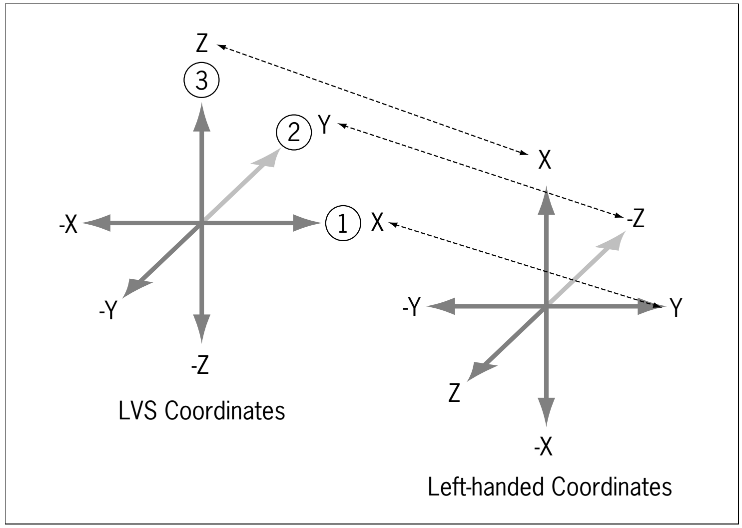


Figure 2-3. *Coordinate System Conversion Example*

   For example, to convert the left-handed coordinate system to the Lightscape coordinate system, draw both sets of axes next to each other on a page, with their positive and negative directions labeled.

2. Draw lines connecting the corresponding axes.

3. Determine the mapping by finding the three Lightscape X, Y, and Z axes in that order, and reading off what they correspond to.

In the example, the axis conversion string is Y-ZX.

Once you have determined the axis conversion string, you can use it to set the appropriate control during the translation process as described in Appendix A.

Block Creation

Although not all systems create objects analogous to blocks, there are some advantages to grouping items together into blocks. In Lightscape, several block creation options are available, depending on what type of file you are importing. The block options organize all entities into blocks based on several options. See Appendix A for a more detailed description of the options available.

Overwrite/Append

When importing geometry, you can choose either to append to or overwrite the current model.

Appending a model to the existing model has the following effects:

- The default properties of the existing file are maintained and overwrite the properties of the incoming file.
- Geometry that exists in specific layers in the incoming model is appended to existing layers of the same name. If new layers are encountered in the incoming model, they are added to the existing Layers Table. This is important to consider if you go back to your original model to export only selected layers on which you made changes. Before importing these altered layers, you should either rename the corresponding layers in Lightscape or delete these layers altogether if their data is no longer relevant.

- In contrast to layers, block definitions in an incoming model overwrite blocks of the same name in the existing model. This has the effect of changing all instances of that block.

If you choose to overwrite, the existing model, entities, and associated tables in memory are completely replaced by the model you import. Before overwriting any existing data, you are asked to confirm this operation and are given the opportunity to save existing data.

Maximum Lighting Intensity

Lightscape provides full functionality for defining and placing physically based luminaires. (See Chapter 6, “Blocks and Luminaires,” and Chapter 7, “Lighting.”) It also supports lighting specifications that exist in supported formats (except DXF).

Because the supported formats do not offer physically based lighting definitions, Lightscape interprets relative intensities by mapping the values encountered to a physically based scale. With this option, you can scale the intensities of lights imported from external models without losing their relative intensities. For example, if the maximum lighting intensity maps to 2500 candelas (cd), then a light that is .5 maximum intensity maps to 1250 cd. The default value is 2500 cd (about the intensity of a 100-watt incandescent fixture).

Note that imported light sources are not generally based on physical principles. Therefore you probably have to tune the lighting in the resulting Lightscape solution to obtain an acceptable result.

Chapter 3

Materials

A material determines the appearance of the surface it is applied to, as well as the amount of light that reflects from the surface during the radiosity process.

A *material* defines the appearance of a surface. It consists of a set of parameters used by the *local reflection model*. A local reflection model is an approximation of the physical interaction of light with the material. (Appendix D, “Reflection Models,” describes the reflection models used in Lightscape.) A material can also use texture maps, both procedural and from images, to modify the appearance of the surface and the effects based on the material parameters. Finally, to make a surface appear to emit light, you can specify that a material has self-emitted luminance.

The Materials Table lists all materials available in a model. You can add individual materials to this table by creating new ones or by loading predefined materials from material libraries. The materials have a significant effect on the solution, so it’s important to have reasonably defined materials on most surfaces before computing a solution.

You can make changes to the materials after computing a solution to see how they look. To update any interreflections caused by the materials, you have to continue the radiosity process. If the materials on many surfaces change drastically, it’s usually better to reset the solution and process the environment again. This topic is discussed in Chapter 8, “Solution.”

Many of the effects of a reflection model are only seen once the system computes a solution and creates an image using the ray tracer. (See Chapter 12, “Rendering.”) Some effects are only seen for those lights for which shadows are refined. Refining shadows is described in Chapter 7, “Lighting,” and in Chapter 12, “Rendering.”

Material Properties

To create a material in Lightscape, you set the material properties that control the appearance of the surface. You can specify these properties directly or you can select a template for various common materials and then make adjustments to the template.

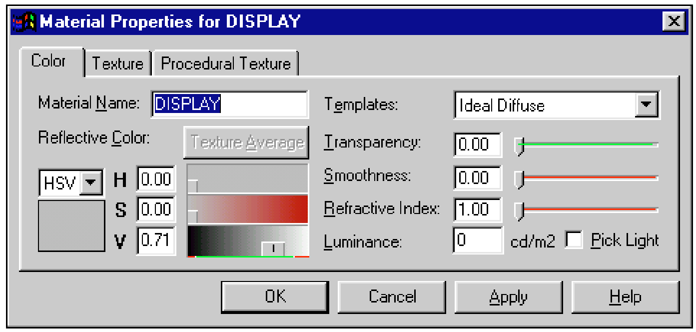


Figure 3-1. *Color page of the Material Properties dialog box*

Lightscape uses the various properties of a material in different parts of the simulation process. For radiosity processing, Lightscape uses these parameters:

- Color
- Transparency

For ray tracing, in addition to color and transparency, Lightscape uses these parameters:

- Refractive Index
- Smoothness

In addition, using texture maps on a surface modifies the color. For more information on texture maps, see “Texture Mapping” on page 3-12.

This section describes how to set these properties for an individual material. The behavior of some materials is the result of extremely complex interactions of light with the surface. In such cases, you may be able to simulate the complex behavior using multiple overlays and texture maps.

Radiosity Processing for Diffuse Lighting

You should set the color and transparency of a material. Lightscape uses these parameters both during radiosity processing and during ray tracing. Lightscape does not use the other material properties during radiosity processing. Radiosity processing accurately calculates the *diffuse* lighting of the model.

It often helps to run quick tests as you define the material. To do so, you should set the material parameters and set crude processing parameters. (For information on setting processing parameters, see Chapter 8, “Solution.”) Run the radiosity simulation until about 75% of the light is accounted for. See if the color bleeding and intensity effects are as you want. If not, adjust the material parameters and try again.

Color

The first property you should set for a material is its color. To correctly set a material’s color, ask yourself these questions:

- How much light is reflected from the material?
- How much does the light change color when it’s reflected?
- What is the material’s color?

You can use the answers to these questions to select the Hue, Saturation, and Value (HSV) components of the material’s color.

As you specify a color, remember that the color you perceive depends not only on the material itself, but also on the light source hitting the material. Consequently, the final color of a material generally appears darker in the color selector than it will in the final solution. You can change the color of a material during the solution process as well as after the solution is complete, so you can make adjustments, if necessary, at any time. For more information, see "Changing Surface Materials and Light Sources" on page 8-33.

If you apply a texture map to your surface, the properties of the texture map override the color of the material. For information on how to turn off this feature, see Chapter 8, "Solution."

How much light is reflected from the material?

Value controls the maximum reflectance of the material; that is, the amount of light diffusely reflected from and absorbed by the surface.

For accurate radiosity processing, choose the reflectance of each material carefully. Increasing Value increases the amount of light reflected and decreases the amount of energy absorbed.

In general, the reflectance of metals tends to be higher than that of nonmetals. The reflectance of metals ranges from about 0.30 (tarnished copper) to 0.9 (highly polished silver), while the reflectance of nonmetals ranges from about 0.05 (coal soot) to 0.7 (white paper). For more information, see Appendix B, "Light and Color."

If you make the reflectance of your material too high, the solution looks washed out and processing time increases significantly.

How much does the light change color when it's reflected?

Saturation controls the degree of coloration of the reflected light. Increasing the Saturation deepens the color of the material and increases how much color is reflected back into the environment.

You should be careful to not make your material overly saturated. As Saturation increases, the light bouncing from the material is highly colored and the color of the entire room takes on the color of that material.

To test the saturation of a material, try this experiment. Hold a piece of white paper perpendicular to the surface. Shine a bright light on the surface so that it reflects onto the paper. Does the paper take on much of the color of the material? If so, the material is highly saturated.

What is the material's color?

Hue controls the color of the material itself.

A blue material reflects more energy in the blue parts of the color spectrum and thus appears blue in color; a red material reflects more energy in the red parts of the color spectrum and appears red in color.

HSV Values

It is usually easier and more meaningful to correctly pick a color using the HSV values, as these correspond readily to the important questions about the color. You can, however, use RGB values. With RGB values, the components correspond to the red, green, and blue parts of the color spectrum.

Each component of the RGB values provides the reflectance for that part of the color spectrum. For this reason, you should keep each of the R, G, and B values in the appropriate reflectance range (0.05 to 0.7 for nonmetals and 0.30 to 0.9 for metals).

Texture Average

If the current material has a texture map, you can use the Texture Average button to set the material color to the average of the pixel values contained in the image. This lets you turn off the Textures option in the Process Parameters dialog box to reduce calculation time and memory requirements in the Solution stage, but still have accurate color reflections on nearby surfaces.

Transparency

Transparency determines how much light passes through the material. The light hitting a material is scattered and attenuated by the material based on its transparency. A material with a small transparency is almost opaque and one with a large transparency is almost completely transparent. All metals are opaque, so their transparency is 0.

A material's transparency and its color are related. Consider a piece of stained glass. The light from a stained glass window depends both on how transparent the glass is and on what color it is. The same is true for apparently clear glass, because glass always has impurities in it. The impurities cause the glass to absorb some light as it passes through the glass.

For example, many types of glass have a transmissivity of 85%, meaning that 85% of the light gets through the glass. In this case, you should set the reflectance (Value) of the glass to 85% (.85) and its transparency to 100%.

Ray Tracing for Highlights and Reflections

When Lightscape ray traces a model, in addition to the color and transparency of a material, it uses other parameters—whether or not the material is metal, its refractive index, and its smoothness. Ray tracing accurately calculates the highlights and specular reflections in the model.

If you do not care about highlights off the surface, you can make your material ideal diffuse and not set these properties.

For these parameters, the questions you should ask yourself are:

- Is the material a metal or nonmetal?
- How shiny is the material? (Refractive Index)
- Are reflections well-defined or blurry? (Smoothness)

If you're going to ray trace your model, you may want to first set up the color and transparency and run the radiosity solution. Once you have the radiosity solution as you want it, you can come back and modify the other material properties to get an accurate ray tracing solution.

It can be tricky to set the refractive index and smoothness correctly. It often helps to use the Ray Trace Area function to do some quick tests of ray tracing. See if you get the results you want, modify the parameters, and try again until you're satisfied.

Refractive Index

The *refractive index* helps determine how shiny a material is. Lightscape uses the refractive index only for nonmetals and only when ray tracing.

A larger refractive index indicates that more light is reflected from the interface between the material and the air—that is, the material is shinier. A refractive index of 1.0 means that all light is transmitted through the material, smoothness does not have any effect, and the surface appears perfectly diffuse.

Most materials have a refractive index between 1 and 1.5, the index for glass. For example, glossy paint has a refractive index of 1.25. By contrast, diamonds have a refractive index of 2.5. You can set the refractive index for some common materials by using the material templates.

To determine the refractive index for a material, shine a bright light directly on the material. If the material is very shiny, then it has a high refractive index. Now, start turning the material to the side. If it only gets shiny when you see it almost edge-on, then it has a low refractive index.

You should model transparent objects that have a refractive index greater than 1 so that they are physically valid. For example, you should model a pane of glass as two surfaces separated by the correct thickness.

Smoothness

Smoothness of the material changes the appearance of reflections and images seen through the material. Lightscape uses the smoothness of a material only when ray tracing.

Note that in Lightscape, smoothness refers to smoothness only as it relates to light. Many surfaces that feel smooth (such as a piece of paper) are not smooth when magnified. Conversely, many surfaces that seem rough (such as an orange) have bumps, which are themselves actually very smooth.

If you ray trace a perfectly smooth material, you get a clear image from a reflection seen through the material. You also get a sharper highlight when shadows are refined. (See Chapter 12, "Rendering.")

As a material becomes rougher, reflections and images seen through the material become less well-defined and blurry.

The best way to determine whether or not a material is smooth is based on how much of a reflection there is and how sharp the highlights on the material are. Sharp highlights means the material is smooth.

If the material has sharp highlights and reflections, but the reflections are distorted and broken up because the material is bumpy, you can use procedural textures to make the material appear bumpy. (See "Procedural Texture Mapping" on page 3-19.)

Smoothness isn't always sufficient to produce reflections and highlights for a surface. It also depends on the reflection model used when ray tracing. (See Appendix D, "Reflection Models.")

A nonmetal with high smoothness but a refractive index of 1.0 does not have any reflections when ray traced using the default high-quality lighting model. On the other hand, the OpenGL-compatible model ignores the refractive index, so reflections will be seen on smooth surfaces when ray tracing with that reflection model.

As with refractive index, setting the correct smoothness for a material can be tricky. You may want to test your settings on a limited set of surfaces.

Templates

Lightscape provides several pre-defined settings for a variety of common materials, including glass, paint, plastic, water, varnished wood, and metal. You can use these pre-defined settings by selecting the appropriate template.

The default template is Ideal Diffuse, whose settings for Transparency, Smoothness, and Refractive Index are all at their lowest possible values. When you choose a template, the values change to the preset defaults. The area of the slider highlighted in green indicates the appropriate value range for that material template. The area of the slider highlighted in red indicates “out-of-range” settings.

Templates for metals are also provided. By default, all materials in Lightscape are nonmetals. In deciding whether or not your material is a metal, remember that what matters to Lightscape (for materials that are not transparent) is the surface of the material. For example, if your material is a painted metal, you should describe it as nonmetal. On the other hand, if you have wood painted with metallic paint, such as gold, you should describe that as metal.

For nonmetals, Lightscape uses both the Transparency and Refractive Index fields. For metals, it deactivates these fields. Metals are not transparent, so the Transparency field is not needed. The system also automatically approximates a metal’s refractive index based on its color.

When ray traced, a metal has colored highlights and reflections and a nonmetal has white reflections.

Luminance

Surfaces do not emit light in Lightscape. All light in a simulation must come from luminaires or daylight. Certain components of real luminaires frequently appear very bright, such as the tubes of a fluorescent light.

To make the 3D models of these components appear bright in Lightscape, you give their materials a luminance value. You can explicitly enter the luminance intensity or you can pick a luminaire in the environment and have Lightscape automatically compute the intensity. The color of the material modifies the color of the luminance.

The luminance value has no effect on the actual lighting of the model; it is only a rendering technique to make a surface appear bright.

Texture Mapping

You can use *texture mapping* to change the appearance of surfaces. Lightscape allows you to vary the color of a material based on an image or on a procedural texture. You can use texture mapping to make a wall appear to be made of bricks or to place a painting on a wall. Lightscape also allows the surface normal to vary across a surface, giving the surface a bumpy or irregular appearance. (The surface normal is the direction perpendicular to a surface at a point on the surface.)

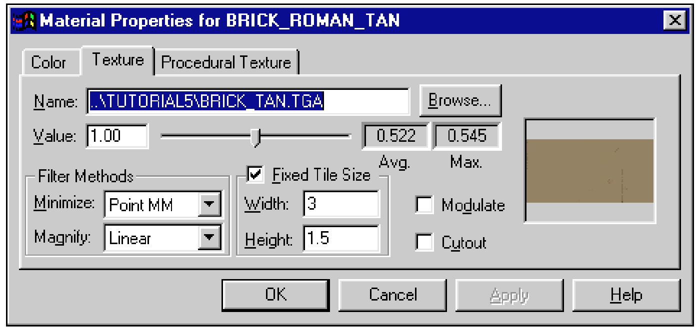


Figure 3-2. *Texture page of the Material Properties dialog box*

If you use a texture map, you can choose whether you want the texture map to be only for display purposes or if you want Lightscape to use it in computing the lighting of the scene during the radiosity process.

If you choose to use the texture map only for display purposes, the lighting of the scene is determined only by the material properties described in "Material Properties" on page 3-3.

On the other hand, if you choose to have Lightscape use the texture map in computing lighting, then the color of the texture map replaces the color of the material for lighting. Each part of the texture should have valid color settings. That is, the Hue, Saturation, and Value components should be appropriate for the material (see "Value" on page 3-16).

Lightscape lets you scale the average and maximum Value (reflectivity) of a texture so you can obtain valid results if the texture represents a physical material. You can also confirm the average HSV values by selecting the Texture Average button on the Color page. If you need greater control of the texture (for example, you want to reduce the Saturation), you may need to use an image editing program.

You decide whether or not to use texture maps in computing lighting as one of the processing parameters in the Solution stage. (For more information, see "Controlling the Simulation" on page 8-10.)

Image Texture Maps

Lightscape allows you to use images to define the color of a material across a surface. As described above, *texture maps* can affect the color of a surface during the radiosity processing as well as during the display of the results. Controlling this effect is described in Chapter 8, "Solution."

File Formats

Lightscape supports image files in the following formats for creating texture maps:

- .bmp (Windows native file format)
- .tga (Targa, TrueVision format)
- .tif (TIFF)
- .rgb (RGB, native Silicon Graphics file format)
- .jpg (JPEG)
- .gif (CompuServe Graphics Interchange format)
- .png (Portable Net Graphics)
- .eps (Encapsulated PostScript)

These images can be of any size. However, larger images require more memory during the rendering process and do not necessarily give any additional quality to the final image if the textured surfaces are small.

When you load a texture image, a thumbnail version appears in the box on the right side of the Texture page.

Modulate

An image used as a texture map can affect a surface color in two ways. It can either replace the existing color completely or scale the existing color by the color of the texture.

When the Modulate option is not selected (the default setting), the image completely replaces the material's existing color. This option is often more useful, because the surface reflectance becomes that of the texture.

When the Modulate option is selected, the existing color is scaled by the color of the image. This is especially useful with black and white textures. In this case, a modulating texture simply varies the intensity of the surface. For example, a blue surface could be modulated to display a range of blues from light to dark. The effect is equivalent to tinting a black and white photograph.

Cutout

The Cutout option allows you to use image maps to cut away portions of a surface. If an .rgb or .tga format image containing alpha-channel information is used as a texture map, Lightscape uses the alpha channel to show through the existing color, or to render underlying surface areas as partly or fully transparent.

When the Cutout option is not selected (the default setting), pixels in the image map containing alpha-channel values other than 255 (white) allow the existing color of the surface to show through. This allows an image to be used as a decal on a surface. The amount of the color allowed to show through is determined by the value of the alpha channel. If the value is 0 (black), the background color is unobscured. With values between 1 and 254, the lower the value, the more the background color shows through.

When the Cutout option is selected, pixels in the image map containing alpha-channel values other than 255 cause the underlying surface areas to be fully transparent (alpha channel of 0) or partially transparent (alpha channel between 1 and 254). For example, if you use the image of a tree in which all background pixels have an alpha-channel value of 0 as a texture map on a flat surface, and then select the Cutout option, the surface will appear to be a tree when viewed from the front. Objects behind the tree will be visible.

If an image map does not contain alpha channel information, the Cutout option has not effect.

Value

Lightscape lets you scale the average and maximum Value (reflectivity) of a texture so you can obtain valid results if the texture represents a physical material. You can also confirm the average HSV values by selecting the Texture Average button on the Color page. If you need greater control of the texture (for example, you want to reduce the Saturation), you may need to use an image editing program. If you are editing multiple materials simultaneously and have more than one image selected, if the values are different, they are not displayed.

There are restrictions imposed by rendering with OpenGL and there are nonlinearities in scaling physical quantities to the dynamic range of a display device. Consequently, the brightness of a texture may be different when using OpenGL or the OpenGL-compatible reflection model in the ray tracer as opposed to the high-quality reflection model. This is described more fully in Appendix B, “Light and Color,” and Appendix D, “Reflection Models.” You can use the Value scaling to adjust the brightness of a texture if it appears too bright or too dark when rendered.

Filter Methods

Lightscape uses two different types of filtering with textures to compensate for discrepancies between the actual image size and the image size as rendered in the scene. The system uses the *Minimize filter* when several pixels in the texture cover the same pixel in the image. The system uses the *Magnify filter* when a pixel in the texture covers more than a pixel on the screen. You can modify how Lightscape does the minimization or magnification.

You have these choices for the Minimize filter:

- Point—Point sample the texture
- Linear—Bilinearly interpolate the value based on the four closest texture pixels
- Point MM—Point sample the closest level in the MIP map for the texture
- Linear MM—Linearly interpolate between point samples from the two closest levels in the MIP map
- Bilinear MM—Bilinearly interpolate between the four closest pixels at the closest level in the MIP map
- Trilinear MM—Trilinearly interpolate between the four closest pixels on each of the two closest MIP map levels

You have these choices for the Magnify filter:

- Point—Point sample the texture
- Linear—Bilinearly interpolate the value based on the four closest texture pixels

The main effect of these filtering options is to blur the texture. The options are in order of increasing blurriness. Blurring a texture is important when the texture contains a lot of small, sharp features. A small amount of blurring may be enough for a static image, but animations usually require more.

Fixed Tile Size

A digital image does not have an inherent physical size; it's simply a rectangular grid of colored pixels. The Fixed Tile Size option defines a physical size for the texture. For example, a painting or a ceiling tile may have known dimensions. When the texture is mapped onto the surface, these dimensions can be used to prevent stretching of the texture.

To specify a size for your image, select the Fixed Tile Size option and then enter the desired dimensions. Once the size is set, you can use the texture alignment tools to position, rotate, and tile the image on a surface. (See Chapter 5, "Surfaces.") If you do not set a Fixed Tile Size for your image, the image can also be stretched to any size using the texture alignment tools.

Aligning a Texture on a Surface

Textures can be placed on the surface using a variety of projections and tiling options, as described in Chapter 5, "Surfaces."

Procedural Texture Mapping

Procedural texture mapping is very different from texturing using images. The procedural textures used by Lightscape use a solid noise field, defined for any point in 3D space. This makes the texture a solid texture that does not need to be aligned on surfaces. (In contrast, you do need to align image-based texture maps.) The surfaces appear to be cut out of the texture.

You can use procedural textures to increase realism by adding variation to the appearance of computer-generated surfaces. Procedural textures generally require experimentation to achieve the desired results.

You can use procedural textures to modify the intensity of a surface by scaling the color at each point or to make a surface appear bumpy by perturbing the surface normal at each point.

The intensity mapping of procedural textures has an average scaling of 1.0 and a bumpy surface tends to be more diffuse than a smooth one. For this reason, procedural textures have no effect on the radiosity processing.

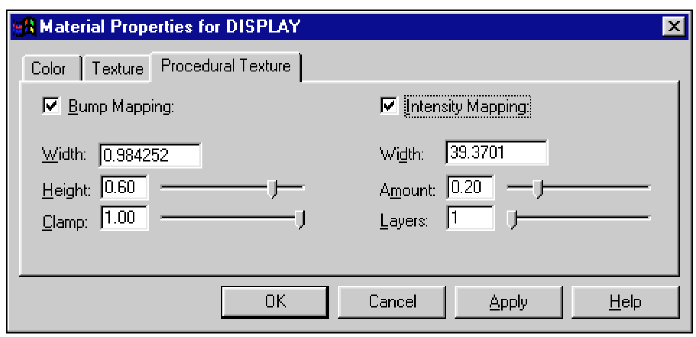


Figure 3-3. *Procedural Texture page, Material Properties dialog*

There are two methods for generating procedural texture maps:

- Bump Mapping
- Intensity Mapping

Bump Mapping

Bump mapping perturbs the surface normal to create the appearance of bumps on the surface. You have control over the frequency and amplitude of the bumps. Setting the clamping value to anything below 1.0 starts leveling the tops of the bumps. A value of 0.3 gives the appearance of a smooth surface with small gouges taken out of it. To get a smooth surface with occasional small bumps, set the height to a negative number. This reverses the orientation of the bumps; that is, they appear as bumps coming out of the surface, rather than gouges into the surface.

Intensity Mapping

Intensity mapping varies the reflectance of the material based on the parameters you specify. This results in smooth variations in intensity over the surface. These variations can make a surface look dirty or slightly wavy. The width and the amount control the frequency and amplitude of the variations. Multiple layers of variation can be added together. Each layer has a different frequency and when several layers are added together the intensity has fractal-like properties, sometimes called *turbulence*.

Materials Operations

The Materials Table lists all available materials for a model. You can add materials to this table either by creating new ones or by loading predefined materials stored in a Material Library.

You initiate operations on the materials from the Materials context menu. You first select the names of the materials you want to operate on and then you choose the desired operation. If you are in Query mode, you can also select materials from the list by picking entities in the Graphic Window. The Materials Table highlights the material currently associated with the selected entity.

You can perform the following operations on selected materials or, where indicated, on the Materials Table as a whole:

- Edit Properties
- Rename
- Add To Criteria List
- Create
- Delete

Edit Properties

Activates the Material Properties dialog box and sets its values to the parameters of the selected material. If more than one material is selected, then only the parameters of equal value among all the selected materials are set. Only the parameters that have new values set are updated in each material. Other parameters are not altered.

"Material Properties" on page 3-3 describes the parameters.

Rename

Use this option to enter a new name for a material. Only one material can be selected in the list. You can also rename materials using the Material Properties dialog box.

Add To Criteria List

Adds the selected materials to the Criteria list. The Criteria list is described in the *Lightscape Visualization System Getting Started* manual.

Create

Adds a new material to the model. The material is given a default name that you can change to a more descriptive name. For example, if the new material is the 15th material in the model, the material name is item15.

You can also create materials using the Material Properties dialog box.

Delete

Deletes all selected materials from the Materials Table. Lightscape associates with the default material any entities in the model that were associated with the deleted material.

There is one level of undo available for delete operations. You can undo this operation as long as you have not saved the file or performed a subsequent action that alters the Lightscape database, such as adding a block instance to the model.

Selecting Materials

The following additional selection tools are available from the context menu:

- Select All
- Deselect All
- Select Pattern

Select All

Selects all the materials in the table.

Deselect All

Deselects all the materials in the table.

Select Pattern

Choosing Select Pattern activates a dialog box in which you can select or deselect material names matching a specific pattern. With the pattern matching, you can use an asterisk (*) as a wildcard character to represent portions of a material name. For example, typing **`fabric*`** selects all the materials that begin with the characters `fabric`. Typing **`paint*semigloss`** selects all the materials that begin with the characters `paint` and end with the characters `semigloss`.

You can also use a question mark (?) as a wildcard character to represent a single character in the name of the material. For example, typing **`fabric?`** finds all materials with one character after `fabric`. It selects all materials named `fabricA`, `fabric9`, and so on. It does not select `fabricWhite`.

You can use any number of asterisks and question marks when selecting material names.

Material Libraries

You can select materials from the Materials Table and save them to a Material Library file. You can then retrieve and reuse these materials at any time. You can load any number of materials from a library into the Materials Table. When you load a material from a library, the loaded material overwrites any existing material of the same name.

After you select a Material Library file to load, the Available Materials dialog box opens. If the correct Path has been set, Lightscape displays a thumbnail representation of all materials in that library. You can load all materials in the library by clicking the Select All button, or load individual materials by selecting the specific material name.

You can create as many material libraries as you like and organize them in a way that makes sense for your application. One common technique is to create global libraries that represent various classes of materials, such as marble, glass, carpet, tiles and so on.

For a specific project, it is also good practice to save all the materials in the Materials Table to a local Material Library that you store with the project files. You can access this local Material Library to retrieve materials from projects without having to open Solution or Preparation files to do so.

Chapter 4

Layers

The Lightscape model structure supports the grouping of entities onto distinct layers. Using layers effectively facilitates the preparation and display of data.

Layer Overview

Many CAD and modeling applications support the concept of *layers*. You can use layers to manage the large amounts of data that may make up your model. A model can contain any number of layers and all data created is placed on a specific layer. Usually the layers relate to logical groupings of entities. For example, all the surfaces that define the walls of a model may be placed on a layer named “walls.” The Layers Table lists all layers that exist in a Lightscape model.

In Lightscape, a layer can be either on (active) or off (inactive). This state is set and displayed in the Layers Table with a check mark to the left of active layers. The Graphic Window displays only the entities on layers that are on. In addition, Lightscape only processes the entities on layers that are on when you initiate the model for the Solution stage. Lightscape deletes layers that are not on at initiation from the Lightscape Solution Model. In the Solution stage, you can still turn layers on and off, for display. In addition, only the entities and lights on layers that are turned on are included in the radiosity process.

Layers serve two general purposes in Lightscape.

- By selectively turning certain layers on or off you can greatly facilitate the process of preparing the surfaces for processing. By reducing the number of displayed surfaces, you increase the speed at which you can move about the model.
- You can use layers as a way of storing various alternatives to a design solution. For example, if you want to test various luminaire layouts in a room, you can set up alternatives on distinct layers. You can then initiate and run various solutions using the alternate layer options.

In addition to turning layers on and off, you can specify one layer in the Layers Table as the *Current* layer. Lightscape adds to this layer any new entities you add to the model. You can also move entities from whatever layer they are on to the Current layer. If you import models from systems that don't support a layering structure, you may want to create new layers in the Lightscape model and selectively move items to these new layers.

In general, it is a good idea to use as many layers as possible in the modeling process in order to simplify the preparation process in Lightscape. In more complex models, it's particularly important to organize your models into logical groupings. If a model is made up of various sections that are independent from a lighting point of view, such as various rooms in a building, it would be much faster to run radiosity solutions on each specific section and append solutions together, rather than trying to create one huge all-encompassing radiosity solution for the entire model. To accomplish such an organization, you either have to create separate models for each independent section or you have to organize each section onto its own set of layers.

Layer Operations

You initiate operations on the layers of a model from the Layers Table context menu. Typically, you first select from the Layers Table the names of the layers you want to operate on and then you choose the desired operation. You can also select layers from the list by picking entities in the Graphic Window while in Query mode. The layer of the picked entity will be selected in the Layers Table.

You can perform the following operations on selected layers or, where indicated, on the Layers Table as a whole:

- Make Current
- Toggle
- On
- Off
- All On
- All Off
- Rename
- Create
- Delete
- Select All
- Deselect All
- Select Pattern

Make Current

Makes the selected layer the Current layer. The Current layer is indicated in the list with a "C" before its name. The Current layer can be on or off for display, just like any other layer. If you add entities to the model and the Current layer is turned off, those items are not displayed, but they are added. Only one layer can be selected for this operation.

Toggle

Switches the selected layers to their opposite state, either on (active) or off (inactive). Active layers are indicated with a check mark next to the layer name. You can toggle the state of one layer at a time by double-clicking its name in the Layers Table. More than one layer can be selected at a time.

On

Turns on all the selected layers regardless of their current state.

Off

Turns off all the selected layers regardless of their current state.

All On

Turns on all layers, regardless of whether or not they are selected.

All Off

Turns off all layers, regardless of whether or not they are selected.

Rename

Renames the selected layer to the specified new name. You can select only one layer for this operation.

Create

Adds a new layer to the model. The layer is given a default name (for example, if the new layer is the 15th layer in the model, the layer name is "item15"), which you can change to a more descriptive name.

Delete

Deletes the selected layers. Lightscape deletes all entities on the selected layers from the model.

There is one level of undo available for delete operations. You can undo this operation as long as you have not saved the file or performed a subsequent action that alters the Lightscape database, such as adding a block instance to the model.

Select All/Deselect All

Selects or deselects all layers in the model.

Select Pattern

Selects the layers whose names match a certain pattern. With pattern matching you can use an asterisk (**`*`**) wildcard character to represent portions of a layer name. For example, typing **`room1*`** selects all the layers that begin with the characters `room1`. Typing **`room*furniture`** selects all the layers that begin with the characters `room` and end with the characters `furniture`. If you are careful about your layer naming convention, this can be a convenient technique for managing layer operations and states.

You can also use a question mark (**`?`**) wildcard character to represent a single character in the name of the layer. For example, typing **`room?`** finds all layers with one character after room. It selects all layers named `roomA`, `room9`, and so on. It does not select `roomWhite` because more than one character follows the word room.

You can use any number of asterisks and questions marks when selecting layer names.

Layer State Files

The current layer states can be saved to a Layer State file (.lay extension). This file records all the layer names and their current state—whether on or off—and also which layer is the Current layer. You can load this file at any time to set the state of the layers to the saved values.

If there is no layer in the model that matches each layer name in the Layer State file, then a new layer is created with the appropriate settings. Layers in the model not present in the Layer State file are left unchanged.

The Layer State file options are located in the context menu of the Layers Table.

Load State

Reads a Layer State file and makes the appropriate adjustments to the Layers Table.

Save State

Saves the current Layers Table settings to a Layer State file. The most recently saved Layer State files are appended to the bottom of the Layers Table context menu so they can be conveniently accessed and set.

Chapter 5

Surfaces

Surfaces are the basic geometric entity of a Lightscape model. This chapter provides information on working with surfaces in Lightscape.

A *surface* is the basic geometric entity of a Lightscape model. A surface can be any convex polygon defined by three or four points located on the same plane. A surface in Lightscape must correctly simulate the actual physical properties of the real surface it represents. Consequently, there are a number of attributes associated with a surface that describe these properties.

If Lightscape cannot determine these attributes from the imported model format, it assigns a default value. You can set the desired attributes explicitly using the tools available in Lightscape. Lightscape also provides a limited suite of tools that you can use to create and edit surfaces in a model. These tools are not designed to be a comprehensive modeler, but rather a means of making minor adjustments to existing models.

You operate on surfaces in Lightscape by first creating a selection set of surfaces in the Graphic Window. Once you create the selection set, you indicate the operation you want to perform by clicking the right mouse button in the Graphic Window to display the context menu. Keep in mind that the full list of surface operation options is available from this context menu when only surfaces are selected—not blocks or luminaires.

Surface Attributes

You can perform the following primary operations on surfaces:

- Assign Material
- Texture Alignment
- Process Control
- Orientation
- Smoothing

Assign Material

Each surface in Lightscape has an assigned material. A material is a set of properties that determines how light energy interacts with that surface. Chapter 3, "Materials," describes materials.

The Materials Table lists all available materials in a Lightscape model. You use the Assign Material option on the Graphic Window context menu to select the particular material you want to assign to the selected surfaces.

Lightscape can automatically make material assignments when importing DXF files by using a technique called a *Material Map*. With a Material Map, you can associate a Lightscape material to a specific color number (that is often used to represent a color in a CAD system—usually in the range of 0 to 255). Appendix A, "Import Filters Specifications," discusses Material Maps.

You generally assign materials during the Preparation stage, although it is possible to set or alter materials during the Solution stage as well. This procedure is discussed in "Changing Surface Materials and Light Sources" on page 8-33.

Texture Alignment

When you use images as textures, you need to align them on the surfaces. You can position, rotate, and stretch (if no tiling size is set) the texture during the alignment process. This alignment defines a way of mapping points on the surface to points in the texture. This section describes how you specify this mapping.

How the texture of the material affects the appearance of the surface is discussed in Chapter 3, "Materials."

You can place textures on objects during both the Preparation stage and the Solution stage. If you position textures during the Preparation stage, you should isolate the block before you align the textures.

If you use textures to modulate or replace the color of a surface during the radiosity process, then the alignment of the texture can be important for calculating the correct interreflections. For this reason, texture alignment should be undertaken prior to starting the processing.

With the Texture Alignment dialog open, you can query the current alignment on surfaces or pick points to use in aligning textures. With the Pick Points check box selected you can also select the Snap to Nearest Vertex check box so that when you select a point in the model Lightscape will select the closest vertex to that point on the same surface.

There are two factors that determine how a texture is mapped onto the surface:

- Projection types
- Mapping modes

Projection Types

There are five different ways of projecting a texture onto a surface:

- Orthographic
- Cylindrical
- Spherical
- Reflection
- Object UV

Orthographic

The orthographic projection is defined by choosing points for the lower left corner, the lower right corner, and the upper left corner.

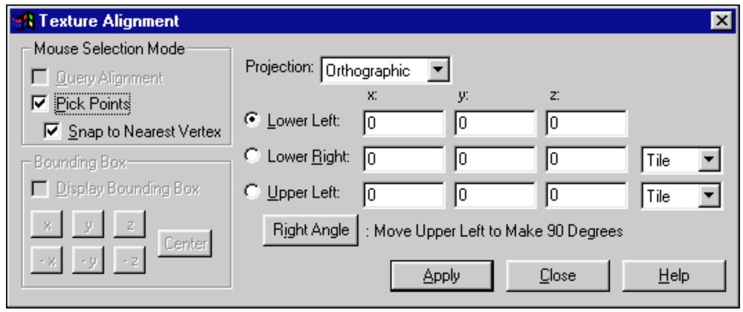


Figure 5-1. *Texture Alignment—Orthographic Projection*

These three points create a plane. Any point on a surface that is above or below the plane is projected onto the plane. Lightscape uses this information to determine the upper right corner (Figure 5-2).

This allows the same set of points to be used for many different surfaces, although the texture may be stretched or distorted, depending on the orientation of the surface. All mapping modes can be used for the orthographic projection.

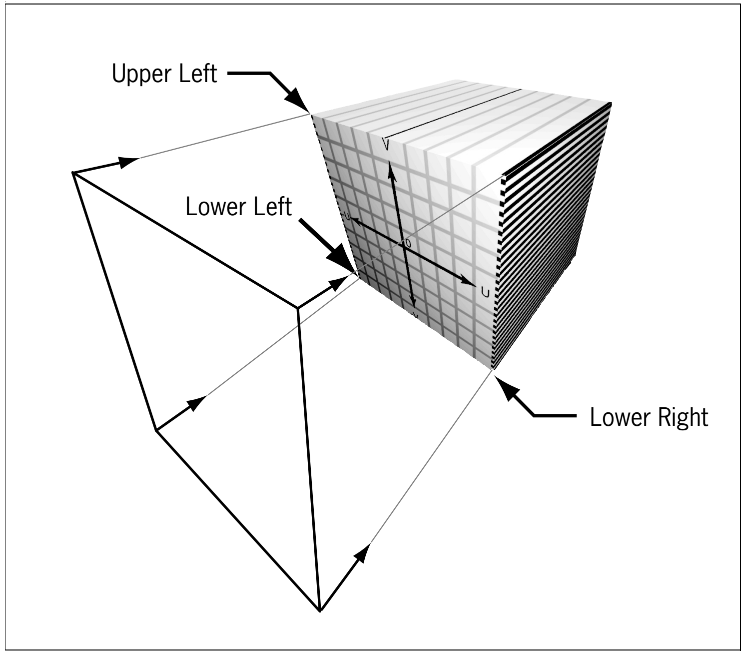


Figure 5-2. *Orthographic Projection of a Texture*

When you pick points to align a texture, you can adjust how the image is placed on the surface by clicking the Right Angle button after you have selected three points to specify the projection plane. Clicking the Right Angle button moves the upper left corner so the points form a 90 degree angle. This is useful when you are aligning a texture and there is no easy way to pick three points at 90 degree angles to each other.

Cylindrical

The cylindrical projection (Figure 5-4) is defined by a lower center point, an upper center point, and a seam direction. The seam direction determines where the right and left sides of the texture meet as they are wrapped around a cylinder.

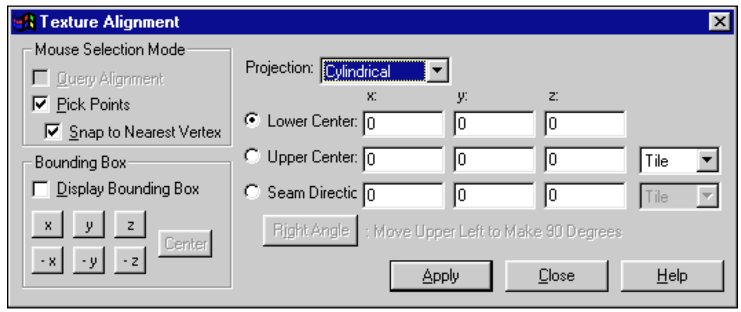


Figure 5-3. *Texture Alignment—Cylindrical Projection*

The mapping wraps the texture around the cylinder exactly once. Points in space are projected toward the center axis until they fall on the cylinder. Mapping modes can be used for the center axis, but are not used for the seam direction. In other words, an image map can be tiled vertically, but not horizontally.

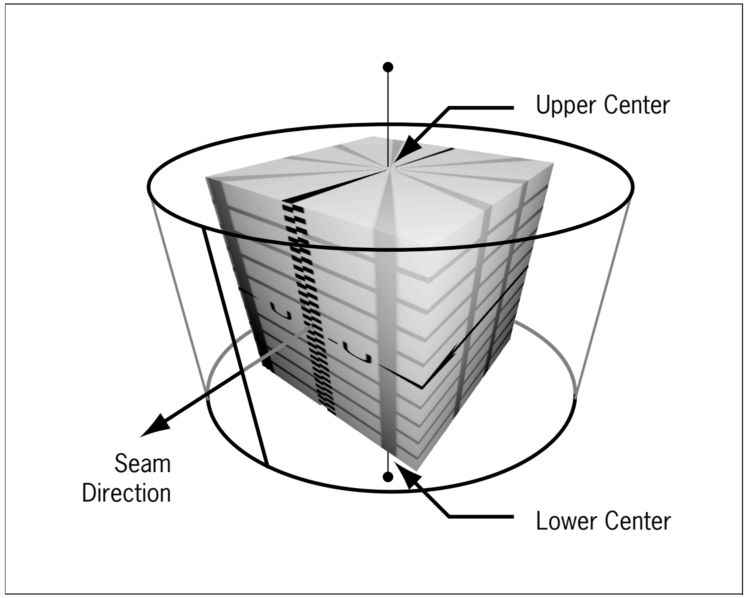


Figure 5-4. *Cylindrical Projection of a Texture*

If you are using the cylindrical projection mode to map an image, you may not be able to pick points to indicate the center axis of the set of surfaces on which you want to align the texture. In such cases, you can use the Bounding Box option to help you select points. With this option selected, you can use the corresponding buttons to pick the X, Y, and Z coordinates for the points to use in mapping the image. You can also display the bounding box around the entire set of selected surfaces to make it easier to see which points you are picking. (You can use the Display > Show Axis option to help determine the axis of the bounding box.)

Spherical

The spherical projection (Figure 5-6) is defined by a center, a top pole, and a seam direction. The seam direction determines where the right and left sides of the texture meet. Points in space are projected onto the defined sphere.

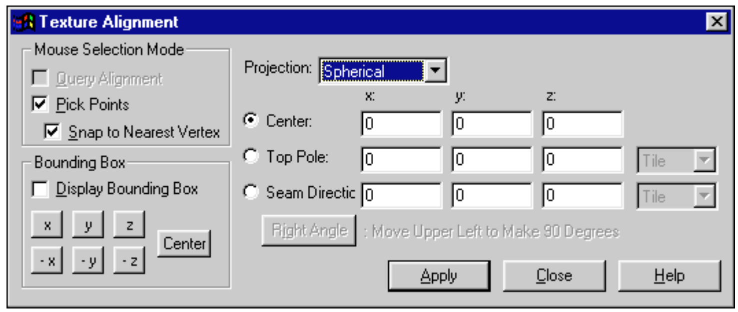


Figure 5-5. *Texture Alignment—Spherical Projection*

No mapping modes can be used for this projection. You should use a Mercator projection (an image based on a spherical coordinate system) to create a texture map for use with a spherical projection.

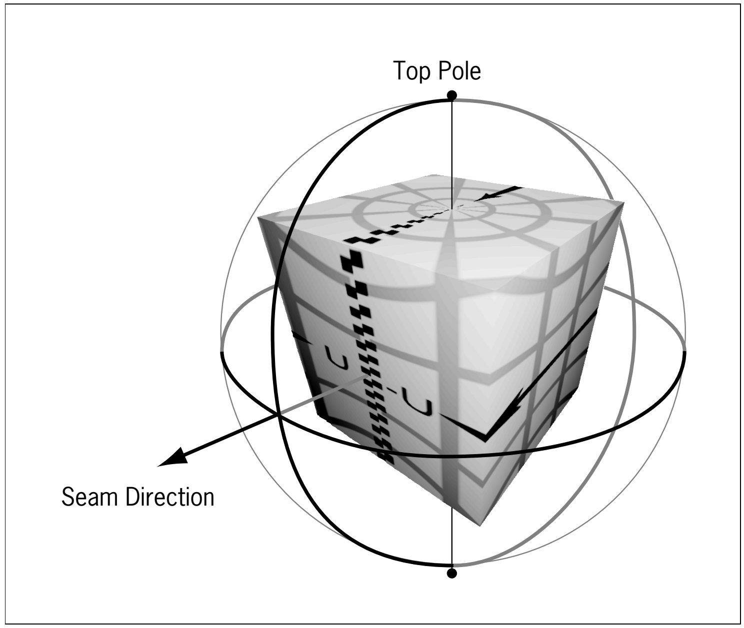


Figure 5-6. *Spherical Projection of a Texture*

As with Cylindrical projection, you can use the Bounding Box option to accurately map a texture onto a set of surfaces.

Reflection

You can use reflection mapping to simulate reflection of objects. It is similar to a spherical projection. A reflection map is defined by an object center (the point from which it is generated), a top pole, and a seam direction to orient the reflections. The reflection map should be created using a Mercator projection.

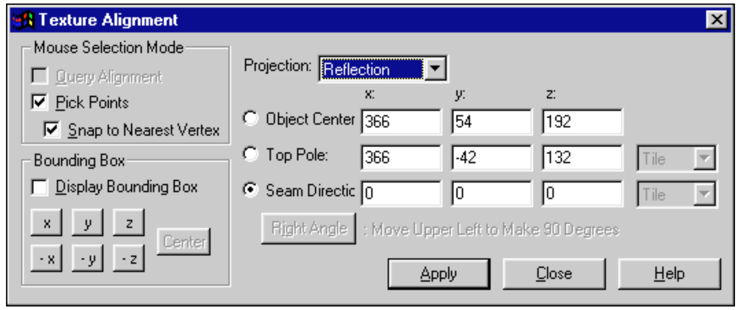


Figure 5-7. *Texture Alignment—Reflection Projection*

Reflection maps add irradiance to the surface based on the position of the camera and the orientation of the surface. This irradiance is modified by the color used for specular reflection for that surface—white for nonmetals, the material color for metals. The smoothness of the surface determines how much of an effect the reflection map has on the surface. Reflection maps do not appear while ray tracing, because the ray tracer computes its own reflections.

As with Cylindrical and Reflection projections, you can use the Bounding Box option to accurately map a texture onto a set of surfaces.

Object UV's

Some modelers can output texture coordinates for each vertex on a polygon.

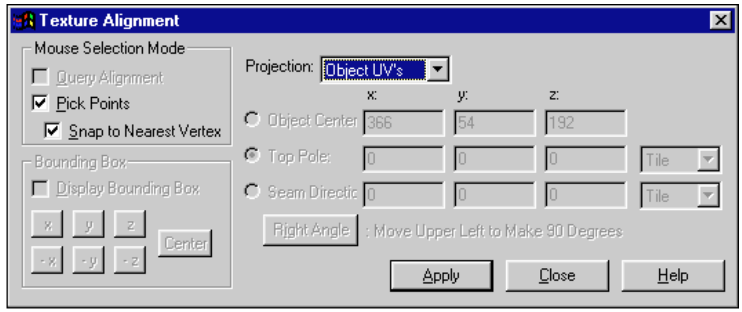


Figure 5-8. *Texture Alignment—Object UV's Projection*

These texture coordinates can be interpolated across the polygon instead of projecting a point to determine the texture coordinates. The UV projection (Figure 5-9) simply notifies the system to use the texture coordinates at the vertices. This projection can only be used if the vertices have texture coordinates set by the original modeling system.

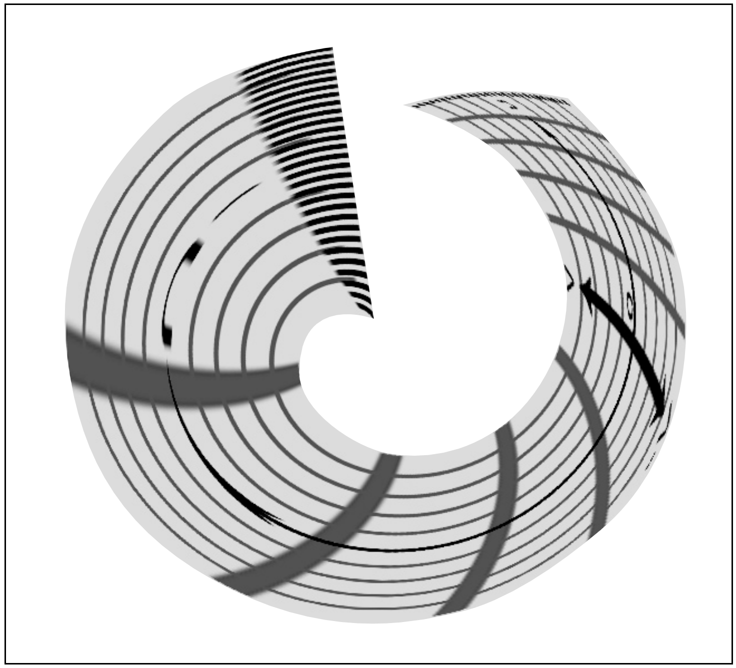


Figure 5-9. *UV Projection of a Texture*

Mapping Modes

When placing textures on a surface, you first determine where to place a single copy of the texture. There are several ways of handling regions of the surface outside this copy. Some of these *mapping modes* are not supported with OpenGL rendering and can only be rendered using ray tracing. (See Chapter 12, "Rendering.")

Mapping modes, as shown in Figure 5-10, include the following:

- Tile
- Flip
- Clip
- Expand

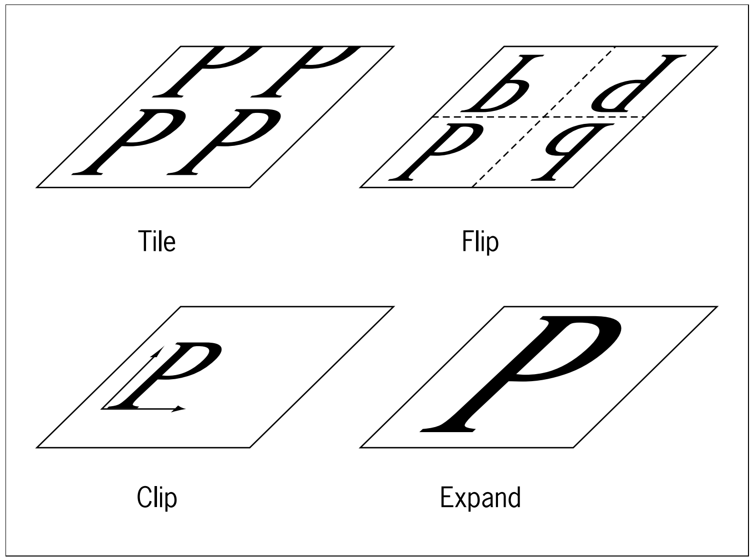


Figure 5-10. *Texture Mapping Modes*

Each mapping mode can be applied separately in the horizontal and vertical directions. For example, it is possible to tile horizontally and clip vertically to create a single row of tiles along the base of a wall.

Tile

Tiling is useful for repeating a texture in exactly the same orientation across a surface. This technique is supported by OpenGL rendering. Pick points on the surface to determine the projection direction of the texture as it is applied to the surface and to set the size of each tile.

Setting a fixed tile size for a texture on the Texture page of the Material Properties dialog box will affect how a texture is tiled onto a surface. (See Chapter 3, “Materials.”)

If a fixed tile size has been specified, Lightscape uses that size to determine the size of each tile on the surface. Lightscape only uses the picked points on a surface to determine the projection direction for the texture only.

If no fixed tile size has been specified, the size of the tile is determined by the picked points.

Flip

Flipping is much like tiling, except that every other copy is reversed. This can make some textures appear seamless as they are repeated across a surface. The OpenGL renderer does not support this option; it can only be seen using the ray tracer. OpenGL tiles across the surface.

Clip

Clipping tells the system to ignore the texture outside the copy you have placed. It is useful for placing decals on surfaces. It can only be seen using the ray tracer and appears tiled when rendered by OpenGL.

Expand

If you do not set a fixed tile size, expanding forces the copy to cover the surface no more than once. The placed copy is expanded and moved but not rotated, so that the edges of the texture just touch the edges of the surface. OpenGL rendering supports this technique.

Process Control

With Lightscape, you can set specific radiosity processing parameters for individual surfaces in the model. This gives you greater control over the quality and complexity of the final results. You can set these parameters during either the Preparation stage or the Solution stage. Because they relate to the general discussion on processing controls, these parameters are discussed in Chapter 8, “Solution.”

Orientation

In the physical world, there is no such thing as an infinitely thin surface. All objects, even a piece of paper, have some thickness. Some modeling systems, often referred to as *solid modelers*, require that all the objects they create maintain this physical property.

Most CAD and modeling systems, however, are not as concerned with accurate physical representations of physical objects and permit the representation of surfaces as planes or curves with no thickness. Although these representations have certain advantages, geometry that is not physically based can create problems when calculating physically based lighting simulations, as is done in Lightscape.

The most significant problem is that of surface orientation. If you use a surface-based model, it is important to specify the proper orientation of the surface. “Orientation” means which side of a surface is the side to be considered when calculating the lighting reflections.

The orientation of a surface is defined by the surface *normal*. In some modeling systems, you can set the normal of a surface during the modeling process and preserve that information when you export the model to Lightscape. (Sometimes the models are imported with all the surfaces consistently reversed, but this can be easily corrected with a single operation.)

Unfortunately, many CAD and modeling systems don’t consider surface orientation at all or don’t preserve the orientation when exporting files. In these cases, if Backface Culling is used in the display, the geometry imported into Lightscape may appear incorrect. Surfaces you might expect to see from a particular view are not visible. These surfaces are usually in the model but are oriented the wrong way (their surface normal points away from the viewer).

When this happens, you need to use the Orientation option in Lightscape to reverse the orientation (normal direction) of these surfaces. The Orientation option offers four methods for changing a surface’s orientation:

- Reversing a surface’s orientation
- Auto orienting backfacing surfaces
- Making a surface double-sided
- Orienting a surface toward or away from a focus point

Note: When orienting surfaces, you should turn off Enhanced display mode.

When the Orientation dialog box is activated, the display of the model in the Graphic Window changes so that backfacing surfaces are no longer culled and are displayed in bright green. This helps you to identify those surfaces whose orientation is incorrect.

Keep in mind, however, that just because a surface is green, doesn't necessarily mean that it needs to be reoriented. It simply means that you are looking at the "back" of the surface, which, depending on your point of view, may or may not be what you would expect to see. From a specific point of view, you only want to select the green surfaces that you would expect to be looking at the "front" of.

It may be more difficult in this special display mode to see all the surfaces in the model. You can use the Near Clip Plane control in the Orientation dialog box to "cut away" sections of the model from the display. This allows you easier access to other surfaces to select them. Other ways of managing the complexity of a model, such as isolating layers (Chapter 4, "Layers") and blocks (Chapter 6, "Blocks and Luminaires"), also facilitate the orientation process.

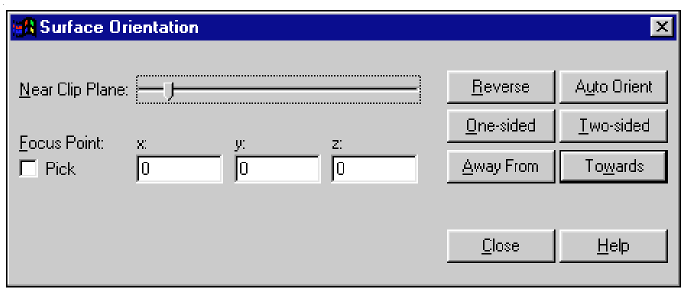


Figure 5-11. *Surface Orientation dialog box*

Reverse Orientation

When you use this option, all of the selected surfaces have their orientation (normal direction) reversed.

Auto Orient

This option automatically reverses all backfacing surfaces (those displayed in bright green) visible in the Graphic Window.

Double-Sided Surfaces

You may occasionally want to use a single plane to represent both sides of a very thin surface such as a plate of glass. In such situations, it may be appropriate to set the surface to double-sided. Keep in mind, however, that for a material such as glass, modeling both sides of the plate of glass with correct thickness between them is important in rendering accurate refraction effects.

It is a good idea to model objects as two surfaces with the correct thickness whenever possible. Because double-sided surfaces are nonreflecting, they may cause display artifacts in the radiosity solution. If you use double-sided surfaces and encounter display artifacts, you will need to return to the Preparation model to correct for these artifacts and then re-initiate the model.

Setting all the surfaces in the environment as double-sided also effectively doubles the number of surfaces in the model. This can increase the time required for processing. It also makes it impossible to use backface culling as an effective display technique.

By default, Lightscape treats all surfaces as single-sided. Surface orientation is an important consideration during the initialization process for the Solution stage. For this reason, you must do all surface orientation operations during the Preparation stage. You cannot alter surface orientation during the Solution stage.

Focus Points

When you use this option, you select a focus point anywhere in the environment. All the selected surfaces can then be oriented so that they either all face away from or toward the focus point.

As illustrated in Figure 5-12, Lightscape determines the orientation of surfaces oblique to the focus point by extending the plane of that surface. Notice how the slight difference in the placement of the focus point between B and C can produce very different results.

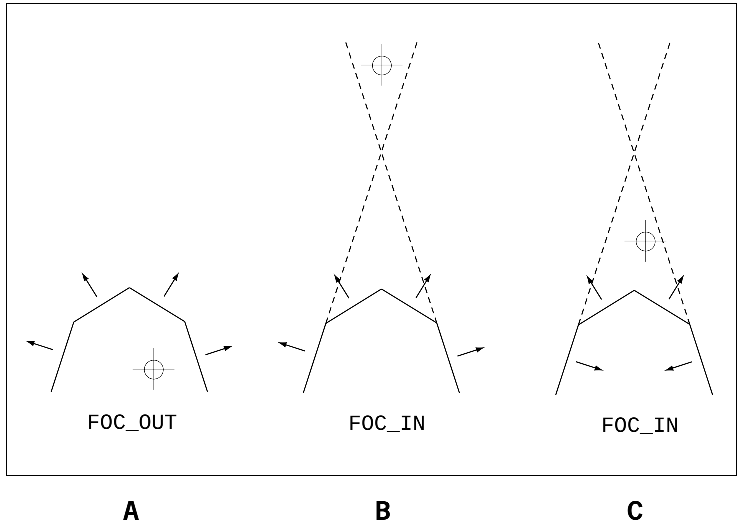


Figure 5-12. *Focus Point Examples*

You can also set focus points in your modeling program if your program supports block output in a DXF file. Using focus points in your modeling program can minimize the amount of reorientation work you must do when you import your model into Lightscape. This technique is discussed in “Importing DXF Files” on page A-2.

Smoothing

Curved surfaces in Lightscape have polygonal representations. In many cases, if the representation of a curve is explicit in the incoming data, Lightscape automatically calculates the vertex normals for the surface and renders these curves smoothly. Otherwise, surfaces in Lightscape are assumed to be independent planes and are rendered as such.

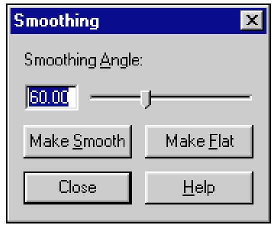


Figure 5-13. *Smoothing dialog box*

Make Smooth

If smoothing information is not explicit in the incoming data, you must explicitly select the group of surfaces that represent a curve and smooth them using the Make Smooth option. As illustrated in Figure 5-14, the Make Smooth option sets the internal angle threshold at which smoothing occurs. If the angle between the surface normals of two faces is less than the smoothing threshold value, smoothing occurs. If the angle is greater than or equal to the threshold value, no smoothing occurs and the boundary between the two polygons appears as a sharp edge.

You can see the results to the smoothing operation more clearly using the Enhanced display mode.

This smoothing operation does not affect the geometry of the model; it only smooths the shading between adjacent edges of the surfaces. The profile or silhouette of the curved surface still shows the faceted edge of the polygons. If this is an issue, then you should take care to create finer polygonal representations of the curve. Certain Lightscape translators provide options for setting the polygonal resolution of the curved surface being imported (see Appendix A, "Import Filters Specifications").

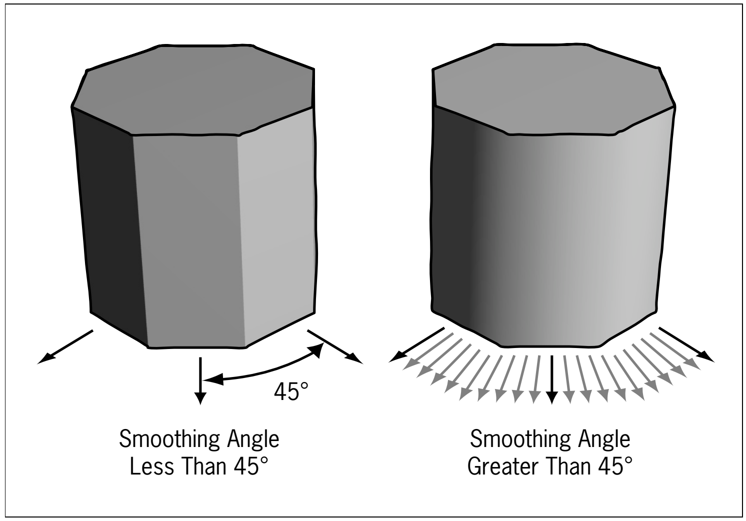


Figure 5-14. *Smoothing*

Make Flat

Use the Make Flat option to remove any smoothing applied to selected surfaces with the Make Smooth option.

Surface Operations

Except for Delete and Change to Current Layer, the following surface operations can only be performed during the Preparation stage:

- Create Block
- Replace with Block/Luminaire
- Change to Current Layer
- Delete
- Transformation
- Duplicate
- Isolate View

Create Block

Groups the selected surfaces into a block with the specified name. Lightscape adds the block to the Blocks Table and inserts a single instance of that block on the Current layer in the current position. (The display remains the same but the surfaces selected are now grouped as the new block.)

This operation does not appear in the context menu if any of the selected surfaces are already part of a block (see Chapter 6).

By default, the insertion point of the block is set to the origin point of the model (0,0,0). This insertion point can be repositioned, as discussed in Chapter 6.

As in AutoCAD, the surfaces in the block retain their layers unless all the surfaces in the selection set are on layer 0. If this occurs, then all surfaces inherit the layer on which the block instance is inserted.

Blocks are discussed in Chapter 6.

Replace with Block/Luminaire

Replaces each of the selected surfaces with the block or luminaire named in the input field. Lightscape inserts the block at the center point of each surface and deletes the surfaces from the model. It orients the block so that the -Z axis of the block is aligned with the normal (perpendicular axis) of the surface being replaced.

This operation, together with the Create Block operation described earlier, is provided primarily as a technique for inserting luminaires into models imported from formats that don't support the block structure. Essentially, it allows you to use surfaces as "placeholders" for where you want to position the luminaires in the model.

To facilitate this substitution process, you should make easily selectable all surfaces you want to represent as a particular type of luminaire. You can do this either by making them all of one unique color and then using the Material Criteria to select them or by placing them all on a unique layer and then turning off all the other layers in the model.

For a full discussion of positioning luminaires and blocks, see Chapter 6.

Change to Current Layer

Changes the layer of the selected surfaces to the Current layer specified in the Layers Table.

Delete

Deletes the selected surfaces from the model.

There is one level of undo available for delete operations. You can undo this operation as long as you have not saved the file or performed a subsequent action that alters the Lightscape database, such as adding a block instance to the model.

Transformation

You can change the relative positions of surfaces either by interactively dragging a surface in the Graphic Window or by entering a specific offset. If you use dragging, you can specify an increment length to constrain the movement of the surface along each axis.

Duplicate

Copies the selected surfaces in a model. All the attribute and layering information of the original surface is preserved.

CAUTION: Duplicated surfaces are coincident with the original surfaces, so you will want to move them immediately.

This operation is useful for setting up workplanes for lighting analysis (see Chapter 10, “Lighting Analysis”).

Isolate View

Isolate View displays only the selected surfaces, turning off the display of any surfaces not in the current selection set. Only the selected surfaces are displayed in the Graphic Window. The current view does not change. To change the focus point to the center of the isolated surfaces, choose View > Extents.

This operation is specific to the current selection set and gives you an easy way to view and operate on a selected group of surfaces without having to turn off layers of the model. If you isolate surfaces that belong to a block, the entire block is isolated.

Isolate View can have multiple levels. For example, you can select several surfaces in a model, then use Isolate View to make only those surfaces (and/or the blocks to which they belong) visible. You can then deselect all but one or two of those surfaces, and use Isolate View again to see only the remaining selected surfaces.

At any level of view isolation, you can use End Isolate View to return to the full view of the model (before you isolated any view). If you've performed more than one level of view isolation, you can use Previous Isolate View to back out one level at a time.

Create Surface

You can add individual surfaces to your model with the Create Surface tool on the Tools menu. With this tool, you can create surfaces by entering four coordinates for quadrilaterals, three coordinates for triangles, or two coordinates to represent opposite corners of a rectangle.

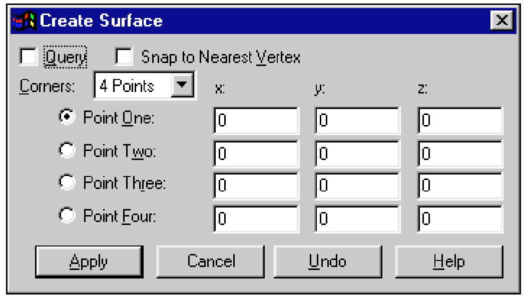


Figure 5-15. *Create Surface dialog box*

If you use two coordinates, then one and only one of the X, Y, and Z components of the two coordinates must be equal. The new rectangle is perpendicular to the axis of the equal component. Surfaces in Lightscape must be planar. If you specify four coordinates and they are not in the same plane, then the surface is broken into two triangles.

New surfaces are added to the Current Layer of the model and they are oriented to face the camera.

Measure Distance

This tool lets you measure the distance between any two points in your model. The coordinates of each point you pick in the Graphic Window appear in the input fields in the dialog box. The Distance field shows the straight-line distance between the two points in the current length units of the model.

The points you pick are indicated with yellow markers in the Graphic Window. The yellow markers disappear when the Graphic Window is redrawn. You can change the size of these markers for better visibility. When you change the size, the markers are redrawn.

Chapter 6

Blocks and Luminaires

The block is an important entity supported by the Lightscape Preparation Model. In Lightscape, luminaires are a special type of block. This chapter discusses the geometric operations you can perform on blocks and luminaires.

A *block* is a collection of entities (surfaces and/or other blocks) grouped together to form a single entity. You give each block a unique name and an insertion point (also called an origin or base point). You define a block once; Lightscape stores it in the Blocks Table associated with a Lightscape Preparation Model.

Once you have defined it, you can insert instances of the block in the Lightscape Model as many times as you want. Each instance can have its own position and orientation in the model, but the actual definition of the geometry and attributes of the block are stored only once. If you change the definition of the block, every instance of the block in the model inherits that change.

The block structure is a very powerful concept supported by many CAD and modeling applications. Although it can be called different names—Cells, Symbols, Repeat Items, Objects—Lightscape has adopted the AutoCAD terminology. The use of blocks can greatly reduce the amount of time required to prepare a model. If the model consists of various repetitive elements and these elements have been modeled as blocks, in Lightscape you need to prepare the surfaces of the block only once. All instances of that block in the model will inherit the results. In addition, you can isolate blocks for display and editing, making their preparation easier and more interactive.

A luminaire in Lightscape is a special type of block having associated photometric attributes. With few exceptions, the operations you can perform on regular blocks and luminaires are the same.

You must define all artificial lighting in your model using luminaires defined from blocks. If your modeling application does not support or export the block structure, you can either create blocks in Lightscape (see Chapter 5, "Surfaces") or you can import the blocks and luminaires from a library, as discussed in “Block Libraries” on page 6-19. (Lightscape provides a library of luminaires for you.) You can then place instances of luminaires in your model using the techniques discussed in this chapter or using the technique discussed in “Replace with Block/Luminaire” on page 5-25.

Blocks and Layers

As in AutoCAD, the surfaces in the block retain their layers unless all the surfaces in the block are on layer 0. If this is the case, then all the surfaces inherit the layer on which the block instance is inserted. (This is also the case when you create blocks in Lightscape, as defined in Chapter 5, “Surfaces.”)

Although surfaces that make up a block may be on different layers, if the layer on which you insert an instance is turned off, you will not see any surfaces regardless of whether the individual surface layers are on or off. If you delete a layer, you must be careful to keep in mind that it may contain surfaces that are part of a block definition.

You may want to insert all blocks on Layer 0 and make sure that Layer 0 is always on. In this way, the layers containing primitive geometry can be easily manipulated.

When Lightscape initiates a Preparation Model for radiosity processing, it explodes all blocks and luminaires. Their component surfaces remain on their original layers. Lightscape places the light source of a luminaire on the layer on which the original luminaire instance was inserted.

Block and Luminaire Operations

Lightscape provides two types of operations you can perform on blocks and luminaires—those that affect the definition of the block or luminaire and those that affect the individual instances of a block or luminaire in the model.

You can perform some operations only on definitions of blocks or luminaires. For these operations, you can select an instance of the block or luminaire in the Graphic Window or you can select its name in the list in the table. Since these operations affect the definitions of blocks or luminaires, all instances of the block or luminaire in the model are affected.

You can perform some operations only on instances of blocks or luminaires. For these operations, you must select the instance (or instances) in the Graphic Window and then select the operation.

For some operations, the effect of the operation depends on where your selection takes place. If you select an instance in the Graphic Window, the operation applies only to that particular instance; if you select a block or luminaire name in the list area of the table, the operation applies to the definition of that block or luminaire (all instances will be affected).

Block and Luminaire Definition Operations

All the operations that affect the definition of the block or luminaire can be accessed through the context menus associated with the Blocks or Luminaires tables.

- Isolate/Return to Full Model
- Query Instances
- Rename
- Change to Current Layer
- Define as Luminaire (for blocks only)
- Luminaire Processing (for luminaires only)
- Photometrics (for luminaires only)
- Create Single Instance
- Delete

Isolate/Return to Full Model

You can isolate the display of a block or luminaire in the Graphic Window so you can more easily operate on it and prepare its surfaces. Certain operations on blocks or luminaires (such as setting insertion points and defining photometrics) can only be undertaken when the block or luminaire is isolated. All instances inherit any changes you make to the block or luminaire while it is isolated.

When you isolate a block or luminaire you are, in essence, temporarily making it the Lightscape Model. All operations you could perform on entities in the full model you can perform on this block or luminaire model. If a block contains other blocks (called *nested blocks*) then those blocks are still listed in the corresponding table. You can now select one of these nested blocks and in turn isolate it.

To return to the full Lightscape Model display, select the Return to Full Model option.

Query Instances

When you select a block in the table and then select Query Instances, Lightscape displays in the status bar the name of the block and the number of instances in the model. In addition, it highlights all instances of that block in the Graphic Window.

If you select a luminaire in the table and then select Query Instances, Lightscape displays in the status bar the source type of the luminaire; its distribution type; the state of the ray trace, shadows, and store direct illumination options; the number of instances in the model; and the name of luminaire. If one of these settings is different for one or more instances of the selected luminaire, Lightscape does not display that information in the status bar.

In addition, Lightscape highlights all instances of the luminaire in the Graphic Window.

Note: If the status bar message is too long to fit in the Graphic Window, the message will be cut off. To see the full message, simply resize the Graphic Window.

If you select multiple block or luminaire definitions, all instances of the blocks or luminaires are displayed in the model, but Lightscape does not display information about those instances in the status bar.

Rename

Lets you rename the block or luminaire. You can select only one block or luminaire definition for this operation. When you select Rename, the name of the selected block or luminaire is highlighted and a blinking text cursor appears at the end of the highlighted text. Type a new name and press **Enter**. This operation has no effect on the instances of the block or luminaire.

Change to Current Layer

Moves all instances of the specified blocks or luminaires to the Current layer specified in the Layers Table.

This affects only the insertion layer of the instances and not the layers of the individual surfaces the blocks or luminaires contain (unless, as described earlier, those surfaces were on layer 0 when the block was created).

Define as Luminaire

This operation applies only to block definitions and is selected from the Blocks Table context menu. You can select only one block for this operation.

Lightscape puts the block into Isolate mode and brings up the Luminaire Definition dialog box. Chapter 7, “Lighting,” describes the complete process of defining a luminaire.

After you define the luminaire, Lightscape removes the block name from the Blocks Table and adds it to the Luminaires Table. If there are instances of this block in the model, all the instances inherit the properties of the defined luminaire.

Luminaire Processing

When this operation is selected from the Luminaires Table context menu, it applies only to the definition of a luminaire. You can select multiple luminaires for this operation.

Lightscape displays the Luminaire Processing dialog box, which lets you set processing parameters for the selected luminaire(s). Chapter 7, “Lighting,” describes the processing parameters for luminaires. When you change these parameters for one or more luminaires in the Luminaires Table, all instances in the model inherit the changes.

Photometrics

This operation is selected from the Luminaires Table context menu and applies only to the definition of a luminaire. You can select only one luminaire at a time for this operation.

Lightscape puts the luminaire into Isolate mode and brings up the Luminaire Definition dialog box. You can then alter the photometric properties of the luminaire, as discussed in Chapter 7. All instances of this luminaire in the model inherit the changes.

Create Single Instance

Adds a single instance of the selected block or luminaire to the model at the model's origin point (0, 0, 0). The new instance is inserted on the Current layer.

Only a single block or luminiare can be selected in the table.

You can also add a single instance of a block or luminaire by clicking and dragging the block or luminaire name from its table to a position in the Graphic Window. The new instance is placed where you release the mouse button.

Delete

Delete removes the block or luminaire definitions from the corresponding table and removes all instances from the model.

There is one level of undo available for delete operations. You can undo this operation as long as you have not saved the file or performed a subsequent action that alters the Lightscape database, such as adding a block instance to the model.

Block and Luminaire Instance Operations

All the operations that affect individual instances of blocks or luminaires can be accessed through the context menu of the Graphic Window. As with other entities, you must first select the blocks and/or luminaires you want to operate on before selecting the desired operation from the context menu.

Keep in mind, if your selection set contains both surfaces and blocks and/or luminaires, the context menu only displays the limited subset of operations that are relevant for both of these entity types. To obtain the full set of operations, only block or luminaire instances can be selected.

- Isolate
- Explode
- New Block
- Multiple Duplicate
- Luminaire Processing
- Change to Current Layer
- Delete
- Transformation
- Duplicate
- Isolate View

Isolate

Isolates the selected block or luminaire instance in the Graphic Window so you can more easily operate on it and prepare its surfaces. Certain operations on blocks or luminaires (such as setting insertion points and defining photometrics) can only be undertaken when the block or luminaire is isolated. All instances inherit any changes you make to the block or luminaire while it is isolated.

This operation is only available if a single block or luminaire instance is selected.

Explode

Explode breaks down the instance of a block or luminaire into its component entities (surfaces and/or other blocks). Surfaces maintain their properties, but no longer inherit any changes made to the blocks or luminaires from which they were originally created. Also, you cannot perform any of the instance operations upon them. Each surface remains on the layer it was on originally (unless it was originally on layer 0, in which case it stays on the layer on which the instance was inserted). Note that when you explode a luminaire, all of its photometric properties are lost.

CAUTION: You may want to save your model before using explode, because undo is not available for this operation.

New Block

Removes a single instance from its definition and uses it to create a new definition in the Blocks or Luminaires tables. Lightscape prompts you to name the new definition. The selected instance is now an instance of the new definition in the table.

This function is useful when you need to differentiate one instance of a block or luminaire from the other instances. By creating a new block or luminaire, you can edit it independently without losing the benefits of the block structure.

You can select only a single block or luminaire instance for this operation.

Multiple Duplicate

Creates multiple copies of the block or luminaire. Lightscape positions the new instances in relation to the existing instance. For each of the X, Y, and Z directions, you specify how many new instances to create offset in that direction and how far away from the original to place the new instances.

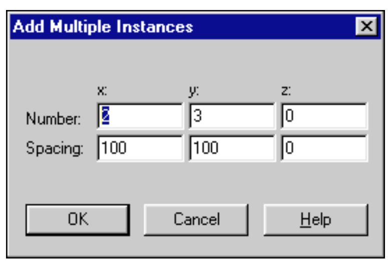


Figure 6-1. *Add Multiple Instances dialog box*

For example, if you specify that you want two copies 100 units away in the X direction and three copies 100 units away in the Y direction, you end up with a total of six blocks or luminaires (5 new ones) positioned in a grid near the original block or luminaire with each block or luminaire offset 100 units from its neighbors.

You can select only one instance for this operation.

Luminaire Processing

When this operation is selected from the Graphic Window context menu, it applies only to the selected luminaire instance. You can select multiple luminaire instances for this operation.

Lightscape displays the Luminaire Processing dialog box, which lets you set processing parameters for the selected luminaire(s). Chapter 7, “Lighting,” describes the processing parameters for luminaires.

Change to Current Layer

This option changes the layer for the selected block or luminaire instances to the Current Layer.

This affects only the insertion layer of the instances and not the layers of the individual surfaces the blocks or luminaires contain (unless, as described earlier, those surfaces were on layer 0 when the block was created).

Delete

Delete removes the selected block or luminaire instances from the model.

There is one level of undo available for delete operations. You can undo this operation as long as you have not saved the file or performed a subsequent action that alters the Lightscape database, such as adding a block instance to the model.

Transformation

You can position, rotate, scale, or move the block or luminaire instances with the transform operation. Lightscape performs the transform operation relative to the insertion point (origin) of the instance.

Position

You can move blocks or luminaires in Absolute, Relative, Drag, or Pick mode.

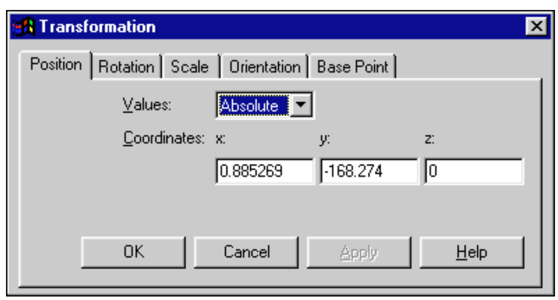


Figure 6-2. *Transformation Position page*

- Absolute— Enter the model coordinates of the new location for the instance.
- Relative—Enter an explicit offset amount.
- Drag—Interactively drag the selected instances in the Graphic Window. You cannot use Drag in perspective view. While dragging, you can constrain the movement of the instances by entering specific length increments for the X, Y, and Z axes. While you are dragging, Lightscape updates only the display of the highlight of the selected instances, to improve interactivity. When you release the mouse button, it updates the display of the entire model.
- Pick—Select a point on a surface in the Graphic Window for the new location of an instance.

Rotation

Rotations can be performed in Absolute, Relative, or Drag mode.

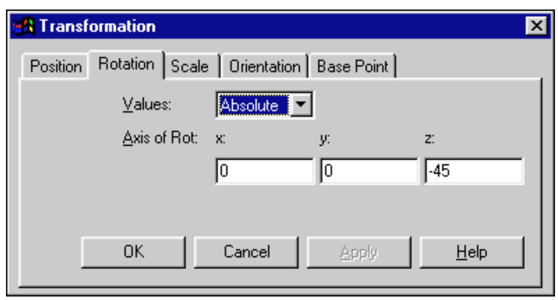


Figure 6-3. *Transformation Rotation page*

- Absolute—You enter an explicit offset angle about the selected axis.
- Relative— Rotation about the selected axis is relative to its original point.
- Drag—You interactively rotate the instance about the selected axis by dragging in the Graphic Window. In Drag mode, you can constrain the rotation of the instance by entering a specific angle increment. While you are dragging, Lightscape updates only the display of the highlight of the selected instances, to improve interactivity. When you release the mouse button, it updates the display of the entire model.

Scale

Scaling can be performed in either Absolute or Relative mode.

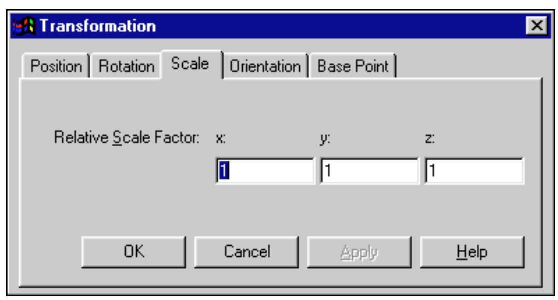


Figure 6-4. *Transformation Scale page*

- Absolute—The instance is scaled the amount indicated relative to its original size.
- Relative—The instance is scaled the amount indicated relative to its current size.

Orientation

Available for luminaire instances only. Orientation simplifies the procedure of orienting certain types of luminaires, such as theater spot lights and track lights.

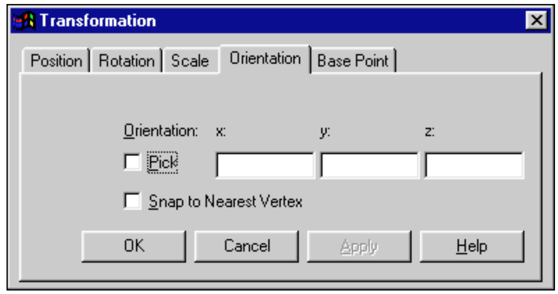


Figure 6-5. *Transformation Orientation page*

When you enter a specific coordinate, either explicitly or by picking a point on any surface in the model, Lightscape rotates all selected luminaires so that their -Z axis aims toward the selected point. For this operation to work predictably, you need to define the luminaire so that its insertion point represents the point of rotation and the photometric aim of the luminaire coincides with the -Z axis as shown in Figure 6-6. For information in defining the photometrics of a luminaire, see Chapter 7, "Lighting."

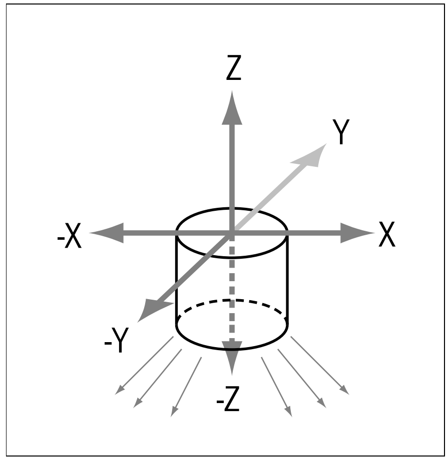


Figure 6-6. *Spotlight*

Base Point

With a block isolated, you can reposition the insertion point in relation to the geometry of the block.

Lightscape provides a number of methods for setting the location of the insertion point. You can either drag the point (represented by green cross-hairs) in the Graphic Window, enter specific coordinates, or pick a point on one of the surfaces. You can also move the insertion point to the geometric center of all the surfaces in the block. This option is available in Absolute mode only.

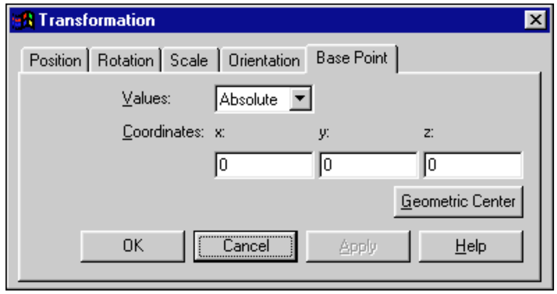


Figure 6-7. *Transformation Base Point page*

Generally you should alter the insertion point of a block only when you first define a block. If you do this operation after you have inserted instances of it in the model, you will be altering the transformations of those instances, with unpredictable results.

Duplicate

When you copy a block or luminaire instance in the model, Lightscape adds a new instance to the model in exactly the same location as the selected instance.

You should immediately move the new instance away from the original instance.

Isolate View

Isolates selected block or luminaire instances. You can select a single block or luminaire instance to isolate or you can select multiple block or luminaire instances.

If you isolate a selection set containing more than one item, you can continue to select objects from your selection set to further isolate. For example, you can select several block instances in a model, and use Isolate Views to make only those blocks visible. You can then deselect one or more blocks and use Isolate View again to see the remaining blocks in your selection set.

Use End Isolate View to return to your previous view of the full model. Also, if you've isolated the view multiple times, use Previous Isolate View to back out of the isolated views one level at a time.

Block Libraries

You can save a block or luminaire definition to an external library to use in other models. You can load block and luminaire definitions from a library into the tables of your model. Block and luminaire libraries provide a convenient means to store specific entities you use frequently.

When Lightscape loads a block or luminaire from a library, it overwrites any existing block or luminaire and material definitions currently defined with the same name. This can be a problem if you're not prepared for it, but it can also be a useful feature.

For example, you might want to replace all occurrences of a specific chair with another chair stored in a block library. If the block in the model does not have the same name as the block in the library, you must first rename the existing chair block to the block name stored in the library. You can then load the new chair block from the block library to replace all instances of the old chair with the new one.

To take advantage of the block-swapping capabilities of Lightscape, you should develop a set of consistent conventions for modeling the various elements you may want to swap. For example, you could decide to always define chairs with their insertion points at the bottom center of the chair and the seats always pointing in the same direction. Then, when you substitute one chair for another, it is positioned predictably.

You can also perform block substitution at the time of importing a file into the system. When importing the file, you specify the libraries you want to load at the same time. During import, Lightscape searches through all the loaded libraries and replaces any block encountered in the file with the block of the same name stored in a library. (See Appendix A, "Import Filters Specifications.")

If your modeling system supports blocks, you can use block substitution to more easily lay out the position of blocks and luminaires in your 3D environment. In this situation, the geometry of the blocks used in your modeling system doesn't matter because it will be replaced with the geometry and attributes of the entities encountered in loaded Lightscape libraries. The blocks in the modeling system can, in fact, be represented with simple 2D graphics. Lightscape considers only its name, insertion point position, rotation, and scaling when substituting blocks or luminaires from Lightscape libraries.

Chapter 7

Lighting

An important feature of Lightscape is its ability to accurately simulate light sources and provide accurate analytic and visual feedback of the lighting in your models. You can work with both natural and artificial lighting.

Artificial Lighting

Lightscape makes creating luminaires (artificial light sources) easy and intuitive. Once you have defined a luminaire, you can store it in a Luminaire Library for ongoing use. Lightscape provides a library of basic luminaires you can use or modify. A powerful feature of Lightscape is the Photometric Web editor. You use it to interactively model any luminous intensity distribution for a luminaire and to import photometric data using a number of industry-standard photometric file formats.

Creating Luminaires

A *luminaire* is the basic lighting primitive in Lightscape. It represents both the physical appearance and the photometric characteristics of a lighting fixture. That is, it includes the geometry of the fixture, the color, the intensity, and the directional distribution of light energy emitted from the fixture.

You create a luminaire by associating photometric data with an existing block definition. Except for defining and editing the photometric characteristics, a luminaire behaves exactly like any other block.

When you create a luminaire, Lightscape removes the block from the Blocks Table and places it in the Luminaires Table. It changes all instances of the block to be luminaires. All of the geometric operations you can perform on luminaires, including saving them to libraries, are discussed in Chapter 6, “Blocks and Luminaires.”

Lightscape supports three basic types of luminaires—point lights, linear lights, and area lights.

Each luminaire has a *luminous intensity distribution* (LID) that describes how the strength of the emitted light varies with the outgoing direction. You set the location and orientation of the LID with respect to the geometry of the luminaire when you define the luminaire.

The geometry of the luminaire may or may not affect its photometrics. Typically, photometric definitions (for example, IES files) provided by manufacturers or ones you create, take into account the effects of the geometry of the luminaire when they are created. In this situation, you would not want the geometry to affect the photometry further and should position the LID to ensure that the surfaces of the luminaire do not shadow the emitted or reflected light when the reflected light is already considered by the LID. (You can also define all the surfaces of the luminaire to be non-occluding and non-reflecting, as discussed in Chapter 8, “Solution.”)

To create a luminaire, select a block in the Blocks Table and click the right mouse button over the list area of the table to display the context menu. Next, select Define as Luminaire. To edit an existing luminaire, select it in the Luminaires Table, click the right mouse button to display the context menu, and select Photometrics.

Both of the above actions display the Luminaire Properties dialog box where you can define how light energy is transmitted from a luminaire.

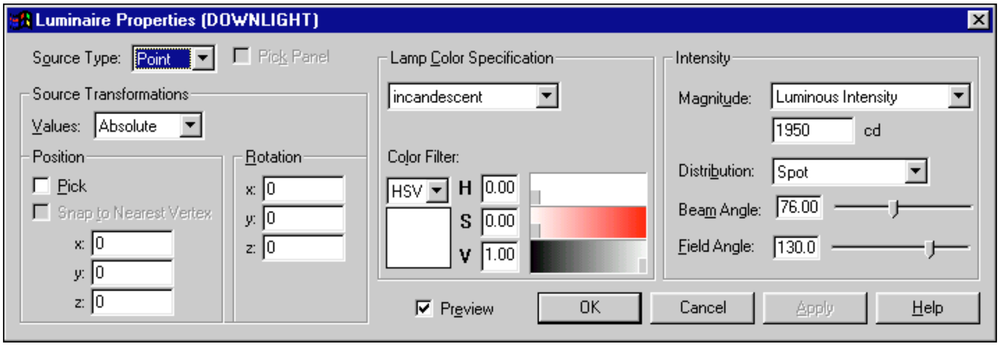


Figure 7-1. *Luminaire Properties dialog box*

Photometric Properties

You can set the following photometric properties of a luminaire:

- Source type (Preparation stage only)
- Lamp color specification
- Color filter
- Intensity magnitude
- Intensity distribution
- Source transformations (Preparation stage only)

Source Type

During the Preparation stage, you use this option to set the most general lighting characteristics of a luminaire. Once you specify the desired type, you must position the LID of the luminaire as described in this section. Lightscape supports three types of light sources:

- A *point light* source distributes energy from a single point. An incandescent bulb is a good example of such a source.

  Lightscape represents the LID of a point light with a spherical 3D icon. By default it positions the LID to coincide with the insertion point of the luminaire block. You can use the Lightscape tools to reposition the LID either explicitly or interactively.

- A *linear light* source distributes energy along a straight line segment. A fluorescent tube is a good example of a linear light source.

  You specify the LID of a linear light by picking a single surface on the luminaire. The LID corresponds to a line down the center of the longest dimension of the selected surface. The LID aims in the direction of the surface normal. You can use the Lightscape tools to reposition the LID either explicitly or interactively. This operation does not affect the surface from which the LID was defined.

- An *area light* source distributes energy from a triangle or convex quadilateral surface. A typical area light is a 2′ x 4′ fluorescent fixture that emits light fairly evenly over the entire surface of a diffuser panel, even if it contains a number of fluorescent bulbs.

  You specify the LID of an area light by picking a single surface on the luminaire. The LID corresponds to the area of the selected surface aimed along the direction of the surface normal. You can use the Lightscape tools to reposition the LID either explicitly or interactively. This does not affect the surface from which the LID was defined.

Lamp Color Specification

You use this option to approximate the spectral character of the lamp. Lightscape provides the following common lamp specifications:

- D65White
- Fluorescent
- Deluxe warm white
- Deluxe cool white
- Warm white
- Cool white
- White fluorescent
- Daylight fluorescent
- Incandescent
- Xenon
- Halogen
- Quartz
- Metal halide
- Mercury
- Phosphor mercury
- High-pressure sodium
- Low-pressure sodium

As discussed in Appendix B, "Light and Color," Lightscape is currently limited to using only RGB values when calculating the radiosity solution. As a result, subtle differences between lamp types may not always be visibly apparent in the final image.

Color Filter

You use this option to set an HSV or RGB color that simulates the effect of a color filter placed over the light source. For example, a red filter over a white light source casts red light.

Intensity Magnitude

You use this option to set the strength or brightness of the light source. Lightscape supplies several ways to set this. The method you choose depends on what form of lighting specification you use to define the light source. Appendix G, "Common Lamp Values," provides a number of common lighting values.

- *Luminous intensity*—The maximum luminous intensity of the luminaire, usually along the direction of aim.

  Measured in candelas (cd).

- *Luminous flux*—The overall output power of the luminaire.

  Measured in lumens (lm).

- *Illuminance at a distance*—The luminous flux density incident on a differential area (a point) a given distance along the aim direction of the luminaire and oriented toward the luminaire. That is, the illuminance caused by the light on a surface at a certain distance and facing in the direction of the source.

  The illuminance is measured in either footcandles (fc) or lux (lx), depending on whether you are working in American or International lighting units. The distance is measured in the current units of the model. Both of these settings can be adjusted in the Document Properties (Edit > Properties) dialog box.

- *Adjust Intensity*—Allows you to adjust the current luminous intensity, based on the maximum luminous intensity.

  Measured in candelas (cd).

Intensity Distribution

You use this option to set the intensity distribution of the light source. Lightscape supplies several ways to set this distribution. The method you choose depends on the type of the light source. Not all methods are valid for each type.

- *Isotropic*—Distributes the light equally in all directions.

  Valid for point lights.

- *Diffuse*—A diffuse distribution is an axially symmetric luminous intensity distribution such that the emitted light varies as the cosine of the emission angle, measured from the axis of the distribution.

  Valid for linear and area lights.

- *Spot*—Defines a spotlight distribution by a beam angle and a field angle. The beam angle is the angle at which the intensity of the light is 50 percent of the maximum intensity at the center of the beam.

  Visually, the beam represents the visible diameter (hot spot) of the spotlight on a surface. The field angle represents the angle where the light is abruptly cut off. A spotlight where the field is much greater than the beam has a soft-edged effect (flood light).

  Valid for point lights.

- *Photometric Web*—Uses a defined photometric web to distribute the light. A photometric web is a 3D representation of the luminous intensity distribution of a light source. See “Photometric Web” on page 7-10 for a detailed discussion of photometric webs.

  Valid for all light source types.

Source Transformations (Position)

During the Preparation stage, you use this option to set the position of the LID within the geometry of the luminaire.

Position

Once you've defined the LID for a point, linear or area light, you can reposition its relationship to the geometry of the luminaire. For point lights, you pick a single point on the geometry. For example, if you have a light bulb in a large square fixture, you might position it either at one of the corners of the fixture or in the center.

You can position luminaires in Absolute of Relative mode.

- *Absolute*—Enter the model coordinates in the X, Y, and Z input boxes.
- *Relative*—Enter an explicit offset amount. Select Drag to drag the position of the LID in the Graphic Window in the specified increments.

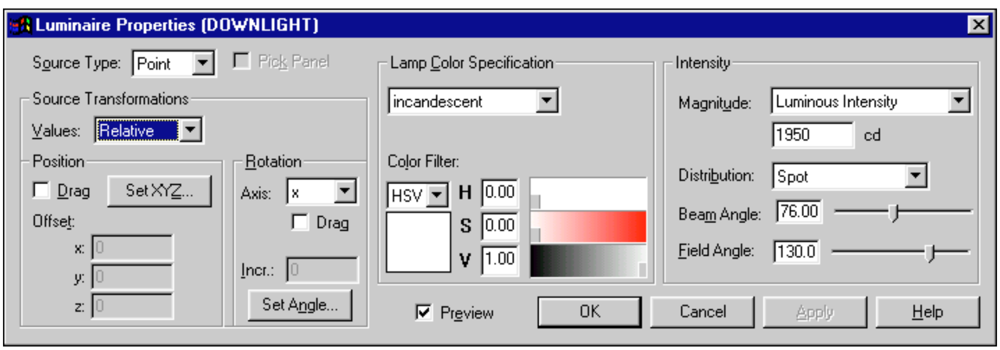


Figure 7-2. *Source Transformations Relative mode*

Rotation

Rotation aims the LID. That is, it determines the direction that the light emitted by the LID travels, relative to the model as a whole. For example, if you position a light bulb on one of the sides of a square fixture, you could point it so that it emits light down out of the fixture, to the other side of the fixture, or in some other direction.

You can rotate LIDs in Absolute of Relative mode.

- *Absolute*—Enter the model coordinates in the X, Y, and Z input boxes.
- *Relative*—Enter an explicit offset amount. Select Drag to drag the rotation of the LID in the Graphic Window in the specified increments.

Photometric Web

Lightscape uses a *photometric web* to represent general luminous intensity distributions. You can use this distribution in the definition of all three types of light sources supported by Lightscape—point, linear, and area lights. You can directly load photometric data files provided by various manufacturers into the photometric definition. Alternatively, you can interactively edit them, or create your own, using the Photometric Web editor.

In order to describe the directional distribution of the light emitted by a source, Lightscape approximates the source by a point light placed at its photometric center. This approximation allows the distribution to be characterized as a function of the outgoing direction only. The luminous intensity of the source for a predetermined set of horizontal and vertical angles is provided and the system can compute the luminous intensity along any arbitrary direction by interpolation.

This kind of representation is widely used in the lighting industry to describe the photometric characteristics of both lamps and luminaires. Lighting companies often make this data available to design professionals for use in lighting analysis programs.

This kind of data is often depicted using *goniometric diagrams*, as in Figure 7-3.

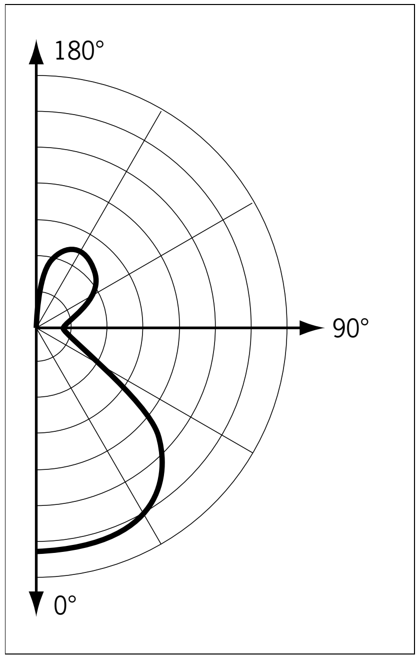


Figure 7-3. *Goniometric Diagram*

These diagrams provide a visual representation of how the luminous intensity of a source varies with the vertical angle. However, the horizontal angle is fixed and, unless the distribution is axially symmetric, more goniometric diagrams may be needed to describe the complete distribution.

Lightscape extends the goniometric diagram to 3D, so that the dependencies of the luminous intensity on both the vertical and horizontal angles can be examined at the same time. The center of the representation is the photometric center of the surface wrapped around it. The luminous intensity in any given direction is proportional to the distance between this surface and the photometric center, measured along a line leaving the center in the specified direction.

For example, a sphere centered around the origin is a representation of an isotropic distribution. All of the points in the diagram are equidistant from the center and therefore light flows equally in all directions (Figure 7-4).

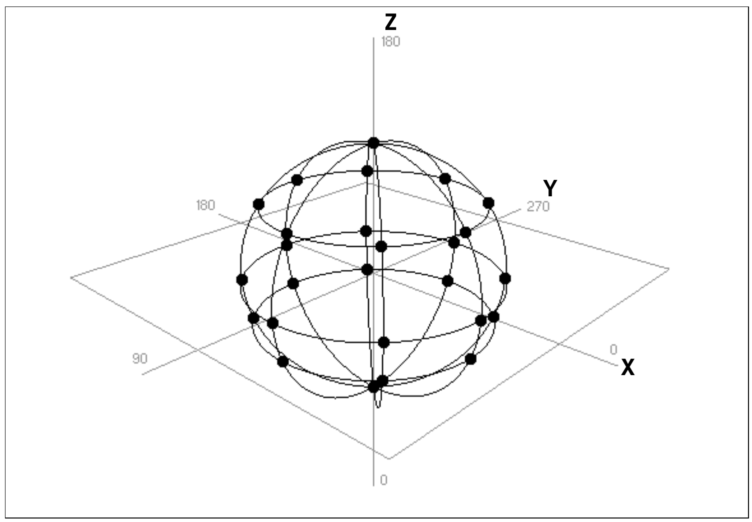


Figure 7-4. *Isotropic Distribution*

In the ellipsoidal example shown in Figure 7-5, the points in the negative Z direction are farther from the origin than the corresponding points in the positive Z direction, so more of the light shines down than shines up from this particular light source. No point has a very large X or Y component, either positive or negative, so very little light is cast laterally from the light source (Figure 7-5).

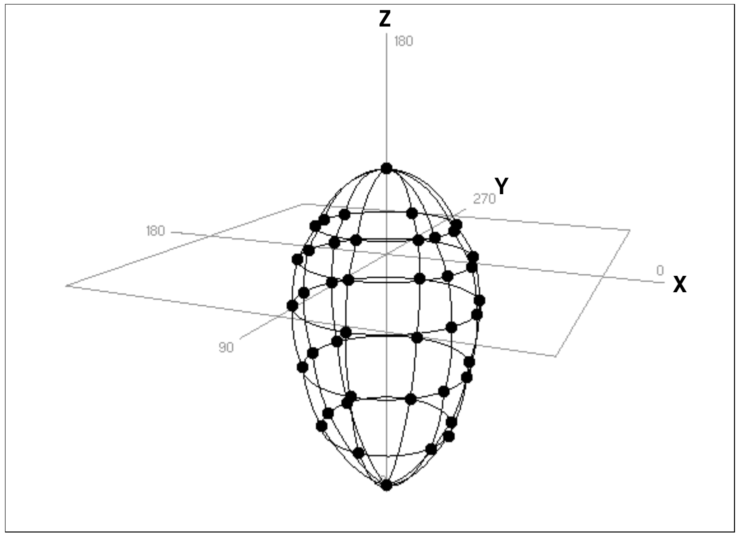


Figure 7-5. *Ellipsoidal Distribution*

You can use the photometric web to create very complex light distributions, including ones that are not likely to be used in reality, as shown in Figure 7-6.

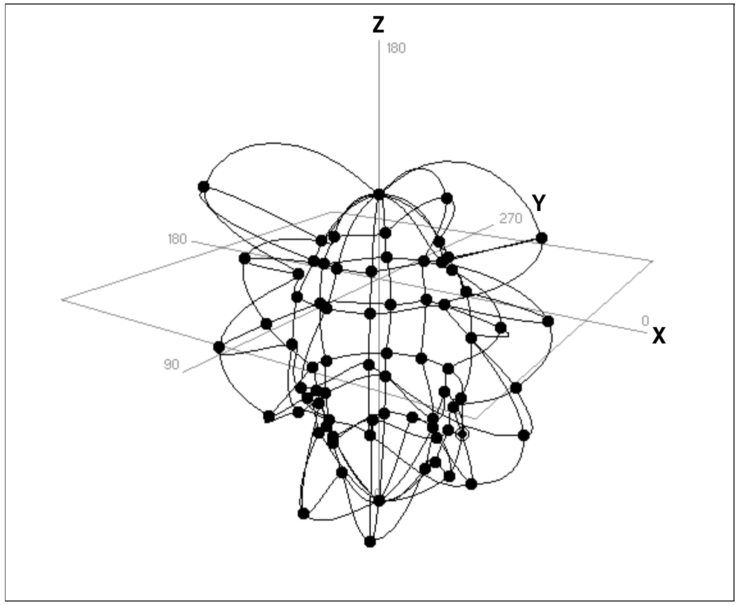


Figure 7-6. *Unusual Distribution*

The Photometric Web Editor

When you activate the Photometric Web editor (by choosing Light > Photometric Web), Lightscape displays the intensity distribution coordinate system in the Graphic Window.

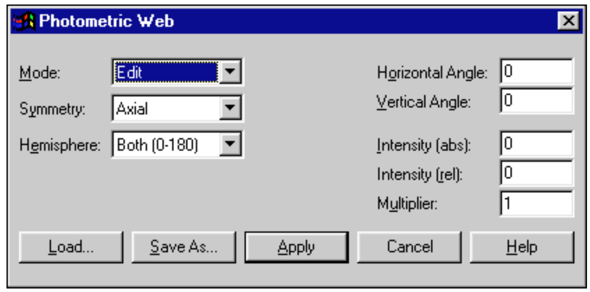


Figure 7-7. *Photometric Web Editor*

This system represents the energy distribution by a single white arc at horizontal angle 0 with a large white control point at either end, as shown in Figure 7-8. The intersection of the three axes represents the photometric center of the light. The white curved line represents the distribution pattern. The default graph represents an isotropic distribution, with the energy values equal in all directions.

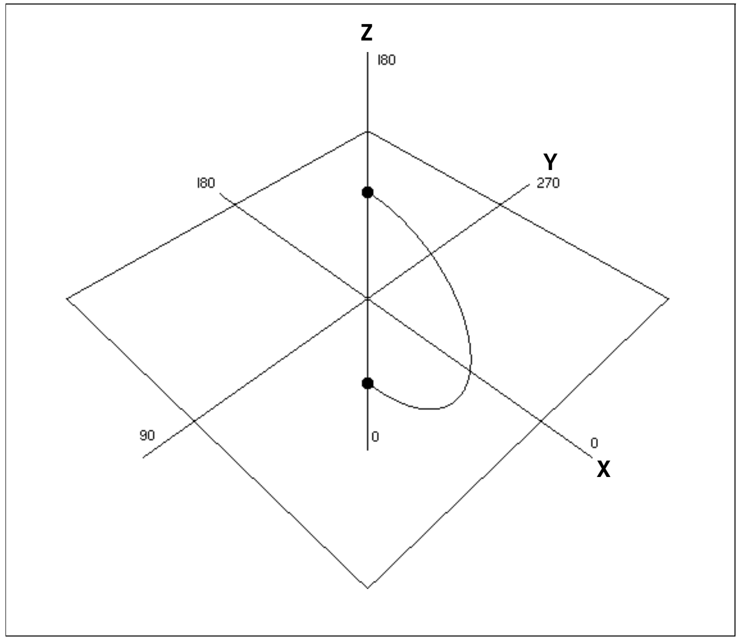


Figure 7-8. *Default Distribution*

You can add more control points to the distribution. You do this in Insert Mode by adding horizontal and vertical arcs, much like lines of latitude and longitude on a globe. As you add these arcs, the system creates control points at the intersections of each one.

To insert a horizontal arc, point at any existing vertical arc and press the left mouse button. The new horizontal arc passes through the point that you clicked. To insert a vertical arc, click any horizontal arc. (You can only insert vertical arcs if axial symmetry is not set.) While in Insert mode, a yellow arc shows what will be inserted.

In Edit mode, you can drag control points toward or away from the photometric center to change the distribution. You can also drag most vertical and all horizontal arcs. The motion of some arcs is constrained to ensure the validity of the distribution. For example, you cannot move or delete the vertical arc of horizontal angle 0. In addition to interactively dragging, you can also set the position of control points by choosing the point and entering explicit values. Arcs change accordingly.

Also, in Edit mode, the Photometric Web Editor shows the current arc's horizontal and vertical angles, the absolute and relative intensities, and the intensity multiplier. You can edit the intensity settings by entering new values in the input fields.

In Delete Mode, you can remove arcs by clicking on the arcs you want to delete. When the cursor points at an arc, the arc is drawn as a dashed gray line.

You control the overall character of the photometric web by selecting the symmetry and the hemispheres in which to place the distribution.

The default display shows a distribution that covers both hemispheres. Choosing only one hemisphere defines a light that shines either entirely up (top) or down (bottom).

You can enforce symmetry on the photometric web.

- *Axial symmetry* implies that the distribution is the same around all 360° of the light source (about the Z axis).
- *Mirror symmetry 0-180* implies that everything is mirrored about the XZ plane.
- *Quadrant symmetry* implies that everything is mirrored about both the XZ and YZ planes.

You can also have a distribution that is not symmetrical. When symmetry is enabled, Lightscape automatically enforces it while you edit, insert, and delete.

Once you have created a photometric web, you save the values in an IES file (Illuminating Engineering Society photometric data format) and can later associate them with a particular luminaire. Appendix E, “IES Standard File Format,” describes the format of the IES file.

Luminaire Processing

In addition to specifying the photometric characteristics of a luminaire, you can also specify the way you want the luminaire to behave during the radiosity processing using the Luminaire Processing parameters. You access the Luminaire Processing dialog box from the Graphic Window or Luminaires Table context menus.

During the Preparation and the Solution stages, changing these parameters for a luminaire instance affects only the selected instance. Setting these parameters for a luminaire definition, affects all instances of that luminaire definition.

You can control the following processing parameters from the Luminaire Processing dialog box:

- Cast Shadows
- Store Direct Illumination
- Ray Trace Direct Illumination

Cast Shadows

You can specify whether or not selected luminaires cast shadows. If you set luminaires to not cast shadows, Lightscape distributes the energy from the light to each surface in its path as if there were no other surface blocking it.

This considerably reduces the number of calculations required for a solution, so it's a quick way to get a general feel for the lighting characteristics of a model. However, this procedure does not produce accurate results and is generally not suitable for final solutions.

Store Direct Illumination

During a radiosity solution, all light used to illuminate a model, with the exception of sunlight and sky light, comes from direct light sources (luminaires). Normally, it is this light that you first see during the generation of the solution, after which light reflected from surfaces in the model (indirect illumination) is added to the solution.

This option, when turned off, lets you prevent Lightscape from displaying the direct illumination from the selected luminaires. The system calculates the light from the luminaire(s), but uses it only to generate indirect lighting. Essentially, you're eliminating the effect of direct lighting, leaving only reflected light to illuminate the model.

If you intend to ray trace a luminaire, you can save time by turning off the Store Direct Illumination option. If this option is off, Lightscape will not have to run iterations to subtract the direct contribution before ray tracing the luminaire.

Ray Trace Direct Illumination

This option lets you specify whether Lightscape (re)computes direct illumination from a selected luminaire during a ray tracing operation.

When ray tracing with the Ray Trace Direct Illumination option turned on, Lightscape removes the direct light contribution it calculated during the radiosity processing (unless the Store Direct Illumination option for the luminaire is turned off) and recalculates it with the ray tracer. Although this adds time to the ray tracing procedure, it also improves the quality of shadows and lighting effects in the final image.

If the Store Direct Illumination option is off during radiosity processing, you can still turn on the Ray Trace Direct Illumination option. This causes the ray tracer to add the direct illumination to the images it produces, while leaving the radiosity solution (stored in the mesh) unaffected. If you know you want to ray trace the direct illumination and do not need to compute realistic renderings using the GL display, you may want to set your luminaires to not store the direct illumination. This way, you do not need to remove the direct illumination prior to ray tracing, and may save you a considerable amount of time.

Natural Lighting

Natural lighting is provided by two sources—the sun and the sky. The sun is modeled as a parallel light source making the incident direction of the sun constant over the surfaces in the scene. The direction and intensity of the sun can be specified directly by the user or can be computed by Lightscape from the site location, time, and sky condition settings.

The sky is modeled as a sphere of infinite radius placed around the scene. The contribution of the sky light to the illumination of a point in the scene is computed by considering all directions around that point along which the sky is visible. The sky luminance (brightness) is not constant over the sky sphere (the sky dome) but changes with the direction under consideration. The shape of the sky light distribution is determined from the site location, time, and sky condition settings.

Because the sun is modeled as a parallel light source, Lightscape can compute its contribution to the illumination of the scene very efficiently. The sky, however, requires much more complex calculations. You may have to balance the trade-offs between accuracy and computation time when computing the sky illumination contribution. This contribution is computed by breaking down the sky dome into several small sectors, computing the illumination from each sector, and adding these figures to get the overall result. The higher the accuracy setting used, the larger the number of sectors, but the slower the computation.

Interior and Exterior Models

Because Lightscape sees the model as a group of polygons, computing the sky illumination onto the center of an interior model requires looking for sky contributions from all directions around this center. For the large majority of these directions, the sky is occluded by the walls and ceiling of the model. Only a few of the sky dome sectors considered during this computation are visible through a window. If the sky dome is not broken down into enough sectors, the system may determine that none of the sky is visible at all. Also, those sectors that are visible through a window will, in general, only partially overlap with the window. The system, however, adds in their contributions as if they are fully visible. This overestimates (or underestimates) the sky illumination.

Lightscape allows you to explicitly identify the surfaces in a room that are windows or represent openings. Knowing that the sky light can only reach the room floor through a designated window, allows the system to automatically eliminate any directions other than those going through the window, speeding up computation time.

To speed up computation time even further, Lightscape precomputes the illumination from the sky onto the window and then uses the window as a diffuse light source to illuminate the interior of the room.

Using this strategy (rather than computing the illumination from the sky through the window and onto a target point) has one drawback. Although the amount of light emitted from the window into the room's interior is correct, its directional distribution is replaced by a diffuse distribution. As a consequence, the ceiling receives somewhat more light than it should and the floor receives somewhat less light. (The appearance is more natural, however, because in most rooms, some light will reflect off the outside environment, up through the window, and onto the ceiling.)

When simulating the effect of daylight on an exterior scene, Lightscape looks at the entire sky dome to compute the illumination contribution from the sky.

The most expensive part of this calculation is the determination of the shadows cast by the sky dome. If the shadows are not a concern, you can turn off their computation and achieve dramatic savings in processing time. Once you turn off shadow calculations, however, the only factor determining the level of sky illumination at a surface in the scene is given by the surface orientation (surface normal). For example, the roofs of the buildings in a model of downtown New York will get the same level of illumination, no matter what their height and no matter what other buildings are next to them. This typically results in a very flat appearance for models where many surfaces are oriented in the same directions.

Lightscape allows you to control the trade-off between sky illumination accuracy and computation time. This accuracy is largely determined by the accuracy of the shadow computations. In fact, when shadow computations are turned off, the accuracy of the sky illumination is fixed (you can no longer adjust it).

When the accuracy of the shadow computations is low, there may be a high variance between the computed sky illumination at neighboring mesh vertices on a surface. This can result in a blotchy effect. The size of the blotches decrease as shadow accuracy increases.

The variance in the shadow computations is highest in models where sky is visible only through small openings. For these interior models, you should specify that light enters the scene through windows and openings only. This speeds up the radiosity computations and can also help reduce light leak artifacts.

When you use a natural light source, you use the Daylight dialog box to define conditions of where and when your model is located. You access the Daylight dialog box by choosing Light > Daylight.

Sun and Sky

Use the Sun and Sky page to define information about the sun and the sky. There are two sections of this page—Color and Sky Conditions.

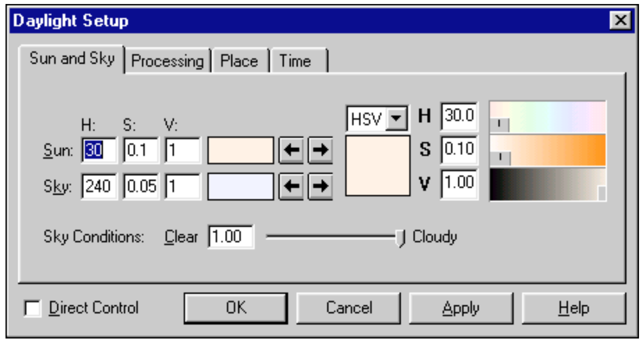


Figure 7-9. *Daylight Setup Sun and Sky page*

- The *Color* section contains a color selector for setting both the sun and sky colors.

  Use the color sliders to choose the color you want for each setting. You can use HSV or RGB values.

  As you adjust the color sliders, Lightscape displays the new color in the box adjacent to the sliders. When you are satisfied with the selected color, click one of the left arrow buttons to apply the color to a particular setting.

  You can make further adjustments to each color by clicking the right arrow button corresponding to the color you wish to edit. Lightscape displays the color you selected in the box adjacent to the color sliders. The color is now the active color so you can edit it.

- *Sky Conditions* uses a slider to approximate the fraction of the sky covered by clouds. 0 indicates a clear sky and 1 indicates total cloud cover.

Place

Use the Place page to specify information about the location for lighting your model. This page has a clock face for setting the North direction and a Location section for setting the exact location.

Figure 7-10. *Daylight Setup Place page*

- *North* specifies where north is in your model.

  That is, it lets you set the orientation of the model to the compass points. You can specify north by choosing a position on the clock face. The clock face shows the north direction relative to a top view of the model. You can also specify north by entering an angle directly.

- *Location* specifies the city or latitude and longitude of your model.

  You can set the location by using either a predefined location from the city file or entering a specific longitude and latitude. If you use a predefined location, you need to set the appropriate time zone (on the Time page) before selecting the city.

Time

Use the Time page to specify information about the timing for lighting your model.

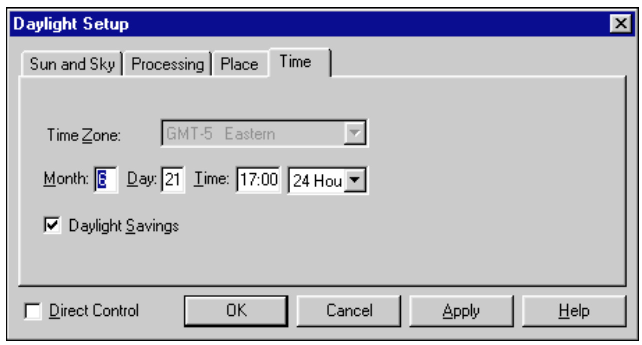


Figure 7-11. *Daylight Setup Time page*

- Using the *Time* settings, you specify the time zone, month, day, hours, and minutes for lighting your model. You also specify whether or not daylight savings time is in effect and whether you're measuring time using 12-hour or 24-hour time.

Direct Control

Selecting the Direct Control option at the bottom of the Daylight dialog box allows you to set the sun's direction and intensity directly.

Lightscape replaces the Place and Time pages with the Direct Control page.

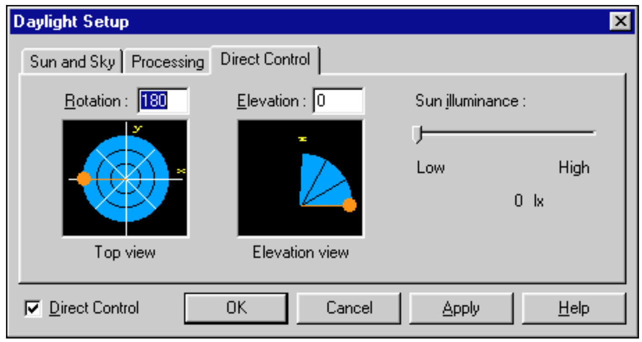


Figure 7-12. *Daylight Setup Direct Control page*

Set the Rotation and Elevation of the sun by dragging the orange pointers in their respective scales. The Rotation control is displayed in top view and the Elevation control is displayed in side view. You can also enter exact values in the input fields for each setting.

Use the Sun illuminance slider to set the normal illuminance of the sun in either footcandles (fc) or lux (lx), depending on whether you are working in American or International lighting units. (You set this with Edit > Properties.)

Processing

You can specify how you want light from the sun and sky to behave during the radiosity processing, using the Daylight Processing parameters. You access these parameters from the Processing page in the Daylight Setup dialog box (Light > Daylight).

You can control the following processing options for both sunlight and sky light:

- Cast Shadows
- Store Direct Illumination
- Ray Trace Direct Illumination

Cast Shadows

You can specify whether or not sunlight and/or sky light casts shadows. If you set sources to not cast shadows, Lightscape distributes the energy from the source to each surface in its path as if there were no other surface blocking it.

This considerably reduces the number of calculations required for a solution, so it's a quick way to get a general feel for the lighting characteristics of a model. However, this procedure does not produce accurate results, and is generally not suitable for final solutions.

Store Direct Illumination

Normally during a radiosity solution that incorporates daylight, the direct effect of daylight is calculated after the model is lit by any luminaires set to store and/or ray trace direct illumination. During the generation of the solution, you see this lighting effect immediately after direct lighting from luminaires followed by light reflected from surfaces in the model (indirect illumination).

This option, when turned off, lets you prevent Lightscape from displaying the direct illumination from the sun and/or sky. The system calculates light from the daylight sources, but uses it only to generate indirect lighting. Essentially, you're eliminating the effect of direct lighting from the sun and sky, leaving only reflected light to illuminate the model.

If you intend to ray trace the sun and/or sky, you can save time by turning off the Store Direct Illumination option. If this option is off, Lightscape will not have to run iterations to subtract the direct contribution before ray tracing the sun and/ or sky.

Ray Trace Direct Illumination

This option lets you specify whether Lightscape (re)computes direct illumination from the sun and/or sky during a ray tracing operation.

When ray tracing with the Ray Trace Direct Illumination option turned on, Lightscape removes the direct light contribution it calculated during the radiosity processing (unless the Store Direct Illumination option for the sun and/or sky is off) and recalculates it with the ray tracer. Although this adds time to the ray tracing procedure, it also improves the quality of shadows and lighting effects in the final image.

In order for this setting to take effect, the Ray Trace Direct Illumination option in the Rendering dialog box must be turned on.

Chapter 8

Solution

Once you add light sources and materials, the model is ready for radiosity processing. This step simulates the propagation of light through the environment and its interaction with the surfaces in the scene. You can modify light sources and materials at any time so you can quickly explore design alternatives.

This chapter discusses the radiosity solution process. At the core of this process is the simulation of light propagation through the environment and its interaction with the surfaces in the scene. Unlike traditional rendering systems, the result of this simulation is not a single image of the scene. Instead, Lightscape stores the illumination values computed during the simulation with the surfaces in the three-dimensional environment. You can use them to quickly generate images of the scene from any viewing location.

For a solution to be computed, the input model must include a specification of the light sources, materials, and texture maps associated with the surfaces in the environment. You define this data for a model during the Preparation stage.

During the Solution stage, you can modify the characteristics of light sources and materials at any time; the simulation compensates for the resulting changes in illumination. This feature promotes an interactive approach to design, so you can quickly evaluate and make refinements to obtain precisely the look you want.

Once your simulation is complete, you can generate presentation-quality images and walkthrough animations of the model. This is explained in Chapter 11, “Animation,” and Chapter 12, “Rendering.” You can photometrically analyze the results of the simulations using the tools described in Chapter 10, “Lighting Analysis.”

This chapter explores the Solution stage in detail, starting with a review of the basic concepts and an overview of the solution process, followed by a presentation of all the parameters that control the simulation, and ending with a discussion of the tradeoffs between computation time, memory requirements, and quality of results.

Basic Concepts

As discussed in the *Lightscape Visualization System Getting Started* guide, the lighting simulation software used in Lightscape is based on a technology called *radiosity*. Radiosity computes the illumination of a surface due both to light shining from a source directly toward the surface and to light reaching the surface indirectly after being reflected one or more times from other surfaces in the environment.

Meshing

Lightscape stores the computed illumination directly on the surfaces in the scene, so you can quickly generate multiple views of the environment from the same radiosity solution. In order to represent variations of illumination across a surface, Lightscape automatically breaks down the surface into smaller pieces, called *elements*. The simulation then computes the illumination from a light source to each of the corners, or *vertices,* of each element. The set of all the elements and vertices of a surface is a *mesh* (Figure 8-1).

Thus, rather than trying to store the illumination at every possible location on a surface, Lightscape only computes and stores the illumination at selected sample points, the mesh vertices. It then computes the illumination across any given mesh element by simply interpolating the illumination values stored at the vertices of the element.

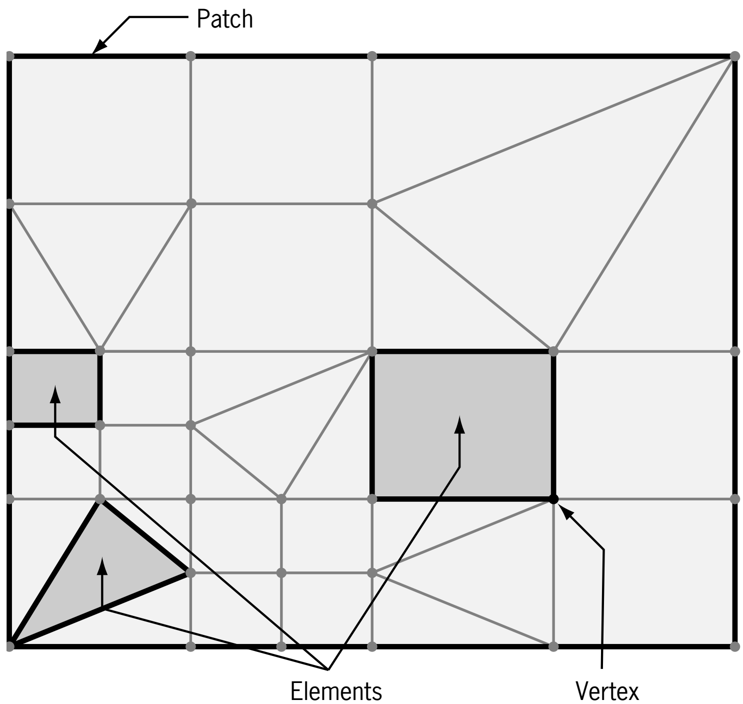


Figure 8-1. *Constituents of a Radiosity Mesh*

The number of mesh elements needed to capture the illumination across a surface depends on the complexity of the illumination. Small patches of light, shadow boundaries, penumbrae, and any other area across which the illumination changes quickly all add to the complexity of the illumination. The greater the amount of detail, the greater the number of mesh vertices and elements needed to capture it accurately.

In order to maintain as efficient a solution as possible, the system starts with a coarse mesh (that is, few elements) and automatically refines the mesh locally where high illumination gradients are detected. This process, called *adaptive meshing,* is controlled by a number of parameters you can vary to provide the optimal balance between computation time, storage use, and simulation accuracy.

Progressive Refinement

Lightscape computes the simulation in successive iterations. At each iteration, the system selects the brightest light source and computes its contribution to all the surfaces in the scene. Once all the primary light sources are accounted for, the system starts computing the interreflections of light between surfaces. At each iteration the system selects the brightest reflecting surface and computes its contribution to the rest of the environment.

This process is called *progressive refinement* because the system refines the radiosity solution at each iteration—that is, each iteration is a better approximation of the final result.

In principle, the refinement process continues until it accounts for all the multiple interreflections of light. In practice, however, the simulation converges very rapidly toward the final result, so that visual differences between successive iterations become unnoticeable after only a fraction of the surfaces (but the most important of them) have reflected their light contribution back into the environment.

Ambient Approximation

Because each progressive refinement iteration adds light to the environment, displaying the radiosity solution during processing shows an initially dark scene that gets brighter with every iteration. Rather than displaying this progression toward the final lighting of the scene, the system can add in a rough approximation of the yet uncomputed lighting so that the average brightness of the scene is approximately the same after every iteration. When you use such an ambient approximation during display, the lighting of the scene initially appears very flat and uniform; but at each iteration the system replaces this coarse approximation by all the subtle variations in lighting typical of radiosity solutions.

Overview of the Solution Process

The solution process starts by initiating the model (Process > Initiate). During this step, Lightscape breaks down every surface in the model into an initial coarse mesh. Because the illumination on a surface changes depending on its position and orientation relative to the rest of the scene, multiple instances of the same block must be "exploded" to their component surfaces.

Lightscape organizes the surface data during the Solution stage quite differently than it does during the Preparation stage. As a consequence, you cannot convert a Solution model back to a Preparation model.

After initiation, every surface in the model has a radiosity mesh with an illumination value of 0 at each of the mesh vertices.

The next step is to begin the processing by choosing Process > Go. You can also choose Process > Direct Only to calculate a solution that doesn't include reflected light.

Once the system starts the radiosity process, the progressive refinement iterations begin to propagate light to the surfaces in the scene. As each iteration completes, you can display the intermediate results of the simulation using the current display mode. You can also run radiosity solutions as a batch process, using the utilities described in Appendix C, "Lightscape Utilities."

You can interrupt and resume processing at any time. You can halt processing gracefully or abruptly. During a graceful stop (Process > Stop or press **Esc**), the system completes calculation of the current iteration. This preserves a consistent state where the current light source contribution has been distributed either to all of the surfaces in the environment or to none.

On the other hand, if the results are clearly not what you want, you can halt processing immediately (pressing **Shift+Esc**), before the system finishes the iteration. If you do and you make changes to the model, the solution will not be in a consistent state. In that case, you should reset (Process > Reset) and restart the solution after (or before) you make the changes.

You control the simulation with a number of parameters. Most of the parameters guide the adaptive meshing of the receiving surfaces and the way in which light sources contribute light to the environment.

These parameters give you a high degree of control over the tradeoffs between computation time, memory use, and solution accuracy. For example, if you make the adaptive meshing scheme generate more mesh elements, the accuracy and visual quality of the simulation improves, but the computation time and storage space required increase correspondingly. Selecting the optimal combination of process parameters requires an understanding of the simulation technique and some experience using the system.

Lightscape freezes the geometry of the scene at initiation time. However, you can modify materials and light source properties at any time during the simulation. Because the solution takes into account the interreflections of light between surfaces, even a simple change to the color of a single surface may affect the illumination of the entire scene. In these situations you can simply resume processing and the system automatically compensates for the change without having to reset the solution and restart the simulation from scratch.

Initiating the Model

Before the radiosity processing can start, Lightscape must *initiate* the model. During this step, the system converts the data describing the surfaces and light sources in the model to a more efficient form for radiosity processing.

Once the system completes this conversion, you can no longer create or reposition any of the surfaces or light sources. You must make any such changes to the scene geometry on the original Preparation model, reinitiate the model, and restart the radiosity processing. Consequently, you should always save your Preparation model before initiating it.

Although the initiation step does not change the form or surface characteristics of the objects in the scene, it substantially transforms the underlying data representation. The main steps of the conversion process are as follows:

1. Flattens the model hierarchy. Explodes all instances in the model into individual surfaces.

   The system stores the illumination values on the surfaces themselves. Since instances of the same block may have different illumination, their surfaces need to be separate.

2. Converts double-sided surfaces to two separate surfaces, oriented in opposite directions, each corresponding to one side of the original surface.

   The system stores the illumination in a radiosity mesh attached to the surface itself. You should use double-sided surfaces only where strictly necessary.

3. Groups the resulting surfaces into larger surfaces.

   In order to be part of the same larger surface, input surfaces must be on the same layer, share the same material and surface properties, be coplanar, and form a connected surface.

   The Length Tolerance parameter in the Process Parameters dialog box allows for inaccuracies (noise) in the input data. A surface lies in a given plane if all of its corners are within Length Tolerance distance from the plane. The system merges two separate vertices if they are within Length Tolerance distance apart.

4. Creates an initial radiosity mesh, with an illumination value of 0, for each resulting surface by connecting the vertices of the included input surfaces to form triangular and convex quadrilateral mesh elements.

5. May split long and skinny elements into smaller elements in an attempt to generate elements whose shape is close to an equilateral triangle or a square.

   Meshes made of well-shaped elements are more efficient and less likely to produce visual artifacts. However, improving the shape of mesh elements by splitting them can increase the overall number of mesh elements, to the point where the added computational cost of processing these new elements outweighs the original gain.

   For this reason, the Initialization Minimum Area parameter on the Process Parameters dialog box prevents the initiation process from subdividing mesh elements whose area is smaller than this value, allowing you to limit the number of mesh elements in the model.

For large models, the initiation step may be completed more quickly by making sure that no one layer contains a large number of input surfaces.

Once the initiation process is complete, the Solution model replaces the Preparation model in main memory. From this point on, you save the data as a Solution file with an .ls extension rather than as a Preparation file with an .lp extension.

Controlling the Simulation

After initiation, the model is ready for the lighting simulation process. The accuracy, speed, and memory usage of a radiosity simulation are controlled by a number of parameters, organized into two main groups: global controls and local controls.

Process parameters affect the simulation over the entire scene. *Surface processing parameters* only affect the processing of a particular surface or group of surfaces.

Understanding how these parameters control the simulation is the most important factor for generating high-quality and efficient radiosity solutions.

Process Parameters

Use Process > Parameters to modify global control parameters. This option brings up the Process Parameters dialog box. The parameters are in four groups:

- Receiver
- Source
- Process
- Tolerances

Receiver Control Group

You control the meshing of light receiving surfaces with receiver parameters. There is a direct correspondence between the number of mesh elements and the time and memory required to compute and display the radiosity solution.

If the mesh is too coarse, the results look crude and may contain visual artifacts. If the mesh is too fine, the visual effect may be outstanding, but the memory requirements and calculation time may grow beyond acceptable levels.

Running a short test using a crude mesh and working up to stricter parameters with a few more tests is often the fastest way to achieve the desired balance between solution quality and computational resources.

The following parameters control the mesh subdivision on the receiving surfaces:

- Minimum Mesh Spacing
- Maximum Mesh Spacing
- Subdivision Contrast Threshold
- Disable Solution Changes
- Lock Mesh

Minimum Mesh Spacing

Subdividing mesh elements based exclusively on illumination contrast can lead to excessive subdivision when a sharp shadow boundary crosses a surface. You limit the number of mesh elements that can be created as a result of the subdivision with the Minimum Mesh Spacing parameter.

No matter how high the illumination contrast, the subdivision does not create any new mesh elements smaller than the value of the Minimum Mesh Spacing. The size of a mesh element is defined as the length of its longest side and is shown in the current units of the model.

Maximum Mesh Spacing

Lightscape estimates the illumination contrast on a mesh element exclusively on the illumination values at its corners. Therefore, it cannot detect a small illumination feature that crosses the element, but does not cover any of its vertices.

You can minimize the chances of this problem with the Maximum Mesh Spacing parameter. The subdivision scheme splits the initial mesh elements so they are always below the specified size.

Subdivision Contrast Threshold

Rather than meshing a surface using a uniform grid of mesh elements, the simulation process uses a more sophisticated adaptive subdivision scheme to create smaller elements in areas that contain smaller illumination details (such as shadow boundaries) and larger elements in areas where the illumination is fairly constant. This is a very efficient technique in that it allocates processing resources to the areas of the model that require them.

The simulation starts by computing the contribution of the current light source to the vertices of the initial surface mesh. Then, for each mesh element, the system compares the values between the darkest and brightest of its vertices to compute an estimate of the illumination contrast over the element.

The *illumination contrast* is a measure of the variation in illumination across the given mesh element. A small contrast (close to 0) between two vertices of a mesh element indicates an approximately uniform illumination across the element. A larger contrast (close to 1) suggests that fine illumination details may cross the mesh element.

If the illumination contrast of an element is larger than the value of the Subdivision Contrast Threshold parameter, the system subdivides the element into four similar smaller elements and computes new illumination values for the new mesh vertices. It then computes the illumination contrast for the new elements and compares them against the threshold, possibly causing more subdivisions.

This process continues until the mesh elements are small enough to accurately reproduce the illumination of the surface of interest, or until the Minimum Mesh Spacing is reached.

Mesh Spacing Examples

The following pages contain four examples generated in Lightscape to show the effects of the various meshing parameters on the quality and efficiency of the mesh.

For each example, the figure on the left shows the display result and the figure on the right shows the generated mesh.

The wall is 5 meters wide by 3 meters high and a single spot source is pointed toward its center.

Example A **ReceiverMesh Sample Spacing**
Min 300 mm; Max 5000 mm
Subdivision Contrast Threshold 0.4

In this example (Figures 8-2 and 8-3), no light beam is visible because the maximum sample spacing is set too high (larger than the surface itself) and none of the original sample points fell within the beam of the light. There were no original sample points in the light beam, so no adaptive subdivision was triggered—in a sense, the light beam "fell between the cracks."

This demonstrates the significance of the maximum setting. It is important to select a value that ensures that at least one initial sample point falls within each light beam in your model. The default is a good starting point, but if you find that certain light sources don't seem to be illuminating the intended surfaces, it may be that the initial mesh parameter is too large.

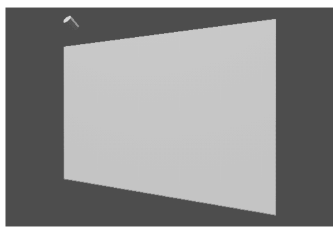


Figure 8-2. *Display*

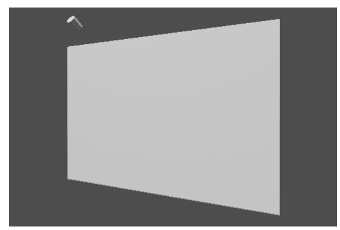


Figure 8-3. *Mesh (None)*

Example B Receiver Mesh Sample Spacing
Min 300 mm; Max 1000 mm
Subdivision Contrast Threshold 0.4

In this example, the maximum setting has been decreased but the result looks crude because the minimum sample spacing was not small enough to sufficiently capture the shape of the light beam (Figures 8-4 and 8-5). Notice the adaptive subdivision around the light.

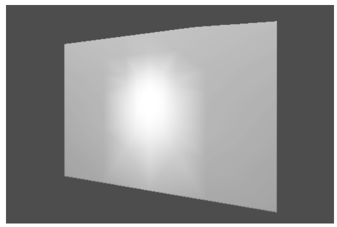


Figure 8-4. *Display*

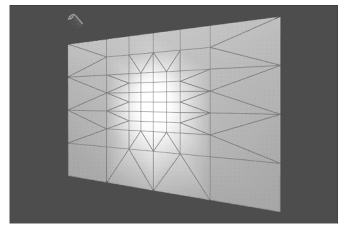


Figure 8-5. *Mesh*

Example C **Receiver Mesh Sample spacing**
Min 100 mm; Max 1000 mm
Subdivision Contrast Threshold 0.4

The result looks much better, although the number of mesh elements generated is greater (Figures 8-6 and 8-7). The mesh is well shaped, meaning that the adaptive subdivision has been triggered only where desired—near the light beam.

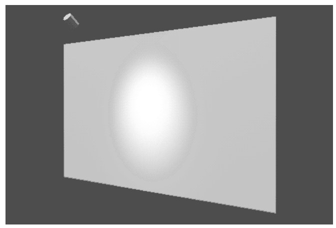


Figure 8-6. *Display*

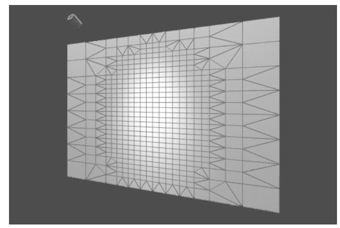


Figure 8-7. *Mesh*

Example D **Receiver Mesh Sample spacing**
Min 100 mm; Max 1000 mm
Subdivision Contrast Threshold 0.1

The only difference between this example and the previous one is that the Subdivision Contrast Threshold has been changed to make it more sensitive to adaptive subdivision (Figures 8-8 and 8-9).

The final image looks the same as the one in Figure 8-6, but the mesh display shows that the whole surface unnecessarily subdivided to the Minimum Sample Spacing. Although the display results are the same, this example generated a considerably larger number of mesh elements, most of which just wasted processing time and memory.

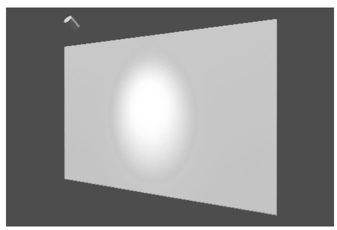


Figure 8-8. *Display*

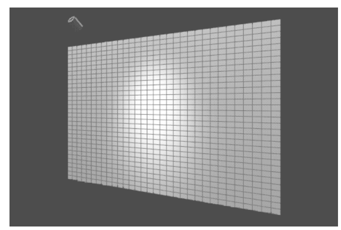


Figure 8-9. *Mesh*

Disable Solution Changes

So that you can change surface materials and light sources and be able to compensate for the resulting change in the illumination without restarting the solution process from scratch, the system must be able to undo the effect of one or more light sources (primary and secondary). The system undoes lighting effects by propagating negative light from the source to the receiving surfaces, thus removing light from the illumination of the scene.

During this step it is important that the mesh subdivision be exactly the same as that resulting from the original positive light contribution from that source.

The system guarantees this requirement. There is, however, a cost—it uses a variation of the meshing scheme that may increase the number of mesh elements a small amount. If you know you will not want to make any changes to a solution, you can use the Disable Solution Changes parameter to obtain a more efficient result.

Running a simulation with Disable Solution Changes on doesn't prevent you from later changing surface materials and light sources. However, if you do so, the system warns you that it may not be able to correctly compensate for such changes in the radiosity solution or when refining shadows with the ray tracer.

Lock Mesh

Turning on the Lock Mesh parameter prevents successive iterations of the lighting simulation from subdividing any surface mesh further than the current configuration.

Also, if this parameter is on when you reset a solution, the system restores all illumination values to 0, but preserves the current mesh subdivision.

This feature is useful for advanced, special applications where you need to preserve the arrangement of the mesh elements. Generally, you should leave this parameter off.

Source Control Group

The source parameters control how accurately Lightscape computes the contribution from a light source to each of the receiving mesh vertices.

The following parameters allow independent controls for direct light sources (luminaires, windows, and openings) and indirect light sources (surfaces):

- Direct Source Minimum Size
- Direct Source Subdivision Accuracy
- Indirect Source Minimum Size
- Indirect Source Subdivision Accuracy
- Shadow Grid Size

Direct Source Minimum Size

In certain geometrical configurations, such as when an area source shines light onto an adjacent surface, the subdivision criteria may break down the source into too many regions.

The source subdivision does not generate regions smaller than the threshold you specify with the Direct Source Minimum Size parameter.

For most cases, setting this parameter to the same value as the receiver Minimum Mesh Spacing produces good results. However, there may be times when reducing the minimum size of the source helps prevent visual artifacts, as discussed in "Handling Meshing Artifacts" on page 8-35.

Direct Source Subdivision Accuracy

The energy contribution of a point light source to a receiving target (a receiver mesh vertex) is directly proportional to the luminous intensity (brightness) of the source in the direction of the target and inversely proportional to the square of its distance from the target.

For linear and area light sources, the direction and distance from the receiving target change across the source. If the target is very far from the source, the source can be treated as a point source without introducing any significant errors in the computations. However, if the target is close to the source with respect to the size of the source, then treating linear and area sources as point lights would lead to inaccurate results.

To prevent this problem, Lightscape subdivides the source so that each resulting piece is small when considered from the point of view of the particular mesh vertex under consideration. This subdivision is very similar to that of the receiver mesh, but is less intuitive because the system cannot let you visualize it. Furthermore, the source subdivision can change for every receiving target, since it depends on the distance between the two.

You control the accuracy of the computed light transfer from a linear or area source to the receiving target with the Direct Source Subdivision Accuracy parameter. The value of this parameter determines the likelihood that Lightscape will subdivide the source. When you set the parameter to 0, the system never subdivides these sources. As you increase its value towards 1, the subdivision is triggered more easily and for more distant targets.

This parameter does not affect point sources or natural lighting, except for window/opening sources in interior models.

Take care when manipulating this parameter. If the value is too low, illumination from an area light may look like that caused by a point light or even by a grid of point lights. If its value is too high, the accuracy of the calculation may be remarkable, but the computation speed may be unacceptably low.

Indirect Source Minimum Size

You specify the minimum possible size for secondary sources with the Indirect Source Minimum Size parameter. It works in the same way as the Direct Source Minimum Size.

Indirect Source Subdivision Accuracy

You control the accuracy of the computed contribution from a secondary source to the rest of the surfaces in the scene with the Indirect Source Subdivision Accuracy parameter. It works in the same way as the Direct Source Subdivision Accuracy parameter.

The recommendations made for the Direct Source Subdivision Accuracy parameter apply to this parameter as well. In general, you can set the indirect sources to match the direct sources. For certain models, or to reduce the processing time, you may decide that indirect sources don't need to be calculated to the same level of quality as the direct sources.

Shadow Grid Size

The amount of light transferred from a source to a receiving target depends on the strength of the source and its position and orientation with respect to the target. It also depends on the possible presence of other objects in the scene acting as obstacles between the source and receiver.

Lightscape tries to estimate the attenuation (falloff) of light due to possible occlusions by casting rays from the target toward the source. It computes the attenuation factor as the fraction of rays cast that actually reach the source without being blocked by any other obstacle. You control the number of rays cast between a receiving point and a source with the Shadow Grid Size parameter.

For linear sources, the value of this parameter is the number of rays cast. For area sources, indirect sources, and windows, the system casts these rays toward a regular grid of points spread over the source. This grid size is equal to the control parameter in each direction. In other words, the number of rays cast is equal to the square of the value of the control parameter.

Setting this parameter to 1 results in fast computation times, but may also cause the shadows to appear too sharp. Increasing its value improves the quality of the radiosity solution, but also increases computation times.

In general, you should increase this parameter hand in hand with the Subdivision Accuracy parameters. Finding the best values for these parameters requires some experience and experimentation. For example, if you set the Shadow Grid Size parameter to 1, the shadow of a table cast by an area source onto a floor always appears too sharp, no matter how much you subdivide the mesh of the receiving surface.

Furthermore, setting the Shadow Grid Size parameter to a small value may not always result in faster processing. In fact, the overly sharp shadows may trigger unnecessary subdivision of the receiving surface, thus consuming more processing time and memory.

Process Control Group

The following process parameters control how textures, shadows, and daylight participate in the lighting simulation:

- Textures
- Shadows
- Daylight

Textures

You control whether or not textures are taken into account when computing the radiosity solution with the Textures parameter.

When this parameter is on, the system color filters light going through transparent objects with textured materials according to the texture values. A typical example of this situation is a stained glass window.

Also, when the Textures parameter is on, a textured surface reflects light back into the environment by coloring the incoming light according to the texture values. This feature yields more accurate color bleeding effects.

In general, if you use textures in the materials of the model, you should use textures during the simulation to obtain an accurate result. If you disable use of the textures during the simulation to save memory, you can preserve a certain degree of accuracy by setting the reflectance of the textured material to be the same as the average reflectance of its texture. (See Chapter 3, "Materials.")

Shadows

Computing shadows is the most expensive part of the simulation. When you run the initial tests on a new model, you can significantly accelerate processing by turning off the Shadows parameter.

Of course, allowing light to go through obstacles unaffected means the results of the computation will be incorrect. However, this feature can prove extremely useful for rapidly testing the position, orientation, and strength of light sources in relation to the receiving surfaces and for testing the meshing configurations for receiving surfaces. Once you have adjusted all of these parameters, you can reset the solution, turn on shadow computations, and start a physically accurate simulation.

When you turn on shadow computations, you can use the Direct Only parameter to determine whether the system computes shadows only for light cast by direct sources or for light cast by indirect sources as well.

Once again, testing for shadows for direct sources only is not physically accurate, but can significantly speed computation when you're interested in the visual appearance of a solution, rather than its physical accuracy.

Daylight (sunlight + sky light)

You control whether natural lighting should be included in the computation with the Daylight parameter. If this parameter is on, Lightscape includes sunlight and sky light as light sources for the model. If the model is an interior environment, turn on the Daylight Through Windows and Openings Only option.

If the model is an interior environment, natural lighting only affects receiving surfaces that can be reached through at least one window or opening. (See "Surface Processing Parameters" on page 8-29.) Sunlight still takes one iteration during the lighting simulation, but the system breaks down sky light so that its contribution is distributed among the windows and openings in the scene. In this case, each window and opening requires its own iteration to distribute its light contribution to the environment.

Because sunlight is orders of magnitude stronger than artificial lighting, Lightscape always processes it during the first iteration of the simulation.

Sky Light Accuracy

The Sky Light Accuracy parameter controls the accuracy of the sky light computations when the sky light Cast Shadows option is selected. This parameter only affects the radiosity iteration that accounts for the illumination from the sky dome. The iterations corresponding to sunlight, windows or openings, and luminaires are unaffected.

The smaller the value of the sky light accuracy, the faster the computation, but the lower the accuracy. Low accuracy can lead to blotchy illumination artifacts. As the value of sky light accuracy increases, these artifacts become smaller. As the value increases, however, the time required to compute the illumination from the sky increases very quickly.

Daylight Through Windows and Openings Only

When the Daylight option is on, you can select Through Windows and Openings Only. If this option is on, sunlight illuminates only those areas (mesh vertices) that can be seen through surfaces in the scene that are marked as windows or openings. The sky light is computed as the sum of the contribution of the light emitted by these windows and openings. This method improves visual quality and computation speed, and prevents light leaks from the sun that can occur in scenes that are modeled incorrectly (for example, using a single surface for a wall that should be modeled as two surfaces with the correct thickness).

If you are modeling an exterior scene, do not select this option.

If your scene is not clearly an interior or an exterior scene, you can calculate the daylight contribution in one of two ways. If most of the sky is occluded by objects in your model and can only be seen through cracks or relatively small openings, you should cover those cracks with actual surfaces, mark those surfaces as openings, and select Through Windows and Openings Only. These surfaces will be used as placeholders during the daylight computations and will not be rendered in the final images. Your model does not need to be “air tight.” Simply add surfaces that approximately cover the cracks through which daylight can be seen.

If most of the "sky dome" is visible, however, you should not select Through Windows and Openings Only. Instead, select Cast Shadows for the sky source and set the sky light accuracy slider in the Daylight Setup dialog box to a relatively large value. If your surfaces look blotchy after the daylight computations, try increasing the sky light accuracy value.

Tolerances Control Group

The parameters in this group control tolerances used in various parts of the computations to allow a certain level of imprecision in the input data and the numerical approximations required to implement arithmetic operations on real quantities. The following parameters are available:

- Length Tolerance
- Initialization Minimum Area
- Ray Offset

Length Tolerance and Initialization Minimum Area

The Length Tolerance and Initialization Minimum Area parameters were discussed in "Initiating the Model" on page 8-8, in relation to their role in initiating the Preparation model for the lighting simulation. The Length Tolerance parameter is also used during the computation of light transfer between source and receivers. The value appropriate for initiating the model usually works for this task as well.

Ray Offset

You can use the Ray Offset parameter to prevent numerical approximations from affecting the accuracy of the shadowing computations discussed in "Shadow Grid Size" on page 8-22.

Because of these approximations, the ray cast from a surface to a source is sometimes found to intersect an adjacent surface very close to the origin of the ray. The Ray Offset parameter specifies the minimum distance from the origin of the ray before the system can consider an intersection valid.

The value of this parameter is usually slightly greater than that of the Length Tolerance parameter. Setting the value to 0 may result in shadow artifacts.

Process Parameters Wizard

As an alternative to setting the process parameters manually, you can use the Process Parameters Wizard. This feature asks you a series of questions and then, based on your responses, automatically sets the parameters for you.

When setting parameters with the wizard, various aspects of the model are considered (such as the size of the model). For this reason, the parameters set by the wizard for one model may be different from the those set for another model.

To use the wizard, click the Wizard button in the Process Parameters dialog box. Move through the wizard by clicking Next when you are finished with the current page. You can also click Back to return to the previous page to review your settings or make changes. Click Finish when you are ready to calculate the solution based on the current parameters.

Surface Processing Parameters

With the Process Control option on the Surface Context Menu, you can set local control parameters for specific surfaces in the model. Use these parameters to fine tune the radiosity process to maximize the quality of the results while minimizing computation time and storage requirements.

The following parameters are available:

- Occluding
- Receiving
- Reflecting
- Window
- Opening
- Solve with Textures
- Display Raw Textures
- No Mesh
- Mesh Resolution
- Reset Mesh

If you change any of these parameters after starting processing, the system considers them only for the part of the process run after the change. To affect the complete radiosity solution, you must reset the solution and start again.

Occluding

The Occluding parameter controls whether a surface casts a shadow or whether light goes straight through it unaffected.

Surfaces are occluding by default.

Receiving

The Receiving parameter controls whether light reaching a surface is recorded in its radiosity mesh. You can turn this parameter off to save computation time on a self-emitting surface. (See “Luminance” on page 3-12.) The initial luminance of such a surface, in fact, may be much larger than that resulting from the illumination incident on the surface, so the effect of omitting that contribution would be unnoticeable.

Surfaces are receiving by default.

Reflecting

The Reflecting parameter controls whether a surface should reflect light incident on it back into the environment.

One useful application of this feature is found in lighting analysis. By turning off the Occluding and Reflecting properties of a surface, you can place the surface anywhere in a scene and use it as a workplane or sensor to measure the illumination incident on it without otherwise affecting the illumination of the scene. For more about this application, see “Workplanes” on page 10-10.

When using IES photometric distributions in luminaires, you may also want to set the surfaces of the luminaire to be non-reflecting so that energy is no emitted twice.

Surfaces are reflecting by default.

Window

The Window parameter controls whether a surface is considered a window and treated as a source during natural lighting computations. You must give the window a transparent material so that natural lighting can go through it.

Opening

The Opening parameter is similar to the Window parameter. When a surface is marked as an opening, it is not considered as part of the scene and does not receive or reflect light. Instead, it is used as a placeholder to indicate that natural lighting can go through it to reach the surfaces of an interior environment. Surfaces marked as openings are not rendered and are not displayed in the model.

Solve with Textures

The Solve with Textures parameter controls whether a texture-mapped surface should use texture values during the lighting simulation (see "Process Parameters" on page 8-10). So that a surface can use its texture map during processing, both this parameter and the corresponding global Textures parameter must be turned on.

By default, the Textures parameter is on for each surface.

Display Raw Textures

The Display Raw Textures parameter controls whether a texture is displayed with lighting from the radiosity solution. You should use this parameter for surfaces with textures on which you have performed the mesh to texture conversion and now have lighting information embedded in the texture itself. Turning on the Display Raw Textures parameter tells Lightscape not to relight the texture. You can also use this parameter for any surfaces on which you don't want Lightscape to calculate lighting effects.

No Mesh

The No Mesh parameter controls whether mesh subdivision on a surface is allowed at all.

Mesh Resolution

You can use the Mesh Resolution parameter to improve the quality of a radiosity solution without significantly affecting its cost. Meshing artifacts in a radiosity solution often appear on only a few surfaces in the scene. Rather than trying to eliminate the problem by changing the global meshing parameters, it may be much more efficient to adjust the meshing controls on the individual problem surfaces.

This parameter scales the minimum and maximum mesh spacing for the selected surfaces. If the global minimum value is 12 inches, setting this parameter to 2 increases the mesh resolution by dividing the global minimum in half (to 6 inches) and applying it locally to the selected surfaces. The maximum mesh value is also halved. In addition, the global Subdivision Contrast Threshold is decreased, making it more likely that the system will subdivide the mesh elements to capture illumination details cast over the surface.

Setting this parameter to a value less than 1 decreases the likelihood of triggering the mesh subdivision process.

Reset Mesh

The Reset Mesh parameter (available only during Solution stage) resets the radiosity mesh of a surface to its coarsest state, with all the illumination values at its vertices set to 0.

Changing Surface Materials and Light Sources

Once you've computed a radiosity solution, you can still modify light sources and materials to fine tune the appearance of the final rendering or to explore different design alternatives.

Rather than restarting the simulation every time you make a change, the system compensates for the changes incrementally, starting from the current solution. This allows you to quickly evaluate the solution and make refinements to obtain precisely the look you want.

Changing Surface Materials

You use the same procedure to change the material of a surface as during the Preparation stage. You either redefine the properties of a material or you create a new material and assign it to specific surfaces.

When you change a material, the system can immediately display the new material on all surfaces to which the material is applied. If the original surface had reflected light into the environment, the changes due to its contribution are not calculated or displayed until you run additional iterations of the radiosity process.

If there is considerable color bleeding and you change many materials, it may be better to reset the solution and restart the processing once you have set the desired materials. This provides a more accurate result. It is good practice to run a test solution, perhaps without shadows, to confirm and set the final materials.

CAUTION If you run the solution, change surface materials, and then ray trace the result while using the Luminaire Processing Ray Trace Direct Illumination option, you will get inaccurate results. If you plan to ray trace your solution, you should reset the solution and restart processing after any changes to surface materials.

Changing Light Values

In addition to changing surface materials, you can also redefine the photometric characteristics of a single luminaire or all the luminaires of a particular name. Remember that you cannot change the position of the luminaire during the solution process. You have to return to the Preparation model to make geometric changes.

Changes to the lighting characteristics work properly only if the Disable Solution Changes control parameter is off when the solution is run. (For information on this parameter, see "Disable Solution Changes" on page 8-18.)

When you change a light source, the system responds by first canceling the original energy distributed from the light. This requires one iteration; in a second iteration, it adds in the direct illumination for the new light source. Computing the indirect illumination due to the new light source throughout the environment may require further iterations.

CAUTION If you run the solution, change surface materials, and then ray trace the result while using the Luminaire Processing Ray Trace Direct Illumination option, you will get inaccurate results. If you plan to ray trace your solution, you should reset the solution and restart processing after any changes to surface materials.

Handling Meshing Artifacts

Because of how the system generates a radiosity mesh, there are a number of visual artifacts that can appear in a radiosity solution. This section examines ways of minimizing their effect.

Some of the artifacts can be managed in Lightscape; others can be avoided by taking additional steps during the modeling process. Some artifacts may be unavoidable or may simply not be significant enough to warrant the additional effort or memory required to eliminate them.

You may encounter:

- Jagged shadow boundaries
- Shadow leaks
- Light leaks
- Floating objects
- Mach bands
- Streaky shadows

Jagged Shadow Boundaries

During adaptive subdivision, Lightscape always subdivides existing mesh elements into four parts by inserting a new vertex at the midpoint of each element edge. Typically, a shadow or light beam does not align with the mesh resulting from this procedure. This can lead to shadow boundaries that look jagged or stepped.

Figure 8-10 shows the radiosity solution of a sharp spotlight on a wall. Notice that the edges of the spotlight are jagged. Figure 8-11 shows the mesh of this solution, demonstrating that the smallest mesh elements are still rather large compared to the illumination details they are trying to capture.

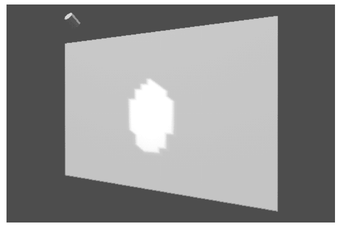


Figure 8-10.

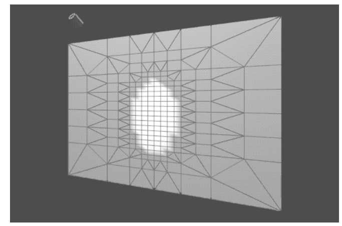


Figure 8-11.

The easiest way to alleviate this problem is to decrease the minimum mesh spacing, either for the entire environment or preferably just for the problem surfaces.

Figures 8-12 and 8-13 show the same scene computed with a minimum mesh spacing four times smaller than in the previous example. Although this looks better, it also required about five times the number of mesh elements.

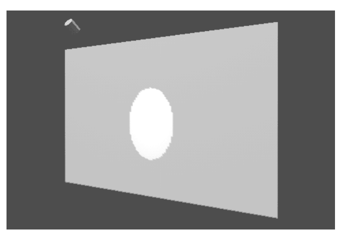


Figure 8-12.

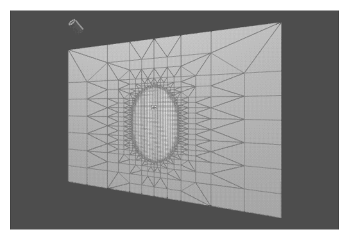


Figure 8-13.

If you have an environment that has a lot of sharp shadow boundaries, such as sunlight or spotlights, generating such a fine mesh can be prohibitive from a memory standpoint.

Another way to avoid jagged shadow boundaries is to ray trace the light sources that generate the sharp shadow by using the Ray Trace Direct Illumination option of the ray tracer. (See Chapter 12, “Rendering.”) This option requires you to turn on the Ray Tracing option (in the properties of the luminaire) for the light sources you want to ray trace. This alternative produces the best visual result.

The benefit of ray tracing these light sources is that the underlying mesh during the radiosity solution can be relatively coarse, as long as there is enough light to ensure some interreflections. That is, you have to see some light on the wall from the radiosity calculations. Figure 8-14 shows the original radiosity solution from which the ray traced image of Figure 8-15 was generated.

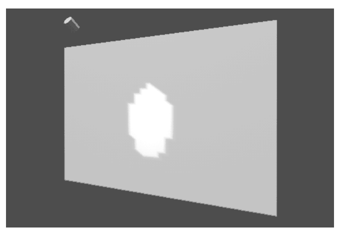


Figure 8-14.

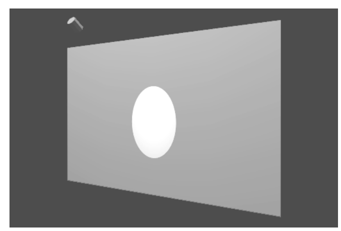


Figure 8-15.

Ray tracing light sources, however, can add a significant amount of time to the process, so you only want to ray trace those lights that appear in a final image. Fortunately, you can set the ray tracing option even after the radiosity solution is complete (see Chapter 7, “Lighting”). In this way you can first evaluate the solution from a particular view before deciding which shadows or light sources you need to refine in the final image.

Finally, for spotlights, it’s easier to get a good radiosity result with a sparse mesh if the edges are soft—a flood light, for example. Figures 8-10 and 8-16 were generated from the same mesh parameters and have the same intensity values. The light in Figure 8-10 was defined with a beam angle of 30° and a field angle of 30° (a sharp spotlight); the light in Figure 8-16 has a beam angle of 30° and a field angle of 90°. Of course, if you use a photometric web distribution or if you really want a sharp spotlight, then you have to resort to a finer mesh or to the ray tracing process described previously.

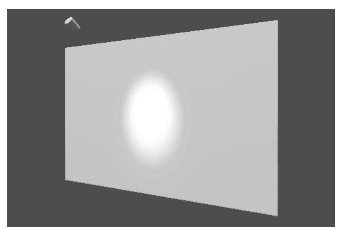


Figure 8-16.

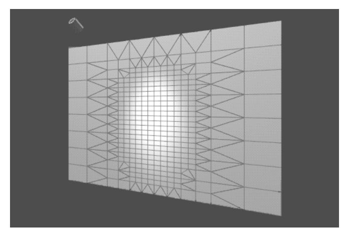


Figure 8-17.

Shadow Leaks

As the name implies, a shadow leak appears as a dark region that seems to start from under an object or wall and "leaks" out to the surrounding surface.

For example, consider the panel against the wall in Figure 8-18. The mesh generated for this radiosity solution (Figure 8-19) shows that one of the initial mesh vertices on the wall surface occurred behind the panel. Although there was some adaptive subdivision, the minimum mesh spacing was again too large.

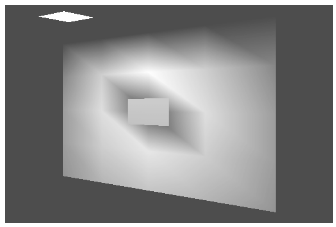


Figure 8-18.

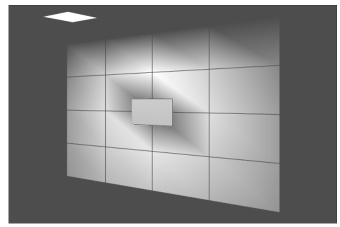


Figure 8-19.

The system renders radiosity solutions by interpolating the color between mesh element vertices. The color interpolation between the mesh vertex behind the black panel and the bright mesh vertices outside the panel caused the shadow leak.

There are four ways to alleviate this artifact.

The first is to model the wall so the intersection between the wall and panel is explicit. This can be a lot of trouble, but in certain situations it is worthwhile to explicitly define the edges between two surfaces. For example, Figures 8-20 and 8-21 show two ways in which you could model two intersecting beams.

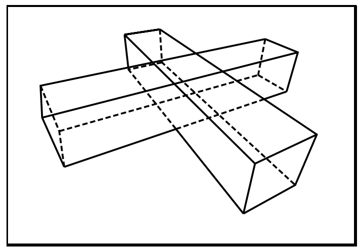


Figure 8-20.

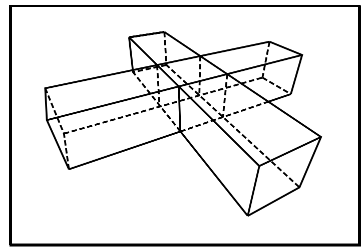


Figure 8-21.

The method shown in Figure 8-20, in which you model the two beams as two long intersecting beams, is not a good technique for radiosity. Although easier to model, sample points can occur on the surfaces of the beams inside the region of their intersection, possibly leading to shadow leaks. By being explicit about the surfaces and their intersections, as in Figure 8-21, you can avoid this possibility.

In general, taking extra care to be as explicit as possible about edges during the modeling process leads to a better solution in Lightscape. This does not mean you need to worry about every point of intersection. For example, you do not have to model a floor to cut around the legs of a table sitting on it; nor do you have to cut walls around light switches.

A second way to alleviate the shadow leak behind the panel is to decrease the minimum mesh spacing to trigger adaptive subdivision so that the edge is properly defined. This approach is shown in Figures 8-22 and 8-23. The problem with this approach is that the system generates a large number of elements to render a rather insignificant part of the model.

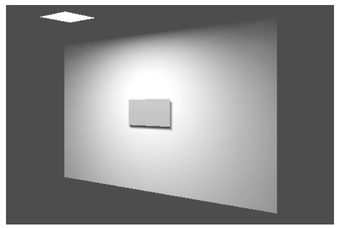


Figure 8-22.

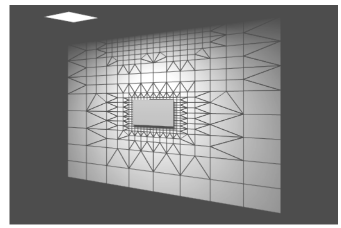


Figure 8-23.

A third approach is to ray trace the light (Figure 8-24). With this approach you can keep the sparser mesh. However, this approach is only good for single images. Ray tracing light sources also adds time to the ray tracing process.

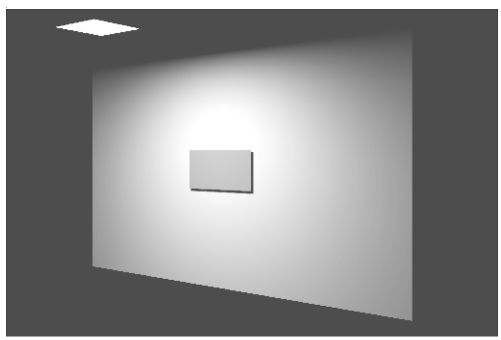


Figure 8-24.

The final way to alleviate the shadow leak problem is the simplest—you can decide to ignore the surface causing the shadow leak. If one surface is directly on another surface, such as a light switch panel on a wall, you can set the surface on top to be nonoccluding—the light simply passes through the surface and does not cast any shadows. In this kind of situation, the nonoccluding approach is the easiest and most efficient.

Light Leaks

Light leaks are the opposite of the shadow leak artifact just discussed. They appear as light extending into a darker region of a surface.

A typical example of a light leak is caused by using a single surface to model the floor of two adjacent rooms. If one room is lit and the other is not, light incident on the floor of the first room can crawl under the separating wall and onto the floor of the second room.

As with shadow leaks, you can prevent light leaks by modeling the floor in two separate pieces or reduce them by increasing the mesh subdivision of the floor during radiosity processing.

As another example, consider a model of a room illuminated by daylight entering through a window. Somtimes the walls and ceiling of that room are modeled as single polygons, not as two surfaces with the correct thickness between to represent the thickness of the wall. Illumination values are computed and stored at the vertices of the mesh capturing the illumination on the floor, and then interpolated across the mesh elements to display the results. On a sunny morning, the sun will cast light on the mesh vertices along the east edge of the floor because there is no wall thickness to block it. Even though the interior side of the wall is above those vertices, it cannot prevent sunlight from coming in from the side.

A solution to this problem is to specify that daylight should enter through windows and openings only. Lightscape then makes sure that sunlight rays cross a window or opening before illuminating any interior spaces. Thus, only mesh vertices that can see the sun through a window or opening are illuminated.

Floating Objects

Consider the table shown in Figures 8-25 and 8-26. Here the initial sampling mesh does not fall under the leg of the table because the surface area of the connection is small in relation to the overall area of the floor. Consequently, the system cannot trigger adaptive subdivision and completely misses the shadow of the table leg on the floor. This produces the visual effect of the table floating over the floor. An enlargement of the table is shown in Figures 8-27 and 8-28.

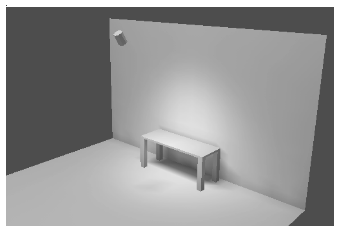


Figure 8-25.

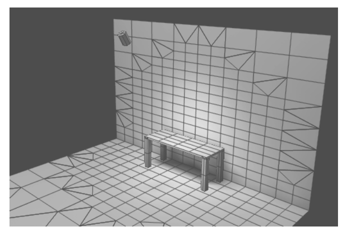


Figure 8-26.

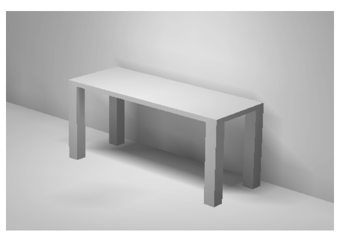


Figure 8-27.

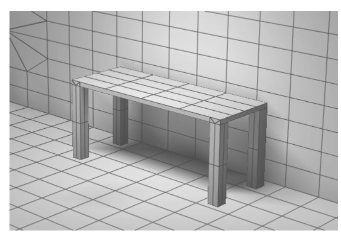


Figure 8-28.

The problem is there was no original sample point in the floor area shadowed by the leg of the table that would have triggered an adaptive subdivision. Setting finer mesh parameters can correct the situation (Figure 8-29, 8-30, and 8-31).

In general, it is difficult to avoid this artifact because it is impractical to make the initial mesh small enough to guarantee obtaining a sample point inside every shadow region. The best solution in this situation would be either to force a mesh element to occur under the table leg by being explicit during the modeling stage (as with the shadow leak artifact) or to ray trace the light in a postprocessing step (Figure 8-32).

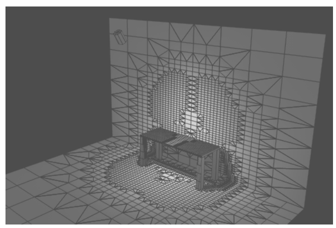


Figure 8-29.

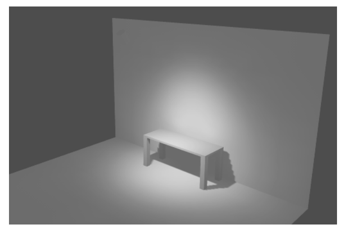


Figure 8-30.

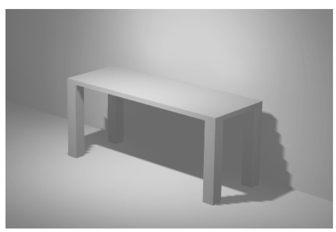


Figure 8-31.

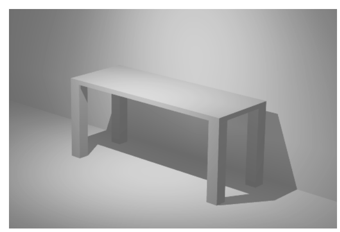


Figure 8-32.

Mach Bands

The mach bands artifact usually appears as a bright line along the edge of two adjacent mesh elements. It usually occurs in areas where the mesh is too sparse and can be eliminated by increasing the density of the mesh.

Streaky Shadows

If a surface is made up of many oddly proportioned surfaces such as long skinny triangles, the mesh generated by the initiation process may also be made up of many oddly proportioned elements. This tends to increase the jagged shadow boundary and shadow leak problems described earlier by making the shadow edges appear streaky.

Again, you can use ray tracing to produce better shadows. If your interest is in a good radiosity solution for interactive manipulation, you may want to try to create the original surfaces from more regularly shaped components during the modeling stage. (You can also do this during the Preparation stage in Lightscape, using the Create Surface option on the Tools menu.)

The optimal shapes are regular, such as squares or equilateral triangles. Figures 8-33 and 8-34 show two examples of surfaces defined from two different configurations of triangles and rectangles. The surfaces on the right would produce better radiosity results than the surfaces on the left.

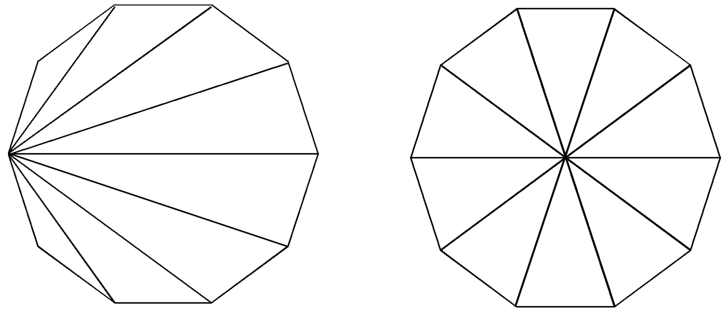


Figure 8-33.

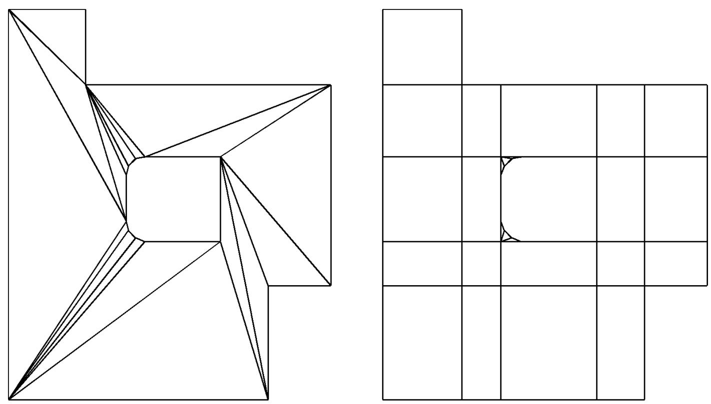


Figure 8-34.

Testing for Artifacts

Typically, you run two radiosity solutions to locate and deal with visual artifacts.

The first solution usually does not have to go past the number of iterations required to process the contribution from the direct light sources, since almost all artifacts are the result of these direct lights. You can use the default meshing parameters as a starting point.

After the radiosity solution has progressed past the processing of the direct light sources, you should interrupt the process and examine the solution for artifacts. If any are visible, you can decide on the best way to handle them and then take the appropriate action.

If you see a shadow leak, for example, one approach would be to set the surfaces casting the shadow to be nonoccluding. Alternatively, you can select the surface with the artifact and increase the mesh subdivision for that surface by changing its Mesh Resolution parameter. You may decide to set the Ray Trace Direct Illumination processing parameter for the light source causing the artifact and not worry about the artifact during the radiosity solution.

How you deal with artifacts also depends on what final output you want. If you are creating a real-time environment or an animation, you want to obtain the best possible result with the radiosity solution. If you will only be generating a limited number of still images, you can ray trace the direct illumination from some or all of the lights to eliminate the artifacts completely.

After you make the desired adjustments, you can reset the solution and run it again. Resetting the solution does not alter any surface operations that may have been done earlier. Be aware that changes to the Solution file are not reflected in the Preparation file. You may want to update the Preparation model to be consistent with the current Solution model.

Chapter 9

Mesh to Texture

You can use Lightscape's Mesh to Texture tool to convert radiosity solutions to texture maps.

Mesh to Texture Wizard

The Mesh to Texture interface is a wizard which takes you through a succession of pages within a dialog that lets you make choices to determine how the mesh to texture conversion will be done.

Converting the radiosity mesh to texture maps can provide several benefits:

- *Reducing complexity*—By transforming polygon meshes into textures, you can reduce the polygon count in your model. This capability is important for improving display speed both in Lightscape and in real-time 3D applications, including interactive games, VRML, and virtual sets.
- *Integration*—If the source model is heavily textured, you can add or incorporate radiosity lighting into these existing texture maps. When the source model is re-opened, the texture maps will contain additional lighting information from the radiosity solution.
- *Panoramic images*—You can easily generate a panoramic image based on a cylindrical or spherical projection for use in specialized model viewers using these projection methods.

Below is an quick outline of steps in the wizard procedure and three simple examples of how you might set the options in the wizard.

If you want to do mesh to texture conversion in a Lightscape model, you should create a copy of the original model before continuing. Once the lighting data has been removed from the radiosity mesh, it will be impossible to interactively update or change the lighting and material properties of the model.

The wizard first provides three general methods to convert polygonal meshes to textures: "Convert each surface to a single texture per surface" creates one texture for every Target surface selected; "Relight existing textures" creates a new version of the textures on the Target surfaces with the radiosity lighting data added; "Project all selected geometry into one texture" creates a single texture from all the Target surfaces selected.

In all three methods, you will next create two selection sets—the first, often empty, defining the surfaces you wish to project onto the surfaces in the second selection set, and the second set defining the Target surfaces upon which the textures will be placed.

Depending on the choices you have made above, you may be prompted to define the orientation and placement of the resulting texture map—using any of Lightscape's texture projection options: orthographic, cylindrical, spherical, and object UV's.

Next, over several pages in the wizard, you will set various options controlling the name and nature of the texture files. You can set the name, image type, size, and directory location of the images as they are created.

You then specify how you wish to render the image—using all of Lightscape's standard rendering options such as ray tracing direct shadows and output of alpha channel information for example.

Finally, you will need to specify what you would like to do with the resulting image (whether or not to place it in the Lightscape model) and with the selected geometry in the Projected and Target selection sets.

Mesh to Texture Examples

The examples below all consider how best to use the wizard with an extremely simple model—a single wall, lit by several lights with a single picture frame hung upon it.

Example 1 : Replace the mesh on the wall

The wall, since it has several lights shining on it and shadows from the picture frame, has been broken into a complex radiosity mesh. To simplify the model you will want to create a single texture to represent the wall and then remove the radiosity mesh from the wall surface. To do this you should:

- select the “Create single texture per surface” option on the Method for Conversion page
- leave the first selection set blank
- select the surfaces of the wall in the second selection set
- choose a name and set a size for the texture on the Texture Output Information page
- set the appropriate rendering options on the Rendering Options page (set Ray-bounces to 0, for example)
- choose “Replace textures on target geometry” and “Reset the mesh on the target geometry” on the final page to use the texture you create, instead of the radiosity mesh, to represent the lighting

Example 2 : Replace an existing texture map in the picture frame with a new lighted texture.

The picture in the frame hanging on the wall is already an image that has been applied as a texture map. You may wish to add the lighting effects from the model to this existing texture. To do this you should:

- select the "Relight existing textures" option on the Method for Conversion page
- leave the first selection set blank
- select the surface of the picture in the second selection set
- choose to use the existing filename on the Use Existing Texture Filenames page
- set the appropriate rendering options on the Rendering Options page (set ray-bounces to 0, for example)
- choose "Replace texture on target geometry", and "Reset the mesh on the target geometry" to use the texture you create, instead of the radiosity mesh, to represent the lighting on the picture.

Example 3 : Create a single texture map of the wall and picture.

The wall and picture together represent quite a bit of geometry and radiosity data. To really simplify the model, you may wish to create a single flat image (the picture applied as a decal to the wall) and delete the picture and its frame from the model. To do this you should:

- select the "Project all selected geometry into one texture" option on the Method for Conversion page
- select the surfaces of the picture and its frame as the first selection set (as you wish to project this geometry as a decal on the texture)
- select the surfaces of the wall in the second selection set
- select an Orthographic alignment and align the texture to the wall on the Projection Method page
- choose a name and set a size for the texture on the Texture Output Information page

- set the appropriate rendering options on the Rendering Options page (set ray-bounces to 0, for example)
- choose "Replace texture on target geometry", and "Reset the mesh on the target geometry" on the final page to use the texture, instead of the radiosity mesh, to represent the lighting
- select the "Delete the projected geometry from the model" option on the Replace/Delete page to delete the picture and picture frame from the model

Note: in this particular example, choosing "Create single texture per surface" will create the same image. A more correct example would be where the wall contained two or more materials (a base moulding for example), and you wished to create a single texture to cover both of them.

The Mesh to Texture wizard settings are remembered from one session to the next as long as you do not exit Lightscape. When you exit Lightscape, the wizard settings return to the default settings.

Select Method for Conversion

There are three general methods available for converting selected geometry to textures. Each of the methods defines a different approach to the conversion.

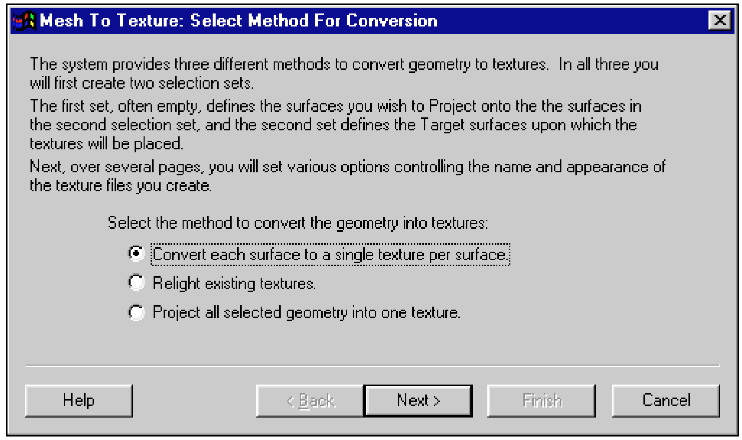


Figure 9-1. *Select Method for Conversion page*

Depending on the nature of the conversion you wish to undertake, you must select one of the following:

Texture per Surface

Select "Convert each surface to a texture per surface" to generate a separate texture for each selected surface. When you select the eight surfaces of a cylinder, eight texture maps (one per surface) will be created. Lightscape determines a separate projection for each surface and creates a corresponding texture. This is the easiest and most automatic way to create textures, but as it creates a texture per surface, you run the risk of creating too many textures.

This method produces an optimal projection that uses as much of the texture area as possible.

Light Textures

Select "Relight existing textures" to use textures and projections you have already defined and generate new textures with the same projections. If you select an eight-sided cylinder with a single texture already wrapped around it, this creates a new version of that texture. This option should only be used when the existing textures are neither tiled nor used on more than one surface in the selection set. In the case where the pixels of a single texture are used only once over many surfaces (the lower left part of the texture used on one surface, the upper right on another), this option will produce the desired effect.

Single Texture

Select "Project all selected geometry into one texture" to create a single texture from all the selected geometry. If you select the eight surfaces of a cylinder, this creates a single texture map. You will later need to select a projection method—Orthographic, Cylindrical, Spherical, or Object UV's and projection coordinates. Lightscape uses these settings to produce a single texture, based on the projection, that represents the mesh on the selected geometry. Although this method is the least automatic, because you are able to group many surfaces together to create a single texture map, it offers the most control and creates the most significant optimization of the model.

Select Projected Geometry

This selection set, the first of two, will generally be empty. This selection contains the surfaces to be projected (as decals) onto the surfaces in the second selection set. Typically, you delete these surfaces from the model after the mesh to texture conversion process to reduce the polygon count.

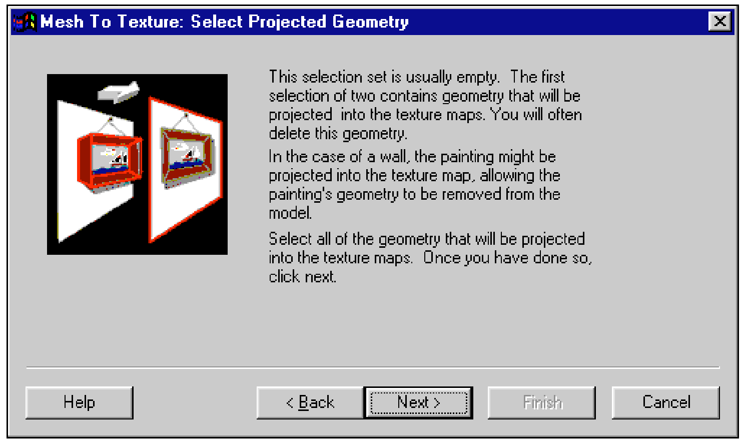


Figure 9-2. *Select Projected Geometry page*

Examples of geometry to select in this step are pictures hanging on a wall and their frames, wall moldings, windows, and any geometry near the wall that doesn't need to be depicted three-dimensionally in the target texture.

When creating a cylindrical or spherical panoramic image, this selection set will contain all the geometry you wish to capture in the panorama. In this case, the second selection set will be empty since the panorama does not need to be applied to any surfaces. Instead, panoramic images can be viewed directly with specialized panoramic viewers.

Select Target Geometry

This selection set, the second of two, contains the Target geometry (the surfaces on which textures will be placed). The lighting information stored in the radiosity mesh of these surfaces, and on any surfaces in the first selection set (Projected geometry), appear in the resulting textures.

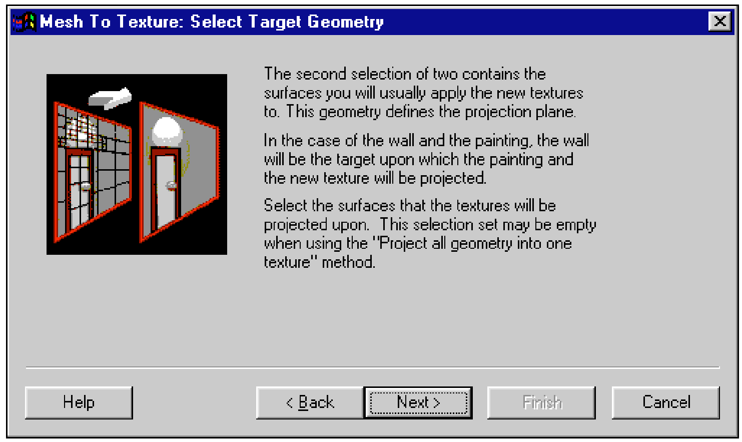


Figure 9-3. *Select Target Geometry page*

If you did not select surfaces in the previous step, you must select at least one surface in this step to continue. If you did select surfaces in the previous step, this selection can be empty.

Typically, the select set is empty when you are creating a panoramic image. In this case, the first selection set is used.

Select Projection Method

If you selected the Single Texture projection method, this page of the wizard appears. You use this page to choose the projection method, coordinates, and related options for the generated texture.

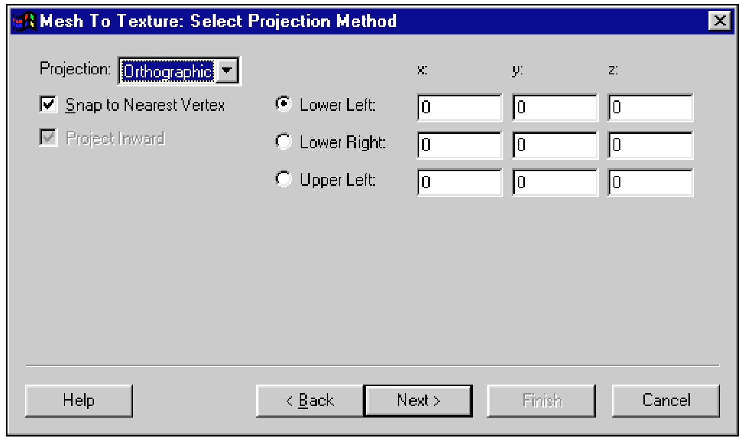


Figure 9-4. *Select Projection Method page*

Projection

Lets you specify the type of projection you want to use. The following projection types are available: Orthographic, Cylindrical, Spherical, and Object UV's.

- In Orthographic mode, you must pick Lower Left, Lower Right, and Upper Left points or type the values you want into the corresponding input fields. These three points determine an orientation and a size for the mapping.

- In Cylindrical mode, you must pick Lower Center and Upper Center points and a Seam Direction or type the values you want into the corresponding input fields. These three points determine an orientation and a size for the cylindrical mapping.
- In Spherical mode, you must pick a Center point, Top Pole, and Seam Direction or type the values you want into the corresponding input fields. These three points determine an orientation and position for the spherical projection.
- No points need to be picked for Object UV's mode.

Snap to Nearest Vertex

When On, selecting a point in the model will select the closest vertex to that point on the same surface.

Project Inward

This option affects the direction from which the texture is projected in the cylindrical and spherical modes. When On, the texture is projected from the outside to the center. When Off, the texture is projected outward, from the center to the outside.

When mapping a texture onto an inside surface, such as when the viewer is standing in the center of a room, turn off this option. When you are looking down at a spherical object (like a ball) turn this option on.

These projection methods are similar to those used to align texture maps. See Chapter 3, "Materials" for more information.

Use Existing Texture Filenames

The settings on this page are of particular importance when you have selected the “Relight existing textures” option, or when some of the surfaces in the selection set already have texture maps applied to them. Otherwise you ensure that the “Use existing texture filenames” option is deselected and proceed to the next step in the wizard.

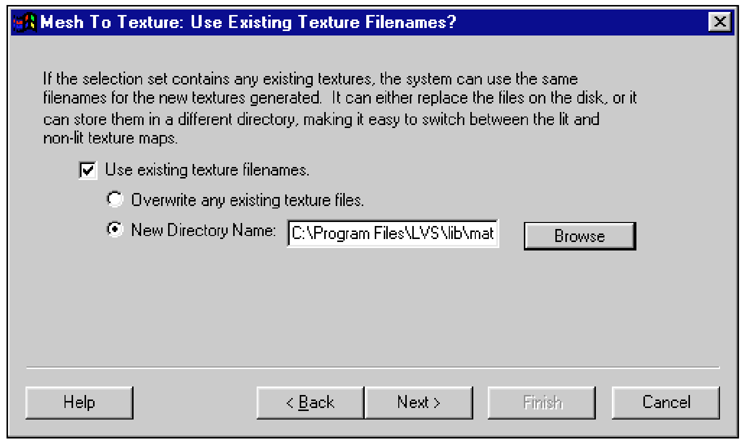


Figure 9-5. *Use Existing Texture Filenames? page*

When you have existing textures, generally you’ll want to save modified texture maps under the same filenames as the originals, but in a different directory to avoid overwriting the existing image files. You can also save the new images over the original files, although this destroys those original image files.

Use existing texture filenames

Select this option to use the existing texture filenames.

Overwrite any existing texture files

Select this option to save textures over the original image files used to create the materials in Lightscape. This is potentially a risky operation, and should be undertaken only if disk space is limited and you have copies of the images saved in another location.

New Directory Name

In general, this is the preferred option when generating texture maps with the same filenames as the originals. Lightscape will save the generated files under the same name or names as the original files, but in a different directory. Enter the new directory name in the text box, or use the Browse button to select a directory.

If using this option, you must specify an existing directory before proceeding.

If the same texture image is applied to more than one surface, Lightscape creates a series of files using the texture map's original filename for each successive file combined with an incremental three-digit number.

Texture Output Information

This page of the wizard lets you tell Lightscape how to save new texture files (their size, image format, and name) that the Mesh to Texture process generates.

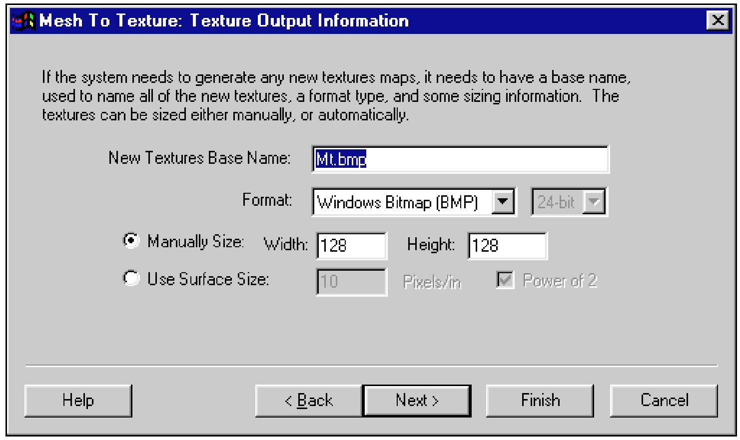


Figure 9-6. *Texture Output Information page*

New Textures Base Name

Enter the base name for the files generated. Lightscape appends three-digit numbers to this name, starting with 000 and incrementing by 1 for each file.

To place the files in a specific directory, enter the path as part of the base name (e.g., c:\textures\test.bmp). If you do not specify a path, Lightscape uses the current model's directory.

You should include the file extension (.bmp for example) or the names will be created without an extension.

Format Type

Selects the file format of the image. The default is the native Windows image format (.bmp)

The file format options are:

- .bmp (native Windows format)
- .tga (Targa, TrueVision format)
- .tif (TIFF)
- .rgb (Native SGI format)
- .jpg (JPEG)
- .gif (CompuServe 8-bit format)
- .png (Portable Net Graphics)
- .eps (Encapsulated Postscript)

With .rgb and .tif formats you can also specify 24 or 48 bit image output.

Sizing Options

These options control the output image size and are available only if you are not relighting textures. If you are relighting textures, the files are generated in the same size as the originals.

Manually Size

Select the Manually Size option to specify horizontal and vertical dimensions in pixels to be used for all generated images.

Use Surface Size

The Use Surface Size option is available only when using the Textures per Surface conversion method.

Select the Use Surface Size option to generate images at a specified number of pixels per the model's current units of measurement. For example, if a selected surface measures 5 x 9 inches, and you specify 8 pixels/in, the resulting texture image will be 40 x 72 pixels in size.

If you select Use Surface Size, select the Power of 2 option to constrain each output image dimension to the smallest power of 2 greater than or equal to its calculated size. In the above example, with Power of 2 selected the output image would measure 64 x 128 pixels.

At this point, you can click Finish to start the Mesh to Texture process using the selected settings.

Because you have not finished choosing all the settings for the Mesh to Texture wizard, clicking Finish now accepts the default settings (or settings from a previous session) for the pages of the wizard that follow (the Rendering Options and the Replace/Delete pages).

Rendering Options

The options on this page control how you render the images—setting the quality and character of the resulting textures.

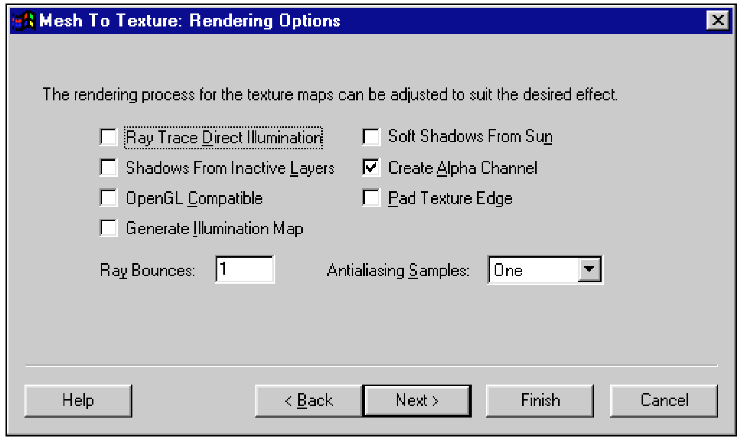


Figure 9-7. *Rendering Options page*

Ray Trace Direct Illumination

When On, the direct energy contribution of the sun and of luminaires that were set to use ray tracing will be recomputed using the ray tracer. This eliminates image artifacts such as the "jaggies" associated with radiosity meshes created by direct contribution of the sun or by individual luminaires, and provides more compelling visual effects.

Shadows from Inactive Layers

When On, shadows from layers that are turned off will be considered when creating an image. This option is used principally when it is necessary to turn off layers to enable a certain view (the floor in a worm's eye view for example), but when the lighting effects of those missing surfaces are important.

OpenGL compatible

When On, the final ray-traced image will more closely resemble the GL rendered image displayed in the Graphic Window. Normally, Lightscape uses more exacting algorithms when creating a ray-traced image to capture subtle lighting effects, but these images often differ to some degree from the image as it is displayed on the screen with GL rendering.

Create Illumination Map

Creates a texture map consisting of only the light striking the surface, instead of the light bouncing off a surface (which is what you normally see).

Soft Shadows from Sun

When On, the edges of the shadows from the sun will be softened, blurring the crisp shadow edge to give a more natural effect.

Create Alpha Channel

When On, generated textures contain an alpha channel based on the cumulative transparency of all model surfaces through which light rays pass. Also, the alpha channel is transparent wherever the background color appears.

Pad Texture Edge

Eliminates potential artifacts around the edges of textures by filling in all the pixels in the texture that do not lie on target geometry with pixels of a similar color. Where there is projected geometry that does not land on the target geometry, the padding will overwrite these areas of the projected geometry.

Ray Bounces

Sets the number of ray bounces used when creating an image. The default is 0.

You may want to set Ray Bounces to 0 to avoid reflections in the texture maps (the reflections don't move with the viewer). However, in the case of panoramic images (the viewer is fixed), you may want to set Ray Bounces higher than 0 to capture reflective effects. You will also need to set Ray Bounces to an appropriate number to get the proper alpha channel when dealing with transparent surfaces (1 per transparent layer).

Ray bounces can be set to any non-negative integer.

Antialiasing Samples

Sets the level of antialiasing. Antialiasing is used to eliminate image artifacts such as the “stepping” or “jaggies” of polygon edges.

At this point you can click Finish to start the Mesh to Texture process. Clicking Finish now, accepts the default (or previously selected) settings for the pages of the wizard that follow (the Replace/Delete page).

Replace/Delete

Once the textures have been created, you are likely to want to use these in the Lightscape model instead of the original radiosity mesh.

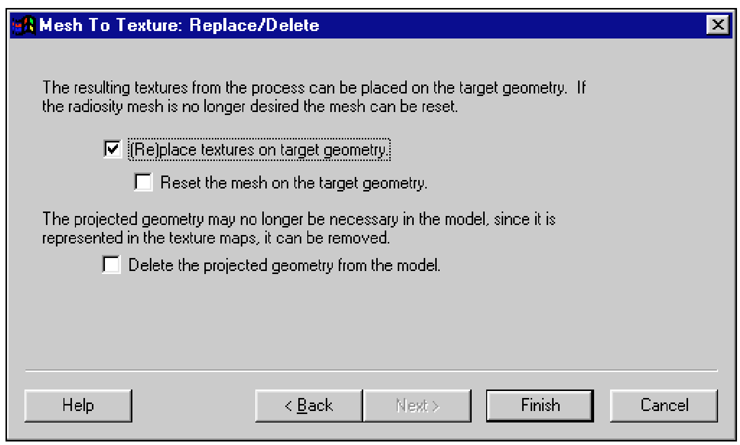


Figure 9-8. *Replace/Delete page*

The options on this page tell Lightscape how to apply the new textures and what to do with the geometry in the two selection sets created above. If you do apply the texture maps in the Lightscape model, you should create a copy of the original model without the texture maps.

(Re)place textures on target geometry

Use this option to apply the new textures to the surfaces you specified as target geometry. With this option, you replace the model's original texture maps and materials with texture mapped materials containing the radiosity solution.

Reset mesh on target geometry

You can reduce the mesh complexity of the selected surfaces by selecting the "Reset the mesh on the target geometry" option. This removes the mesh subdivision created during the radiosity solution, returning those surfaces to their original, pre-radiosity geometry.

Delete the projected geometry from the model

This option lets you further reduce the model's complexity by deleting any surfaces you specified as projected geometry after the new textures are generated. These surfaces and all of their lighting and geometric data will be removed from the model. Once deleted, these surfaces can not be brought back into the model (there is no Undo function for this delete operation).

Chapter 10

Lighting Analysis

You use lighting analysis to evaluate lighting system performance. You can select either luminance or illuminance as the target quantity of the analysis. You can query and visualize the distribution of these quantities for any surface in the environment, including any number of workplanes.

Because Lightscape uses a physically based lighting interface, you can use the radiosity solution as the starting point for a lighting analysis of your model.

The radiosity solution contains all the necessary information for lighting analysis. The options you use for lighting analysis simply filter the lighting data stored in the radiosity mesh to present it in a more appropriate form for evaluating lighting performance.

You can visualize the distribution of light over the surfaces in the environment by using pseudo-coloring techniques or by superimposing a grid of illumination values over selected surfaces. Furthermore, for any surface in the scene, Lightscape can interactively present statistical data such as averages, minima and maxima, and criterion ratings.

As mentioned earlier, lighting analysis presents data stored in the radiosity solution. You can perform an analysis on a complete solution when you want to evaluate both direct and indirect lighting effects. If you want to analyze only the direct lighting, you should run the radiosity solution only for as many iterations as it takes to compute the contribution of the direct light sources to the environment.

This chapter discusses the following aspects of lighting analysis:

- Display of the light distribution
- Statistical analysis
- Analysis grid controls
- Workplanes

To perform a lighting analysis, choose Light > Analysis. The Lighting Analysis dialog box appears.

Display of the Light Distribution

Lighting analysis in Lightscape uses pseudo-coloring techniques to visualize the distribution of light directly onto the surfaces of a 3D scene. You can modify how Lightscape displays this information on the Display page of the Lighting Analysis dialog box.

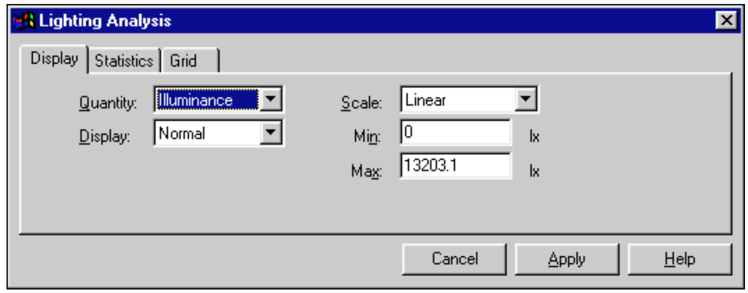


Figure 10-1. *Lighting Analysis Display page*

Quantity

The Quantity parameter specifies which lighting target quantity—luminance or illuminance—Lightscape uses for lighting analysis.

This parameter affects what is shown in the Graphic Window and in all other sections of the Lighting Analysis dialog box.

As discussed in Chapter 2, “The Basics,” of the *Lightscape Visualization System Getting Started* guide, illuminance is a measure of the overall light incident on a surface; luminance is a measure of the light reflected by a surface in a particular direction.

Lighting analysis for the radiosity simulation assumes all surfaces in the scene are ideal diffuse reflectors. Therefore, their luminance is independent of outgoing direction and, as with illuminance, depends only on the location of the illumination on the surface of interest.

Display

The Display parameter controls how Lightscape displays the target quantity (luminance or illuminance) with the model in the Graphic Window. Its possible values are Normal, Color, and Grayscale.

In Normal display mode, Lightscape does not show the target quantity. Instead, it displays the scene as it does when not using lighting analysis.

In Color and Grayscale display modes, Lightscape maps the target quantity to display colors. It then displays the target quantity on the surfaces of the model in place of the normal shading.

When you select either of these modes, the system displays a color chart at the bottom of the Graphic Window to show the correspondence between the illumination values and the displayed colors.

In Color mode, the system maps target quantities to colors ranging from blue to green, yellow, and red. Small illumination values appear closer to blue and large illumination values appear closer to red.

In Grayscale mode, the system maps target quantities from black to white. The larger the value of the target quantity, the brighter the displayed color.

Mapping

The following parameters control the mapping between illumination values and display colors or grayscale levels:

- Scale
- Minimum
- Maximum

Scale

The Scale parameter controls whether the system maps the target quantity to display colors using a linear scale or a logarithmic scale.

By default, Scale is linear.

However, the logarithmic scale can be a good choice when the illumination of the surfaces of interest is low compared to the maximum illumination in the scene. In this case, logarithmic mapping brings out illumination details by reserving more display colors to show differences at lower illumination levels.

Minimum

The Minimum parameter sets the threshold below which the system maps values of the target quantity to the left-most display color or grayscale level of the color chart.

Illumination values smaller than the specified minimum are shown in blue when in Color display mode and in black when in Grayscale display mode.

By default, Minimum is 0.

Maximum

The Maximum parameter sets the threshold above which the system maps values of the target quantity to the right-most display color or grayscale level of the color chart.

Illumination values above this level are shown as red or white, depending on whether the display mode is set to Color or to Grayscale.

By default Maximum is the maximum value of the target quantity in the current radiosity solution.

When most illumination values are clumped in a small subset of the target quantity range, the display may show most of the environment in a single color. To bring out more details, you can use the Minimum and Maximum thresholds to bracket a region of interest.

Statistical Analysis

Lightscape provides statistical data to help you evaluate the performance of a lighting system with respect to a selected surface. It recomputes the data from the radiosity solution every time you select a surface.

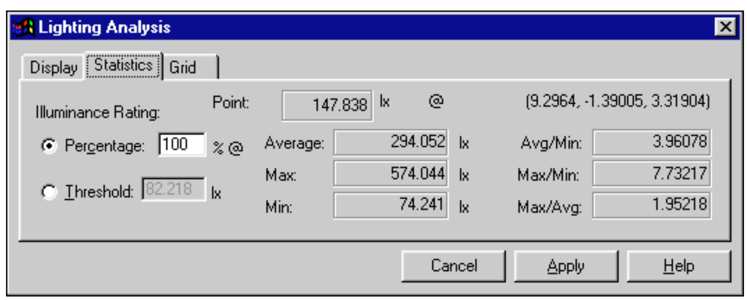


Figure 10-2. *Lighting Analysis Statistics page*

When you select a surface in the Graphic Window, the system shows statistical information for the point on the surface that you clicked. To show information for a different point on the surface, click on that point.

The Point section shows the coordinates of the point you select. This is the point used for the statistical values shown.

Statistics

The Statistics page shows information about the selected point.

- *Point* shows the value of the target quantity at any point on a selected surface.

  Lightscape computes the target quantity during radiosity processing at the vertices of the surface mesh. It then computes the point value at the selected location by interpolating the value of the target quantity at the vertices of the mesh elements containing the selected location.

- *Average* shows the average value of the target quantity over the area of the selected surface.

  The average value is a simple way of characterizing the performance of a lighting system when the shape of the distribution of light over the surface is fairly simple.

- *Maximum* and *Minimum* show the maximum and minimum values of the target quantity over the area of the selected surface.

  You can use these values in conjunction with the average value to describe the uniformity of the distribution of light over the target surface.

- *Avg/Min*, *Max/Min*, and *Max/Avg* are different ratios of the average, minimum, and maximum values. You use them to examine different measures of uniformity.

Illuminance Rating

The illuminance rating is the fraction of the area of a surface that satisfies or exceeds a specific criterion. You can use it in addition to the average, minimum, and maximum values just discussed to gather more information about the distribution of light over a selected surface.

You can either specify the percentage you're interested in and have the system compute the illumination threshold or you can specify the illumination threshold and have the system compute the percentage.

To have Lightscape compute the illumination threshold, select the radio button adjacent to the Percentage parameter input box and enter a percentage (for example, 25). Lightscape computes an illumination threshold based on your input. In this example, the illumination in 25% of the surface area is greater than or equal to the computed threshold.

To have Lightscape compute the percentage, select the radio button adjacent to the Threshold parameter input box and enter an illumination threshold (for example, 13000 lx). Lightscape computes a percentage based on your input. In this example, the illumination in the computed percentage of the surface area is greater than or equal to 13000 lx.

With this technique, you can use the analysis tools to compute the percentage of a surface area whose illumination is equal to or greater than a given value of the target quantity, specified by the Threshold parameter. Conversely, you can use the tools to compute the threshold given the percentage of area of a surface whose illumination is expected to equal or exceed the target quantity. This percentage is specified by the Percentage parameter.

Analysis Grid Controls

The pseudo-color display modes provide an intuitive way of evaluating the distribution of light over a surface. Sometimes, however, a traditional grid of numbers is preferable.

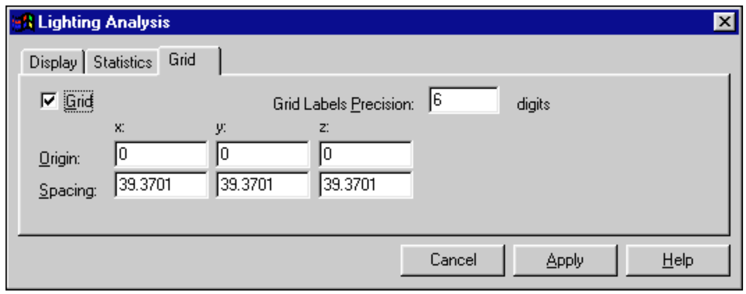


Figure 10-3. *Lighting Analysis Grid page*

To see a grid of numbers for a surface, turn on Grid on the Grid page of the Lighting Analysis dialog box and click Apply. When Grid is on, if you select a surface in the Graphic Window, Lightscape shows a grid of uniformly spaced sample points for that surface.

You do not have to specify the density and orientation of the analysis grid for each selected surface. Instead, Lightscape uses a single three-dimensional reference grid to automatically place an analysis grid on any selected surface in the scene.

You control the grid using the Origin and Spacing parameters.

Lightscape orients the reference grid with the Cartesian axes. The Grid Origin parameter specifies the X, Y, and Z coordinates of its origin. The Grid Spacing parameter specifies the distance between adjacent reference grid nodes along the X, Y, and Z axes.

By default, the grid origin is at (0,0,0).

The Grid Labels Precision parameter controls the number of significant digits the system displays of the target quantity value on the analysis grid.

Workplanes

Most lighting analysis applications allow you to compute illuminance values at evenly spaced points over a *workplane*. The workplane may be one of the surfaces in the scene, or it may be a region on a plane positioned and oriented anywhere in 3D space.

For example, you could use a workplane parallel to the ground and at the height of a typical table to verify that the illuminance levels produced by a proposed lighting system on that plane are within the recommended guidelines for comfortable reading and writing.

In Lightscape, workplanes are just surfaces in the model, set with a particular combination of surface properties. To turn a surface into a workplane, you set its local control parameters (see Chapter 8, “Solution”) so that the surface is a receiver, but neither an occluder nor a reflector.

A workplane must be receiving, so that it can register the incoming illuminance and store it in a radiosity mesh like any other surface in the scene. However, it must not affect the propagation of light through the environment. Therefore, the workplane must not occlude light passing through it and must not reflect light incident on it.

Typically, you place workplanes on one or more distinct layers so you can easily hide them from view during normal display. You do so by simply turning those layers off. Remember that these layers must be turned on during the lighting simulation so that Lightscape can record illumination on these surfaces.

Because workplanes merely act as light sensors and do not reflect incident light, Lightscape shows no luminance values for these surfaces.

Chapter 11

Animation

Lightscape offers a powerful set of tools for creating walkthrough animations. Because the radiosity solution is view independent, individual frames of the animation can be generated much faster than with traditional rendering techniques.

Creating Animations

Lightscape provides a robust set of tools for defining and generating walkthrough animations. You define a camera path by creating a series of keyframes and a path connecting those keyframes. You can also control the speed at which the camera moves along its path. Once you have created the animation path, you can display animations on the screen in the Graphic Window or you can render and save the individual frames of the animation.

Depending on the complexity of the model and the display hardware used, you may be able to see a real-time animation. In other situations, it may be more appropriate to save individual frames and display them using a movie-playing utility.

Generating a walkthrough animation in Lightscape involves the following steps:

1. Define a 3D camera path by setting a sequence of keyframes and, optionally, defining a camera "lookat" path.
2. Define the speed of the camera and its view direction as it travels along this path.
3. Preview the animation using the Preview mode.
4. Adjust the path and speed if necessary.
5. Generate the individual frames and store them on disk.
6. Transfer the frames to a movie file format on disk or export the images to video or film. (You must do this step outside of Lightscape.)

Defining the Camera Path

The first step in creating an animation is to define the path along which the camera travels. Choose Animation > Edit, to open the Animation dialog box, and select the Path page.

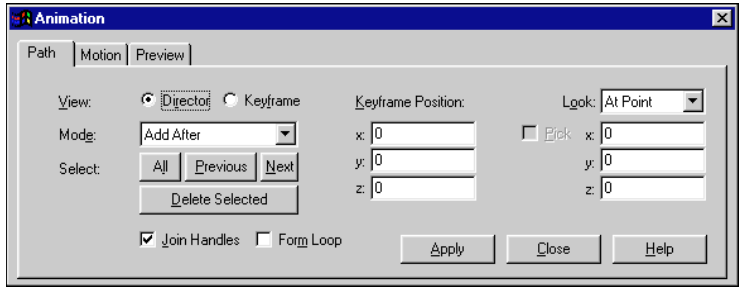


Figure 11-1. *Path page of the Animation dialog box*

You define the camera path by specifying the location of keyframes and then adjusting the path connecting the keyframes.

It is easier and quicker to use Wireframe display mode rather than Solid display mode when setting up the animation path. Also, because the Preparation file (.lp extension) is more compact, it is faster to create the path using the Preparation model. You can save the animation path to a file and reload it with the Solution model to display the animation.

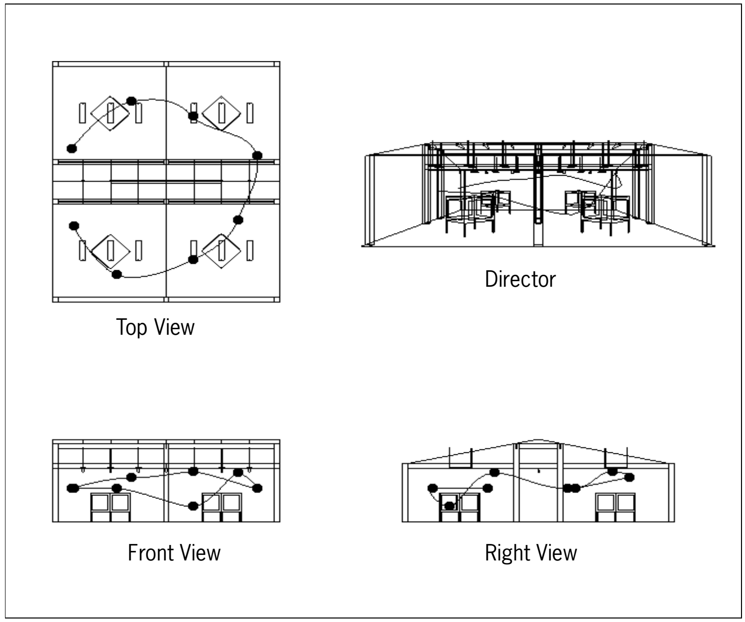


Figure 11-2. *Animation Interfaces*

During camera path editing, Lightscape splits the Graphic Window into four windows—three orthographic views and a perspective view, as shown in Figure 11-2.

The three orthographic windows show top, front, and right side views. When the cursor is in these windows, only the Zoom and Scroll view control options are available.

Setting the Perspective View

The perspective view initially shows the current view of the model and the animation path. This view can be either from the Director's perspective or the keyframe's perspective. This view defaults to the Director's view. The Keyframe view shows a view of the model as seen from the currently selected keyframe.

When you place the cursor in the perspective window and the Director's view is active, all of the view control options such as Orbit and Rotate, are available. When the Keyframe view is active, only the Zoom view control option is normally available.

Creating the Camera Path

You create a camera path by adding keyframe points to a path. You can define and edit the path by positioning the cursor only in the orthographic views, not in the Director's view.

As you define the path, Lightscape displays it in all four windows. Initially each keyframe is denoted by a large red dot which becomes white as you add keyframes. Keyframes are joined by lines representing the camera path.

Before adding keyframes to a path, select either Add After or Add Before in the Control Point section of the Path Editor sidebar menu. In Add After mode, Lightscape adds new keyframes after the last currently selected keyframe in the path. In Add Before mode, it adds new keyframes before the first selected keyframe.

To add a keyframe, pick a point in any of the orthogonal view windows. Lightscape adds a new path segment between the new keyframe and the last (or first) selected keyframe. If the system adds a keyframe between two existing keyframes, either a break or a refinement occurs in the existing path. Once added, the new keyframe becomes the only selected keyframe.

New keyframes take their field of view and their near and far plane distances from those defined by the current View Setup. You can reset these values by choosing View > Setup. All keyframes added after a change to the View Setup use the new values. Lightscape automatically interpolates changes in the field of view between keyframes to produce smooth zooming during the animation sequence.

To move the keyframes and adjust the camera path, select Edit mode in the Control Points section (in the same list as Add Before and Add After). Editing the path is discussed in “Editing the Camera Path” on page 11-12.

Setting Camera Orientation

In addition to setting the path of the camera as it travels through your environment, you also need to set the orientation of the camera as it travels along this path. This is the direction the camera looks.

Lightscape provides a number of options for this setting.

You may find it easier to set the perspective view to Keyframe View while setting the camera orientation. This lets you more easily see the effect of your changes.

When you add a keyframe to the path, Lightscape sets its view to the current keyframe view. You can have the keyframe look in the following directions:

- In Direction
- Along Path
- At Path
- At Point

Once you specify the camera’s path, you can change its direction for individual keyframes in Edit mode. For example, your animation can start out looking along a path and then at a certain keyframe switch to looking at a specific point.

During generation of the animation, Lightscape automatically creates a smooth transition between the different views.

In Direction

The camera looks in a specific direction and/or view tilts for all the selected keyframes. When you set keyframes to look In Direction, Lightscape displays a green arrow showing the view direction (Figure 11-3).

To set the direction, first set the Perspective window to the Keyframe View option. Then, rotate and/or tilt the view to the desired direction. As you rotate the view, the system updates the green arrow to reflect the new view direction.

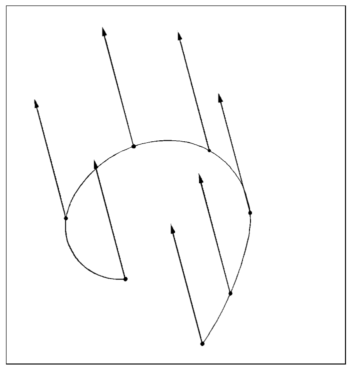


Figure 11-3.

Along Path

The camera looks in the direction of motion. When you set keyframes to look Along Path, Lightscape displays a green arrow showing the view direction. This view direction is tangent to the camera path and in the direction of the camera motion.

Along Path is the default view setting (Figure 11-4).

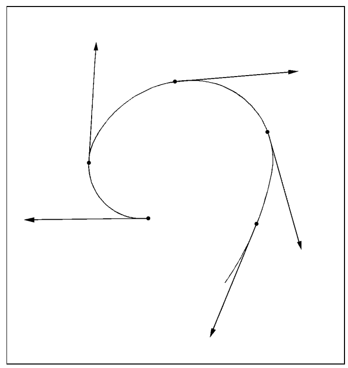


Figure 11-4.

At Path

The camera looks at a point on the camera path where the camera will be at the specified time offset.

You specify the time offset in the dt input box. If you select a negative time offset, the camera looks backward along the path (Figure 11-5).

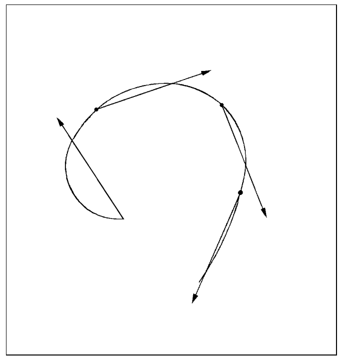


Figure 11-5.

At Point

The camera looks at the specified point for the selected keyframes (Figure 11-6). When you select the At Point option, Lightscape places a focus point at location 0, 0, 0 and draws a bright green line between the selected keyframes and the new point. After the point is placed, you can enter new coordinates for it in the input fields, or use the Pick option to select a location in any of the four views in the Graphic Window.

Lightscape also draws darker green lines between the focus point and the neighboring camera keyframes and/or focus points. This darker green line represents a “look-at path,” and is always visible. (The bright green line is visible only when the associated camera keyframe is selected.)

As with the camera path, Lightscape uses straight-line segments by default between the look-at path’s keyframes. You can edit the path in the same way as a camera path, moving keyframes and changing spline values interactively. (Editing the path is discussed in “Editing the Camera Path” on page 11-12.) You can also edit the speed at which the camera’s aim moves along the lookat path with the Motion Editor (“Defining the Camera Speed” on page 11-15.)

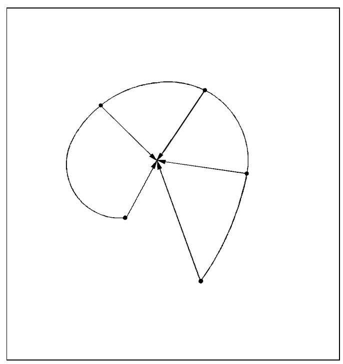


Figure 11-6.

Composite Path Views

You can assign different view types to each keyframe point on a path to establish a composite view path (Figure 11-7). Lightscape tries to smoothly interpolate the camera orientation based on the camera orientation you provide. If the camera orientation is very different between sequential keyframes, the view may not change smoothly. To fix this, you can add more keyframes between the existing ones.

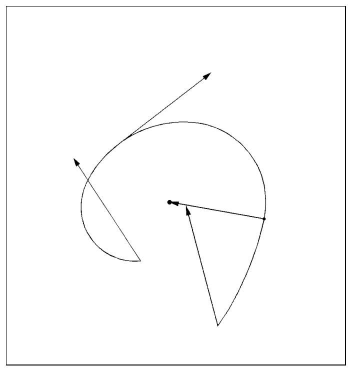


Figure 11-7.

Changing Field of View of Keyframes

In addition to moving the direction the camera looks at each keyframe, you can also change its field of view (zoom) as it moves along the path.

To set the field of view, set the perspective window to keyframe view and zoom in or out. To more accurately set the field of view, choose View > Setup while a keyframe is selected. Lightscape smoothly (linearly) interpolates changes in the field of view.

Editing the Camera Path

To edit keyframes, choose Edit in the Control Point area of the Path Editor sidebar menu. In this mode, you can change the selected keyframes and the camera path in the following ways:

- Move keyframes
- Change the slope of the camera path
- Join or break keyframe handles
- Delete keyframes
- Make a Loop

You can select keyframes in the following ways:

- Select all keyframes with the Select All button.
- Select one keyframe by clicking in the Graphic Window.
- Select a group of keyframes by dragging a box around them.
- Hold the **`Shift`** key while clicking or dragging to add keyframes to the selection set.
- Hold the **`Ctrl`** key while clicking or sweeping, to toggle the selection of affected keyframes.

Move Keyframes

You can drag selected keyframes to the desired position in any orthographic view. If more than one keyframe is selected, dragging one of them drags the entire group.

You can also explicitly position a keyframe by entering X, Y, and Z coordinates in the input field.

Be careful when you move individual keyframes. If you click on a keyframe then drag, instead of moving the keyframe, you'll change the slope of the camera path (as described in the next section). To move a keyframe, click and hold the mouse button while dragging.

Change the Slope of the Camera Path

You can change the shape of the camera path curve by adjusting the handles of a keyframe. You can do this to only one keyframe at a time.

The handles are dumbbell-shaped objects displayed tangent to the camera path at the keyframe location. Initially the handle ends are coincident with the keyframe.

To move a handle, you select a single keyframe and then drag its handle. You must first select the keyframe and let go. Then you can drag its handle. (If you don't let go first, Lightscape assumes you want to move the keyframe.)

The handle direction defines the tangent of the curve at a keyframe. The length of the handle's arms defines the shape of the camera path. Lengthening and shortening the arms changes the path's curvature. The curve is "attracted" to the handles—moving a handle away from its keyframe makes the curve stretch toward the handle (Figure 11-8).

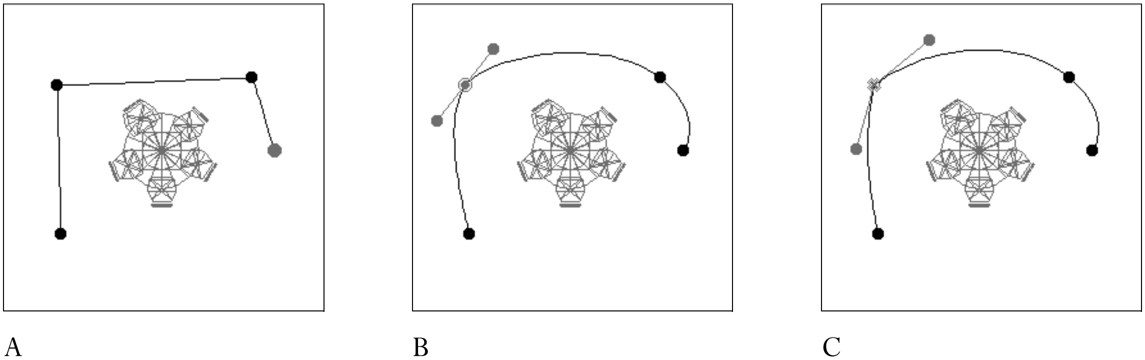


Figure 11-8. *Keyframe Handles*

Join or Break Keyframe Handles

You can introduce a discontinuity in direction into the animation path by breaking a keyframe's handle. You do so by turning off the Join Handles option. If a keyframe is already broken, turning on the Join Handles option will joining the handles again.

Once the handle is broken, you can adjust each side of it independently. This is shown in part C of Figure 11-8. You can use this operation to introduce a sudden change of direction in the walkthrough path. You cannot introduce a discontinuity of position (the camera cannot "jump" to another location).

When a keyframe's handle is broken, Lightscape displays it as a cross instead of a dot.

By default, a new keyframe has a joined handle.

Delete Keyframes

Click Delete Selected to delete selected keyframes from the path.

Make a Loop

You can form the camera path into a continuous loop by turning on the Form Loop option. Turning on this option connects the first and last points in the camera path spline. Turning off the Form Loop option returns the path to its original form.

Defining the Camera Speed

After you set the path and/or aim of the camera, you use the Motion Editor to set the speed at which the camera travels along its path. To access the Motion Editor, select Animation > Edit, and select the Motion page.

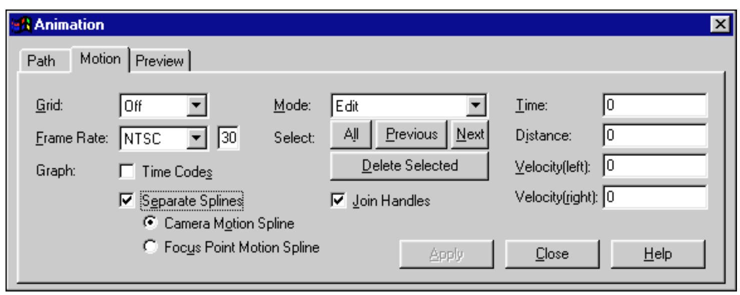


Figure 11-9. *Motion page of the Animation dialog box*

The Motion Editor switches the Graphic Window to display the three views shown in Figure 11-10.

- The Distance Graph at the bottom of the Graphic Window is an editable graph of the camera speed or look-at point.
- The camera view at the top left of the Graphic Window shows the camera's view at a particular time during the animation.
- The Director's view at the top right of the Graphic Window shows the director's view of the model.

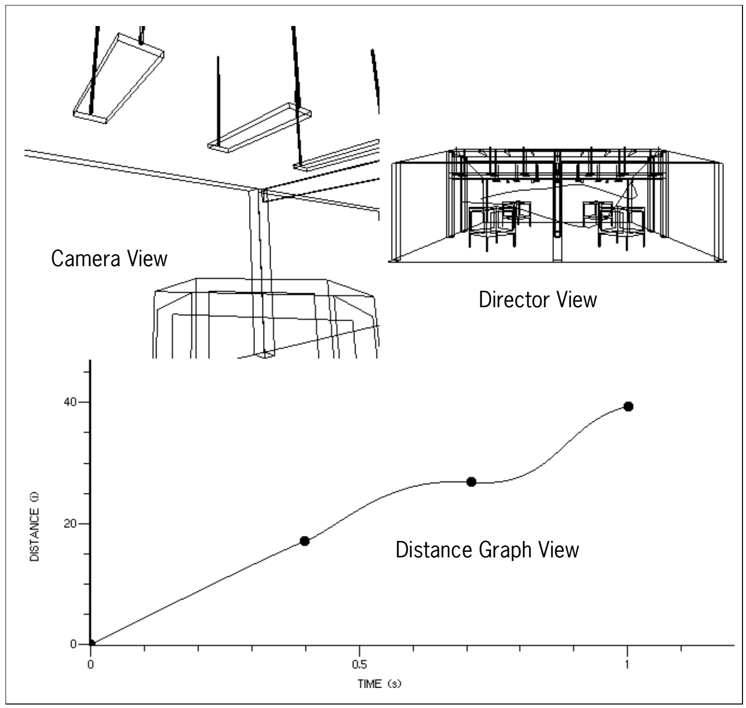


Figure 11-10. *Motion Interface*

Distance Graph

You do most of the work to change camera motion in the Distance Graph. This is an editable graph of the camera speed. You can zoom and scroll this graph using standard view controls.

The vertical axis represents the distance along the camera path or the lookat path. The horizontal axis represents time.

The legends indicate the units of the graph. These are the same units as those chosen for the model as a whole. You can alter these units on the Units page of the Document Properties dialog box.

You can change the time axis on the graph to display time codes based on the frame rate by selecting the Time Codes option.

You can edit the camera and focus point motion separately by turning on the Separate Splines option. When this option is on, you can set the graph to display the motion of either the camera or the focus point motion by selecting the corresponding radio button.

The vertical green line in the Distance Graph indicates the current time shown in the Preview page of the dialog box. You can change the current time by clicking in the time legend at the bottom of the Distance Graph.

By setting the Grid mode to on, you can display a colored grid on the graph.

The horizontal grid lines represent the location of keyframes. That is, each horizontal band between grid lines represents how the camera moves going from one keyframe to the next. If you have multiple keyframes and you do not see any horizontal lines, you need to zoom in the display to show them.

The vertical grid lines represent the location of each frame of animation. That is, each vertical band between grid lines represents how the camera moves between one animation frame and the next.

Because of the large number of frames, Lightscape displays the vertical lines of the graph only when you are zoomed in close enough so that grid lines don't overlap on the display.

When you set the Grid mode to Snap, control points in the motion graph snap to the nearest points on the grid lines, even if the lines aren't displayed.

By default, the camera or focus point speed is set to a constant velocity of 1 meter per second. The default graph displays this constant slope. (If your model uses units other than meters, the graph looks different, but is the equivalent of 1 meter per second.)

Frame Rate

You can use the Frame Rate control to set the number of frames of animation to be rendered by the animation system. The number shown represents the number of frames generated for each second of animation.

Typically, you select from the standard rates. The North American video standard (NTSC) is 30 frames per second. This is the default setting.

Editing Camera Speed

To define the camera or focus point speed, you first define control points on the camera speed curve. Once inserted, you can move and edit these control points to change the curve.

Where the slope of the curve is steep, the camera or focus point moves quickly. Where the slope is gradual, the camera or focus point moves slowly. Where the slope is horizontal, the camera or focus point remains stationary. Where the slope is negative, the camera or focus point moves backward along its path.

Figure 11-11 shows an example of camera motion that slows to a stop and immediately starts moving again, then slows to a stop at a new location, waits for a few seconds, speeds up, and then continues along a path at constant velocity. Four additional control points were added to the graph to accomplish this. Notice how the shapes of the graph represent various types of motion.

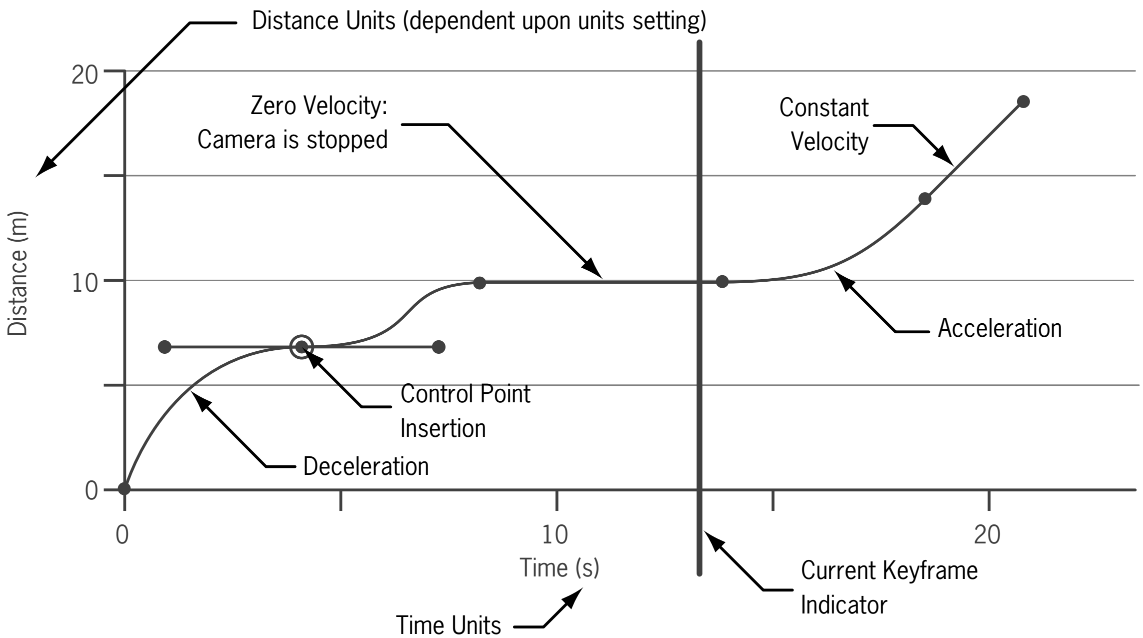


Figure 11-11. *Distance Graph*

In Insert mode, you can add control points to the graph by clicking locations in the Distance Graph. When you insert a control point, it becomes the selected control point. Lightscape shows the time, distance, and velocity coordinates of the selected control point in the sidebar menu.

You can modify the settings of the selected control point by switching to Edit mode. In Edit mode, you can change the selected control points and the camera or focus point speed in the following ways:

- Move control points
- Change the slope
- Join or break control point handles
- Delete control points

You have the same options for selecting control points as you do for selecting keyframes in the Path Editor. For more information, see “Editing the Camera Path” on page 11-12.

Move Control Points

You can move the location of a control point in the graph either by dragging the point in the Distance Graph or by directly specifying Time and Distance values.

The first control point of the graph is always fixed at time zero. You can control its distance along the path.

The position of the neighbors of a control point constrain its position. You cannot move a control point to a time before the previous control point or after the next control point.

You can drag selected control points to the required position. For example, if you know you want your animation to be exactly 30 seconds long, regardless of the speed or distance, drag the last control point to 30 seconds. You can also select the last control point and enter 30 in the Time field.

Change the Slope

The slope of the graph at any point represents the camera or focus point speed at that point. You can control this speed either by entering numbers in the Velocity field or by manipulating the handles of the control point in the Distance Graph.

The Velocity field can have either one or two associated input boxes. If the handles of a control point are joined (see the next section), then there is only one input box. If the handles are broken, there are two input boxes, labelled Left and Right.

The handles are dumbbell-shaped objects displayed tangent to the graph at the control point location. For information on manipulating the handles, see “Change the Slope of the Camera Path” on page 11-13.

Unlike with keyframes, the neighbors of a control point constrain its handles. You cannot move the handle beyond the handles of adjacent control points.

Join or Break Control Point Handles

As with the camera path, you can introduce discontinuities into a motion spline by breaking the handle of a control point. You cannot introduce a discontinuity of distance (the camera cannot “jump” to another position on the spline).

Delete Control Points

Click Delete Selected to delete selected control points from the path.

Previewing the Animation

Once you have set the path and motion, you can preview the animation by choosing Animation > Edit and selecting the Preview page.

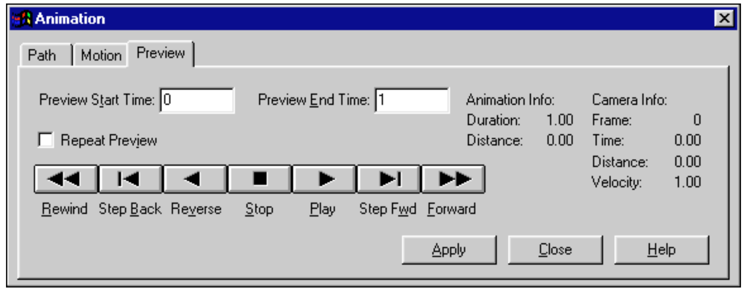


Figure 11-12. *Preview page of the Animation dialog box*

The preview runs in real time. If the computer cannot draw frames fast enough, it skips some frames. This can cause the preview to appear disjointed.

You may want to use Wire mode and the Display Speed option to obtain faster display rates and a smoother result. If there are layers that contain entities not displayed during the animation, you can turn them off.

If necessary, you can more accurately preview an animation by outputting low-resolution test frames to disk and then using a movie utility to play them back.

For Real-Time Previewing mode, you can enter the portion of the animation you want to preview by entering the start time and end time in the appropriate fields. If you type **-1** in the End Time field, Lightscape sets the end time to the end of the animation.

Selecting the Repeat Preview option causes the animation to loop continuously between the start and end times.

The following options control how Lightscape displays the animation preview:

Option	**Description**
Play	The animation plays in real time, from the start of the frame sequence to its end.
Reverse	The animation plays backward in real time, from the end of the frame sequence to the beginning.
Stop	Interrupts a playback.
Rewind	Returns to the first frame.
Forward	Advances to the final frame.
Step Forward	Advances a single frame when clicked; auto repeats when held down.
Step Backward	Reverses a single frame when clicked; auto repeats when held down.

The Animation File

Once you have created an animation path, you should save it to an Animation path file (.la). You cannot save the animation in a Solution or Preparation file.

You can use an Animation file with any model. You can divide large models into smaller models for radiosity processing and then merge them. You may want to generate an animation that moves continuously between these spaces. This involves preparation and testing to set up a single continuous path and then determining at which keyframes to merge or unload the various solutions.

Setting Up an Animation Among Multiple Solutions

First, create the animation file. To do this you load the first model in the group and define the path and speed for the camera as it moves through this model, then save the animation.

Load each subsequent model (overwrite) and the same animation file and continue to define the path and speed for the camera. Be sure to save the animation path before loading each model.

Next, determine at which frames you need to load or unload a particular model during the actual frame-generation process. You can only determine this by using Preview mode to locate the frame at which that model comes into view. For greater efficiency, you should also determine the frame where the current model goes out of view.

Finally, during the frame creation process you load the required files in the Rendering dialog box, then enter the predetermined range of frames in the appropriate fields.

In the current version of Lightscape, there is no interface for loading or unloading files at specific frames. You can do this, however, using a batch file and the batch rendering programs—**lsrender** or **lsray**. An example for setting up this procedure is provided in Appendix C, “Lightscape Utilities.”

Outputting Individual Frames

Once you have defined an Animation file, you need to output the individual frames to disk so that you can transfer them to video or film. You can do this process either interactively in Lightscape or by using one of its batch rendering programs—**lsrender** or **lsray**. These options are discussed in Appendix C.

Chapter 12

Rendering

This chapter covers the techniques and options for generating images in Lightscape.

Creating Images

Rendering is the process of taking the three-dimensional radiosity solution and converting it to a two-dimensional image. With Lightscape, you can perform this operation in two ways—using OpenGL display techniques or using the Lightscape ray tracer.

OpenGL display offers the fastest rendering of a radiosity solution and can be accelerated with an OpenGL-compliant graphic card. OpenGL rendering can display only the lighting effects calculated during the radiosity process.

Ray tracing can produce higher-quality images with reflections and more accurate shadows. See Appendix D, "Reflection Models," for an explanation of the various reflection models.

Lightscape also provides two batch utilities for rendering. **lsrender** renders images using OpenGL and **lsray** renders images using the ray tracer. You typically use batch shell scripts and the batch utilities for large rendering jobs. You can do initial tests for these images at lower resolution in Lightscape. See Appendix C, "Lightscape Utilities," for information on using the batch rendering utilities.

All Lightscape renderers can create a single image, a set of images from a set of View files, or a series of animation frames based on an Animation file. The *Lightscape Visualization System Getting Started* guide discusses views. Chapter 11, "Animation," discusses creating animations. The resulting images can be of any resolution. They can also be antialiased to produce higher-quality output.

This chapter deals with output of 2D images only. You can also export the Lightscape Solution Model, complete with color information at the vertices, into the Inventor, PRISMS, and VRML 1.0c format using the utility programs described in Appendix C, "Lightscape Utilities."

As described in Chapter 1, “The Lightscape Model,” radiosity and ray tracing algorithms are complementary to each other. The radiosity algorithm provides accurate computation of diffuse interreflections, but cannot deal with specular and transparency effects. Ray tracing is capable of modeling these effects, resulting in more accurate images of higher visual quality.

Another important distinction between radiosity and ray tracing is the computation time. Once the system processes the radiosity solution, it can rapidly display a model, often at interactive rates and from different points of view using OpenGL. Thus radiosity is *view independent*. Ray tracing, on the other hand, has to be recalculated for each different view point, which makes it *view dependent*.

Another distinction to consider is that OpenGL rendering occurs at the color depth of your display device. This color depth may be less than 24 bits per pixel, reducing the quality of your output. The **lsray** utility runs in software only and does not depend on your display hardware. Therefore, it can always output images with 24-bit color per pixel. Lightscape can also output images with 48-bit color per pixel in the TIFF and RGB file formats.

Note that even though ray tracing can add a considerable amount of time to the generation of an image, ray tracing in Lightscape generally proceeds faster than traditional ray tracing programs as it makes use of the direct and indirect illumination values already calculated in the radiosity solution model.

Rendering Options

Lightscape can produce images of any resolution that can be output in a variety of standard file formats. The following options control the rendering process:

- Image Resolution
- Image Output Format
- Antialiasing
- Ray Tracing
- Frame Generation (Image View Control)

To access the rendering options, choose File > Render to open the Rendering dialog box.

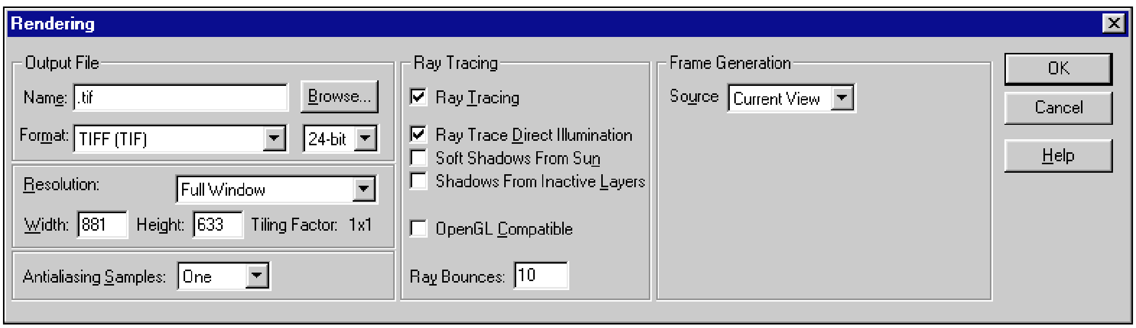


Figure 12-1. *Rendering dialog box*

Image Resolution

Lightscape provides a wide variety of commonly used resolutions. You can also specify a custom resolution. When you specify the resolution, the Graphic Window resizes itself accordingly.

Keep in mind that when the window is resized, the aspect ratio (proportion) of the window may change and the view will be altered. If you intend to set up a number of views for rendering, you should resize the Graphic Window by selecting the desired resolution, before setting your views, so that you can see exactly what you are rendering.

To take advantage of accelerated OpenGL display capabilities, the image has to fit within the bounds of the Graphic Window. Lightscape breaks into tiles images that have higher resolution than the window's dimensions. Each tile is the maximum size that fits within the window while maintaining the original aspect ratio of the image. For example, if you want to create an 1800 X 1200 image (larger than the maximum Graphic Window size), Lightscape breaks up and displays the image as four tiles of 900 X 600 pixels each. Once it has generated the image for every tile, Lightscape creates the final high-resolution image by combining these tiles.

Image Output Format

Lightscape currently supports the following image output formats:

- .bmp (Windows native file format)
- .tga (Targa, TrueVision format)
- .tif (TIFF—24-bit and 48-bit)
- .rgb (RGB—24-bit and 48-bit, native Silicon Graphics file format)
- .jpg (JPEG)
- .gif (CompuServe Graphics Interchange format)
- .png (Portable Net Graphics)
- .eps (Encapsulated PostScript)

Antialiasing

For better quality, images should be antialiased. This smooths out jagged edges and gives better results when the model contains features smaller than a single pixel. Lightscape offers ten levels of antialiasing.

Lightscape uses different antialiasing algorithms depending on whether ray tracing is off or on. If ray tracing is turned off, OpenGL image generation makes use of either a software or (if available) a hardware accumulation buffer. It jitters the original images a number of times by a slight subpixel amount. It then averages these images to produce a single high-quality image.

This process requires the image to be displayed *n* times, where *n* is the square of the antialiasing level selected. For example, an antialiasing level of 2 displays the image 4 times to create the final image. A level of 10 displays the image 100 times (taking 100 times longer to create than the non-antialiased image).

The ray tracer uses a different adaptive scheme that is more efficient for this process. There is no direct correlation (as with the OpenGL method) between the antialiasing level and the time required.

A single still image requires antialiasing for high quality, but the level does not generally need to be as high as it does for animation frames. It is much easier to see aliasing in animations, particularly if the model contains many thin (less than a pixel) features, such as cables or railings. While you may obtain satisfactory single images with a level of 3 or 4, animation frames may require a level of 6 or 7.

Ray Tracing

The following options are available when using ray tracing:

- Ray Trace Direct Illumination
- Soft Shadows from Sun
- Shadows from Inactive Layers
- OpenGL Compatible
- Ray Bounces

Ray Trace Direct Illumination

The Lightscape ray tracer can proceed at a much faster rate than traditional ray tracers because it does not require the shadow rays normally needed to calculate the light contributions from each light source. You can, however, use the Ray Trace Direct Illumination option to ray trace the direct contribution from the sun and specified luminaires. Lightscape ray traces only luminaires that have their Ray Trace Direct Illumination property set on. (See Chapter 7, “Lighting.”)

Use this option to correct shadow aliasing problems. (See Chapter 8, “Solution.”) In addition to providing better shadows, ray tracing a light source provides additional enhanced lighting effects, such as highlights on nondiffuse surfaces. (See Appendix D, “Reflection Models.”) Remember that the image generation time can increase significantly with the number of light sources that are ray traced.

Soft Shadows From Sun

By default, Lightscape renders shadow boundaries caused by the sun as sharp edges. The Soft Shadows From Sun option blurs the edges to provide a more realistic and natural shadow boundary. This option can significantly increase the rendering time of an image.

Shadows From Inactive Layers

This option causes objects on layers that are not active (not visible) to cast shadows. The objects don’t appear in the image, but their shadows do appear. You can use this feature when converting radiosity data into texture maps.

OpenGL Compatible

As discussed in Appendix D, “Reflection Models,” OpenGL and the Lightscape ray tracer use different reflection models, with the result that images created from a Solution model using OpenGL don’t look the same as images generated with the ray tracer.

The OpenGL Compatible option provides a compatibility mode for the ray tracer. It forces the ray tracer to generate images that closely match the OpenGL display. It goes a step beyond the OpenGL reflection model by adding specular reflections, but does not render them to as high a quality as it can when this option is not used. For more information, see Appendix D.

Ray Bounces

The Ray Bounces option controls how many levels of reflection or transmission Lightscape calculates during ray tracing by controlling the number of ray bounces tracked.

For example, if you have two windows you want to see through, set this parameter to at least 2. (Keep in mind, however, that if you actually model the panes of glass with two surfaces, you set the parameter to 4.)

If regions of the image that contain transparent objects look incorrect, increase the number of ray bounces. The ray bounces only need to be small when there are many transparent or reflective surfaces and the rendering time is too slow.

If the number of bounces is set to 0, you will see no specular or transparency effects. The default value for this parameter is 10.

Ray Trace Area

During the Solution stage you can drag a rectangle over an area of the Graphic Window, and then ray trace only that area using the Ray Trace Area option in the Display menu. The results of the ray trace will be displayed to the screen.

You can use this option to test the results of ray trace settings on a selected area before you ray trace your entire model, or to test the effect of material properties changes in the current model. You cannot save the results of the Ray Trace Area operation. The results will remain on the screen until the model is redrawn.

Frame Generation

Lightscape has three options for controlling the view and the number of images it creates:

- Current View
- List of Views
- Animation file

These options are available for both OpenGL rendering and the ray tracer. They are also available in the batch rendering utilities. (See Appendix C, “Lightscape Utilities.”)

Current View

The Current View option creates a single image using the current view.

List of Views

You can add saved View files to the View list. Lightscape creates one image for each view in the View list. To create the filename, Lightscape concatenates the output filename specified in the list with the prefix of the view name.

The system adds the appropriate extension based on the image output type. For example, if you select .bmp output, enter **`set1`** in the output filename field, and there are three view files loaded: `v1.vw`, `v2.vw`, and `v3.vw`, then Lightscape names the resulting images `set1v1.bmp`, `set1v2.bmp`, and `set1v3.bmp`. If you want to preserve the DOS 8.3 naming convention, you need to be sure that the image name, combined with the longest view file prefix, does not exceed eight characters.

Animation File

An Animation file represents a precalculated list of views generated by the Lightscape Animation tool described in Chapter 11, "Animation." When you select this option, Lightscape generates a single image for each animation frame. To create the filename, Lightscape concatenates the output file name specified in the list with the frame number, using four digits for numbering.

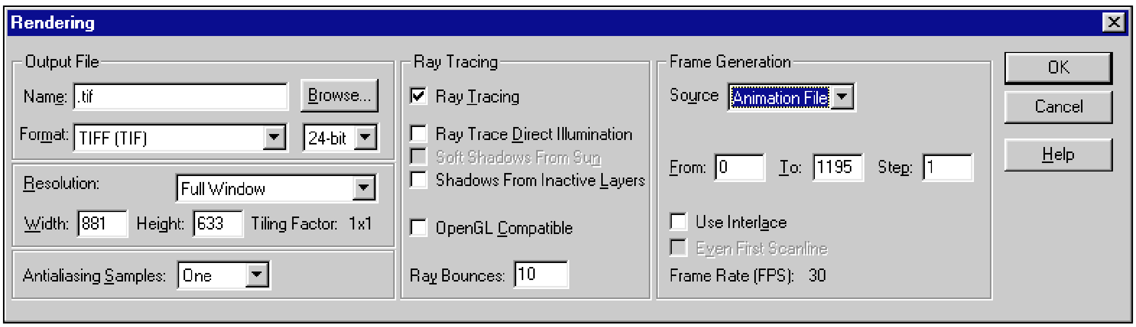


Figure 12-2. *Animation selections on the Rendering dialog box*

For example, if you enter **`anim`** as the filename and choose the Targa format, Lightscape names the resulting images `anim0000.tga`, `anim0001.tga`, and so on. If you want to maintain the DOS 8.3 naming conventions, you must specify an output filename with no greater than four characters.

You do not have to render all frames in the animation. You can specify the first and the last frame to render, as well as the step between consecutive frames. This lets you test the animation path and rendering options.

When you create animations for video, they are *interlaced*. Interlacing is used so that only half the screen, every other scan line, is updated each sixtieth of a second (NTSC frame rate). A set of every other scan line is called a *field;* two fields make up a *frame*. Without interlacing, 60 images per second must be transmitted and displayed by a video signal. With interlacing, this can be reduced to 30 interlaced frames.

To render interlaced frames, select the Interlacing option. You can also choose to have the first field contain either odd or even scan lines. Antialiasing takes interlacing into account in order to generate higher-quality animations.

Appendix A

Import Filters Specifications

This appendix provides detailed specifications for the Lightscape translation modules, including the command line syntax for the stand-alone translators, a description of the options in the import dialog boxes, and a brief description of the file formats themselves.

Importing DXF Files

The DXF file format was designed by Autodesk and is now a de facto AEC industry standard for the exchange of geometric data. Most commercial CAD and modeling applications can export to DXF files.

Lightscape currently imports most of the DXF entities that can be converted to polygons in 3D. The complete list is as follows:

- 2D Polyline
- 3D Polyline
- 3D Face
- 3D Polygon Mesh
- 3D Polyface Mesh
- Arc
- Circle
- Line
- Solid
- Trace

Import Parameters

You control the conversion of DXF entities with the following parameters:

- Block Creation
- Capping
- Smoothing
- Smoothing Angle
- Number of Arc Segments

Block Creation

Some CAD systems export their data to DXF files as flat hierarchies with no blocks and instances. In this case, the Block Creation parameter offers a way to group imported entities into blocks according to one of five criteria:

- As Is—Every entity is read as it exists in the DXF file. Lightscape does not create any additional blocks.
- As One Block—All top-level entities (entities not already included in a block) are grouped in a single block. You specify the block name using the Name parameter. Lightscape creates one instance of the block and inserts it into the model.
- By Color Index—Top-level entities are grouped according to their DXF color index. Lightscape creates one block for each color index that is referenced by at least one entity. The block name is COLOR*dd*, where *dd* is the index. If the index is one digit, it is preceded by the digit 0; if the index is larger than 99, it contains as many digits as necessary. Lightscape creates one instance for each block and inserts them into the model.
- By Layer—Top-level entities are grouped according to their DXF layer. Lightscape creates one block for each layer that is referenced by at least one entity. The block name is the same as the corresponding layer name. Lightscape creates one instance for each block and inserts them into the model.

- By Entity—One block (and corresponding instance) is created for each entity. This does not mean that each block contains exactly one primitive. During loading, the system converts each entity to one or more polygons. For example, if Lightscape reads a polyline entity and converts it to 100 polygons, there is one block containing the 100 polygons, not 100 blocks.

  The block name is *PREFIXdd*, where *PREFIX* is the name of the entity in uppercase letters—for example, CIRCLE or LINE—and *dd* is a unique number assigned to each entity, starting from 1 and incrementing as each new entity is imported.

Lightscape also maintains and takes advantage of the layering and block structure of the DXF format. Both of these structures are supported by the Lightscape Preparation format and can play a significant role in managing data in the system. Blocks are discussed in Chapter 6, “Blocks and Luminaires,” and layers are discussed in Chapter 4, “Layers.”

Capping

The Capping parameter controls how the system converts circles and closed polylines with no width.

If the entities have thickness, this parameter controls whether the resulting extruded objects are capped—that is, whether they have surfaces closing the top and the base.

If the entities have no thickness, this parameter controls whether Lightscape treats the entities as a simple line in space or as the boundary of a surface—for example, a disc in the case of the Circle entity. The system discards lines; it breaks down surfaces into triangles and quadrilaterals during the conversion.

Smoothing

The Smoothing parameter controls whether Lightscape converts thick 2D polylines and 3D polygon and polyface meshes to quadrilaterals with vertex normals. When a curved surface is approximated by a polyhedron, the system can use vertex normals during the lighting simulation and rendering to provide a smooth appearance across polygon boundaries.

Lightscape automatically generates vertex normals for extruded arcs and circles, as well as for 3D polygon meshes with a smooth surface type, regardless of the value of the Smoothing parameter.

Smoothing Angle

The Smoothing Angle parameter allows sharp edges to be preserved even when the Smoothing parameter is on. If the angle between the normals of the faces incident on a vertex is larger than the value of the Smoothing Angle parameter, a vertex normal is not computed for this vertex and the sharp edges are preserved (Figure A-1).

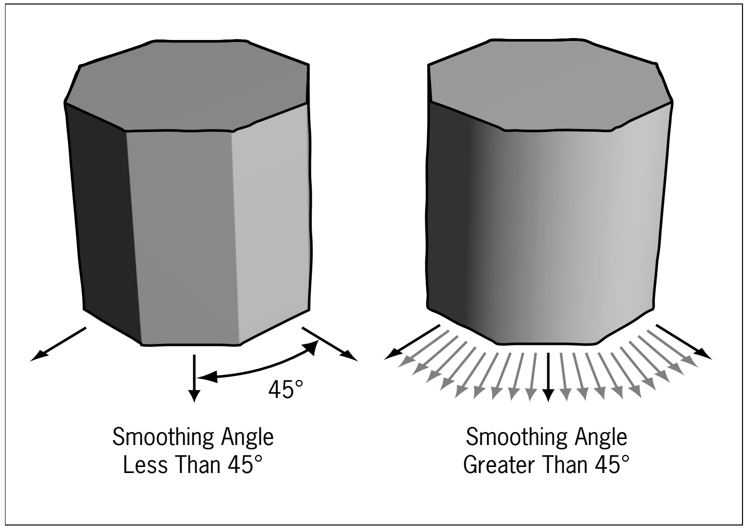


Figure A-1. *Smoothing*

Number of Arc Segments

The Number of Arc Segments parameter controls the number of straight line segments into which the system discretizes arcs, circles, and arc segments in 2D polylines. This parameter specifies the number of segments to use when discretizing a circle. The system discretizes arcs into a number of segments proportional to their subtended angle. For example, the system subdivides an arc spanning 180° into half as many segments as a circle.

Material Maps

Using a *material map* is a convenient technique for automatically assigning materials to surfaces when importing models.

By default, Lightscape assigns a material to a surface whose name and color match the color number in the DXF file. If an item does not have an explicit color associated with it, the system uses the color assigned to the layer the item is on.

Not only are the names of these materials nondescript numbers (usually 1 through 255), but the colors are generally unsuitable for producing acceptable Lightscape results.

With the material map technique, you can associate Lightscape materials with the color numbers used in the DXF file. If the system loads a material map in conjunction with a DXF file, it automatically substitutes all the colors defined in the DXF file with the associated Lightscape materials. For example, you can map a material called "wood" onto every surface that is drawn with color 1.

The first time you work on a model, you should define all the materials you may want to use and then create the material map. Any subsequent DXF files you load for the same model or other models can use the material map to automatically assign the materials defined in the earlier model.

By using the material map technique, you can avoid redefining all the materials each time you reload a DXF file. The actual colors you use when building the model in your CAD application are not important. What is important is to remember that each color number you use represents a specific material in Lightscape. All objects that are the same material should be constructed using the same color number.

The mapping of materials is done from the Material Map dialog box, accessed by choosing Tools>Material Map. Any material defined in the current Materials Table can be assigned to a color index number. Once saved, you can use the material map for any DXF file.

Block Libraries

During DXF import, you can map preexisting Lightscape block and luminaire definitions onto incoming DXF blocks of the same name. If these preexisting blocks or luminaires have been previously formatted for Lightscape, you can save considerable time and effort in the Preparation stage.

During the importing process, the system searches all the loaded block libraries and replaces any block encountered in the DXF file with a block of the same name stored in the libraries. Saving blocks to a Lightscape block library ensures that on each subsequent load of a DXF file, geometry does not need to be reedited in the following Preparation stage.

Orientation Blocks

As discussed in Chapter 5, "Surfaces," the correct orientation (normals) of surfaces is an important issue for generating correct lighting simulations in Lightscape. One method discussed in that chapter for editing the orientation of surfaces is the use of focus points (Figure A-2). To facilitate the orientation process, Lightscape supports a technique whereby you can specify the orientation of surfaces during the modeling process.

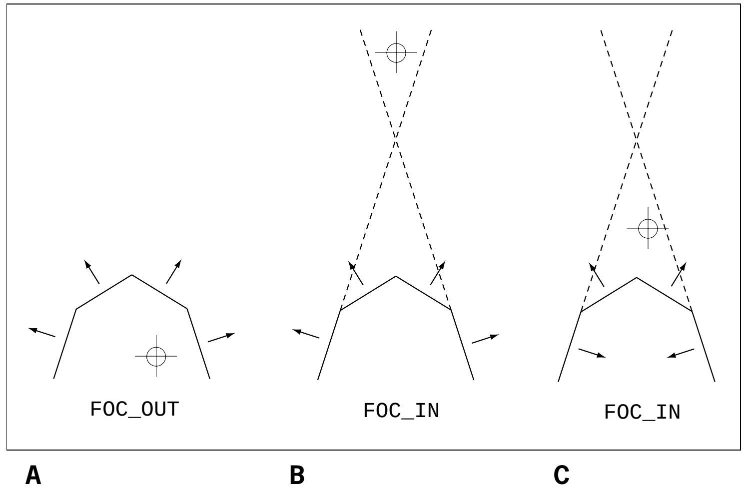


Figure A-2. *Focus Point Examples*

To set focus points in your model, you create a block named either FOC_IN or FOC_OUT. This block can contain some 2D geometry so that it is visible in your modeler, but Lightscape does not import this geometry. The insertion point of this block, regardless of its contents, indicates to Lightscape the orientation (direction of normals) of associated surfaces.

A FOC_IN block orients a surface so that the system sets the surface normals to point toward the insertion point of the FOC_IN block. A FOC_OUT block orients the surface so that the system sets the surface normals to point away from the insertion point of the FOC_OUT block.

Lightscape associates surfaces with a focus point block using the following rules:

1. Each layer can contain a single focus point. The system orients all independent surfaces on that layer in relationship to the inserted focus point block, either toward if the block is called FOC_IN or away if the block is called FOC_OUT.

2. A block can contain a focus point. The system orients all surfaces in the block in relationship to that focus point. In nested blocks, the focus point affects only the surfaces that are part of the specific block into which the focus point is inserted. Surfaces that are part of other sub-blocks are not affected.

3. A focus point in a block takes precedence over a focus point in a layer. For example, if a block with a focus point is added to a layer that has a focus point, the system orients the surfaces in the block in relationship to the focus point in the block and not to the focus point on the layer. However, it orients all other surfaces in the layer in relationship to the focus point of the layer.

Coordinate Translation

Although Lightscape uses the same coordinate system as standard AutoCAD DXF, you may need to transform the coordinates when importing data from other modelers' versions of the DXF file format. Constructing the particular transformation string that converts one coordinate system to another is discussed in Chapter 2, "Importing Geometry."

Command Line Interface

dxf2lp is the stand-alone analog to the DXF Import dialog box in the main application. The following table describes its options.

Usage: **`dxf2lp [options] infile`**

Table A-1. *dxf2lp Translator Parameters*

Option	Extension	Description
-arc	*n*	Number of segments to use in subdividing each circle. Default is 30.
-ang	*n*	Angle for smoothing groups. The -smooth option must be used for this to take effect. Default is 60°.
-bc	[asis | single | color | layer | entity]	Block creation.
	asis	As in DXF file. This is the default.
	single	Single block for the whole file.
	color	One block per color index.
	layer	One block per layer.
	entity	One block per entity.
-bf	*filename*	Block library file to be used for block or luminaire substitution. Can be used multiple times (up to 100) if more than one library files is to be used.
-cap		Set capping on.
-db	*name*	Alternate name for single block. Default is the input filename without its suffix and directory path.
-h		Print a help message.
-mm	*filename*	Material map file to be used for material substitution. If this option is not specified, the default material map file is used.
-o	*filename*	Output filename. Files without an .lp extension are given one. If this option is not used, an .lp extension is substituted for the extension of the input filename.
-s	*scale*	Scaling factor for dimensions in file.
-smooth		Turn smoothing on.
-t	*conversion*	Coordinate system conversion (any permutation of XYZ with optional minus signs). Default is XYZ.
-u	*units*	Length units of model—mm, cm, m, km, in, ft, or mi. Default is m.
-v		Show status messages.
infile		Input DXF file.

Importing 3D Studio Files

3D Studio is a modeling and rendering package from Autodesk that has its own file format for saving scenes. Lightscape imports this format by creating a polygonal mesh based on the objects stored in the 3D Studio file.

Lightscape uses the 3D Studio file toolkit to import 3D Studio files. This toolkit converts all geometry data into triangles, so all Lightscape surfaces that result from importing 3D Studio files are triangles.

Import Parameters

You control the conversion of 3D Studio entities with the following parameters:

- Block Creation
- Layer Creation
- Light Intensity

Block Creation

Although 3D Studio does not create objects analogous to blocks, there are advantages to grouping items together into blocks. In Lightscape, several block creation options are available when you import 3D Studio files. The block options organize all entities into blocks based on the following options:

- None—No blocks are created. Lightscape turns all objects into individual surfaces.
- Single—All entities in the file are organized into a single block.
- Mesh—Each mesh entity is turned into a block.

By default, the name of the block is the same as the object name in 3D Studio. The default block creation mode is none.

Layer Creation

Although 3D Studio does not use layers, several layer creation options are available when you import 3D Studio files. There are advantages to grouping items on layers in Lightscape. The layer options organizes all entities onto layers based on the following options:

- Single—One layer is created and all entities are placed on that layer. The system creates this single layer in addition to the default Lightscape layer. You can specify an alternate layer name.
- Mesh—A layer is created for each mesh item. The name of the layer is the same as the name of the object in 3D Studio. This is the default creation mode.

Light Intensity and Lights

3D Studio supports two kinds of lights—omnidirectional (also know as point lights) and spotlights. For both of these, the system converts the 3D Studio color to a corresponding Lightscape light filter and uses the light intensity multiplier in 3D Studio to scale the luminous intensity. For spotlights, 3D Studio supports circular and rectangular shapes. The system currently converts both of these to the standard Lightscape circular spotlight. Also, Lightscape honors the "no shadow casting" flag of an object.

Note, however, that lighting results using 3D Studio and Lightscape are almost certain to be different because of their completely different lighting algorithms. To convert relative light intensities as specified in 3D Studio files to physical units used by Lightscape, you can use the maximum light intensity scale parameter. (See Chapter 2, "Importing Geometry.")

Coordinate Translation

3D Studio uses the same coordinate axes as AutoCAD (X, Y, Z). You do not need to change the coordinate system translation when importing data from 3D Studio. However, Lightscape accepts a translation, if you want one.

See Chapter 2, "Importing Geometry," for a discussion of how to construct the particular transformation string that converts one coordinate system to another.

Materials

The system converts each 3D Studio material into a Lightscape material definition. It uses the following 3D Studio material attributes—diffuse color, transparency, shininess, shininess strength, shading type, and self illumination.

Texture

Lightscape preserves the texture mapping coordinates set in 3D Studio, but only converts texture map 1 associated to the diffuse color. You can use only texture maps in supported Lightscape formats. (See Chapter 3, “Materials.”)

You can choose to have the texture data of the file ignored on import by selecting the Don’t Read Texture Data option.

Animation

You can only use 3D Studio camera animation. Lightscape does not support other kinds of animation. The system uses Catmull-Rom cubic Bèzier spline construction between provided position points. It linearly interpolates other information (field of view and target point). Lightscape does not currently support 3D Studio spline modifiers, such as bias and tension.

A single 3D Studio file can have several animation tracks. In this case, Lightscape creates separate .la files, named *filename*1.la, *filename*2.la, and so on, where *filename* is the name of the 3D Studio file. You can only generate animation files with the stand-alone command line utility **3ds2lp**. Animation information is ignored from within the Lightscape application.

Other Settings

The system maps 3D Studio background, if it is of type solid color, to Lightscape background. If the background is set to white in 3D Studio, Lightscape converts it to gray so the white lines of the model can be seen. You can change the line color in Lightscape using the options on the Color page of the Document Properties dialog box. If fog is set in 3D Studio, Lightscape converts the fog settings into Lightscape format but turns off fog. It does not translate other background information, such as texture maps, and environmental effects.

When importing files, some translation errors may occur that do not affect the data in the file. You can choose to ignore these error messages when you import a file or you can select the Stop on Translation Errors option to have the import process stop when it encounters an error.

Command Line Interface

The stand-alone 3D Studio reader **3ds2lp** converts a 3D Studio file (.3ds or .prj) into a Lightscape preparation file (.lp). It is the stand-alone analog to the 3D Studio Import dialog box in the main application. The following table describes its options.

Usage: **`3ds2lp [options] infile`**

Table A-2. *3ds2lp Translator Parameters*

Option	Extension	Description
-bc	[none | single | mesh]	Block creation mode.
	none	No blocks. This is the default.
	single	One block for the whole file.
	mesh	One block per mesh.
-db	*name*	Alternate name for single block. Default is the input file name without its suffix and directory path.
-dl	*name*	Alternate name for single layer. Default is the input file name without its suffix and directory path.
-h		Print a help message.
-lc	[single | mesh]	Layer creation mode.

Table A-2. *3ds2lp Translator Parameters (continued)*

Option	Extension	Description
	single	One layer for entire object.
	mesh	One layer per mesh. This is the default.
-li	*n*	Maximum light intensity scale. Default is 2500.0.
-o	*filename*	Output filename. Files without an .lp extension are given one. If this option is not used, a .lp extension is substituted for the extension of the input file name.
-s	*n*	Scaling factor for dimensions in file.
-t	*conversion*	Coordinate system conversion (any permutation of XYZ without optional minus signs). Default is XYZ.
-u	*units*	Length units of model—mm, cm, m, km, in, ft, or mi. Default is m.
-v		Show status messages.
infile		Input 3D Studio file.

Appendix B

Light and Color

This appendix gives a technical overview of light and color and provides information to help you produce higher-quality pictures. It also explains some of the current limitations of the Lightscape system.

The rendering process is primarily concerned with the simulation of light and the display of color. An understanding of light and color is important to use Lightscape effectively. Chapter 3, “Materials,” contains introductory information about light and color. This appendix provides detailed technical information.

Light is part of the physical world; color is our perception of the light that reaches the eye. Radiosity simulates the propagation of light throughout an environment. The image created after the solution should create the same visual response as the real scene. This can be a very difficult problem, either because certain phenomena are not well understood or because current solutions require processing power beyond today’s availability.

Lightscape Technologies continues to research and develop new algorithms to address these limitations, and the company is committed to providing the most accurate and practical solution available.

Light: The Physical World

The radiosity method attempts to model the physical process of the emission of light, its propagation through the environment, and its interactions with materials. An understanding of what light is and how it interacts with materials makes it easier to create realistic-looking images. Radiosity processing gives the best results if the inputs to the simulation are physically accurate. This section describes what light is, how it is represented, and how materials affect it.

Spectra

Light, or the visible spectrum, is electromagnetic radiation with wavelengths between 380 and 780 nanometers (nm). Intensity spectra are descriptions of light. At each wavelength they give the intensity of the light at that wavelength. Spectra are often represented as spectral curves, or graphs showing the intensity at each wavelength.

Luminaires

Luminaires emit energy in the visible spectrum. The spectra of luminaires can vary greatly, depending on the type of luminaire. Figure B-1 shows the spectral curves for two different luminaires. You can get the spectral curves for various lights from lighting manufacturers, but they have not yet adopted an industry standard format such as the IES Data File Format.

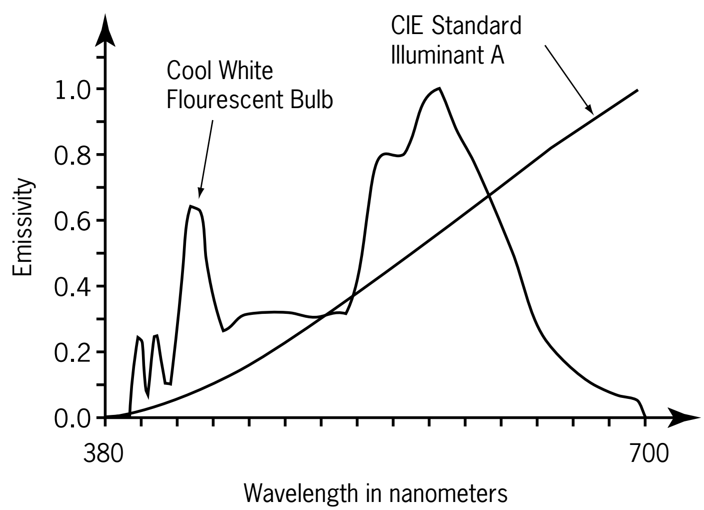


Figure B-1. *Spectral Curves for Two Different Luminaires*

Materials

Materials reflect some of the light that strikes their surface. The reflected light can be determined from the incident light by multiplying the reflectance at each wavelength by the spectrum of the incident light. The result is an intensity spectrum that represents the reflected light. The reflectance of the surface at each wavelength is based on the type of material and is described by a reflectance spectrum.

Materials reflect and absorb some of the light that strikes them at each wavelength. That means that at each wavelength, the reflectance of the surface is greater than 0 and less than 1. In practice, reflectance is significantly greater than 0 and significantly less than 1. Figure B-2 shows the reflectance curves of two different materials. The spectral curves for materials are often difficult to obtain and can vary greatly with different surface finishes and with the age of the material.

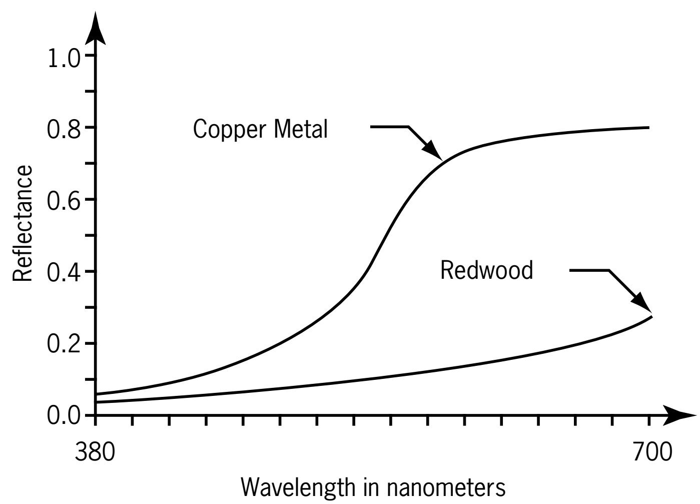


Figure B-2. *Spectral Curves for Two Different Materials*

Reflectance of Materials

The following table gives the average reflectance for a variety of materials.

	Material Type	Reflectance
Nonmetals	Soot, coal	.05
	Felt, black	.18
	Field, plowed	.25
	Marble, white	.54
	Oil paint, white	.70
	Paper, white	.72
Metals	Copper, tarnished	.36
	Stainless steel, polished	.63
	Iron, ground with fine grit	.64
	Aluminum, polished	.80
	Copper, highly polished	.82
	Aluminum, highly polished	.90
	Silver, highly polished	.93

As this table suggests, most nonmetals have relatively low reflectance, but even soot has a reflectance greater than 0. Metals have higher reflectance, but even they are well below 1. Most environments have very little highly polished silver in them.

Setting Reflectance in Lightscape

Proper choice of reflectance is very important for creating realistic images.

If the reflectance is too high, the environment appears flat, because shadows and shading get washed out by the large quantities of indirect illumination. In addition, it takes a long time for the radiosity solution to distribute most of the unshot energy.

If the reflectance is too low, there is insufficient indirect illumination and the environment becomes too dark in regions that are not directly illuminated. You can use the previous table to help define the reflectance of material, as discussed in Chapter 3, “Materials.”

Light in Computer Graphics

Because computer graphics models the interactions of light with surfaces, it needs to represent these spectra. This can be difficult, for several reasons:

- A good representation would increase the time and memory needed to create an image.
- Very little information is available on the spectral reflectance of surfaces and lights; no industry standard formats exist.
- Specifying materials and lights by specifying the spectral curves is not an intuitive process.

For these reasons, computer graphics applications usually approximate spectra using three wavelengths of light—one each of red, green, and blue. These three wavelengths of light are often based on the red, green, and blue values displayed on the screen. In many cases this is not a serious limitation, although it makes it impossible to accurately compute solutions for environments where exact spectral information is known.

Color: The Perceived World

The physical world deals with spectra. When light with a particular spectrum enters the eye it is perceived as a color. This process is very complex and much of it is not well understood. The physiology of the eye determines how the light is transformed into a signal to the brain. Inside the brain, more complicated and less understood perceptual transformations take place that help us to understand the images we see.

The Eye

Two different types of receptors line the retina (back of the eye)—rods and cones. Rods are spread widely over the retina and are sensitive to low levels of light. Cones are concentrated in the center of the retina and are divided into three classes based on the type of pigment they contain.

These pigments make color vision possible because the three classes of cones respond differently to light at different wavelengths. The way cones work also explains why we can describe a color response without needing to describe the entire spectrum of the light striking the back of the eye. It suggests that color can be represented with three values.

Color Matching

Color matching is the process of matching a spot of colored light with some combination of other lights. Researchers have found that if they choose three different lights they can mix various amounts of them and match most test spots. The requirement on the lights is that no two of them can be mixed to produce the third.

The intensity of the lights is a way of uniquely describing the color of the test spot. Some test spots cannot be matched directly. However, all test spots can be matched if one of the lights is allowed to be mixed into the test spot. This is often described as a *negative light*. Negative lights don't exist, but by representing the light shining on the test spot as negative, all test spots can be described as a mixture of the three lights.

Spectra can have a different intensity at each wavelength. Color has only three parameters. This means that there are many more spectra than colors. Many different spectra can give the same perceived color. This is fortunate, because it means that we don't have to store or transmit all the information in a spectrum for each color. It also means a color does not contain enough information to reproduce the spectrum it came from.

Color Spaces

Choosing the three lights to use defines a color space. A color space is just a convenient way of representing a color. Given two different sets of three lights, it is possible to convert from one color space to another.

Because the relationship between spectra and colors is linear and the conversion between these color spaces is linear, most operations on color can be done in any of these color spaces and give identical results.

The problem with all color spaces defined by combinations of three lights is that each color space has ranges of color that can only be described by negative lights (ranges of color it cannot physically reproduce).

Phosphors

The color from a monitor is the result of three different colored phosphors at each pixel mixing at different intensities. The three phosphors act like the three lights in the color-matching experiments. These phosphors are usually described as red, green, and blue, but each manufacturer uses different sets of phosphors for its monitors, based on its needs. This means that the same image shown on two different screens can look very different. A color defined in one color space is being used as if it were defined in another.

A solution to this problem can be found if the phosphors for the monitor on which an image is to be displayed are known. In this case a transformation can be determined between the color space of the image and the color space of the monitor, so that the image can look the same on different monitors.

There is an additional problem with monitors that currently cannot be solved. Because every color space based on physical lights has colors it cannot represent (those requiring negative coefficients), some colors will never show up correctly on a monitor. These colors are called *out of gamut* colors. In general this is not a serious problem, because these colors are very saturated and most real scenes contain few highly saturated colors.

Computing with Color

Although working with color or spectra is equivalent for many operations, they are not equivalent when multiplying two colors or spectra.

This is problematic, because Lightscape spends much of its processing time multiplying colors. In theory, you can get arbitrarily large differences between multiplying two spectra or their equivalent colors. In practice, most materials and most lights, with the notable exception of fluorescent lights, are relatively well behaved.

Color shifts occur if computations are done with color rather than spectra (as they are in Lightscape), but in general they aren't arbitrarily bad. The color shift is minimal with white lights and few interreflections, and is more severe with colored lights and many interreflections. Accurate colors for the lights cause large color shifts and give less than pleasing results.

The colors of the lights in Lightscape are desaturated in order to make the results appear better.

Constraints of Output Devices

This section describes some of the constraints current display devices place on the accurate display of a simulated model:

- White point
- Monitor gamma
- Dynamic range mapping
- Whiteness constancy, adaptation, and surroundings

White Point

All monitors have a maximum intensity color they can produce with the maximum intensities for the red, green, and blue guns. This is called the *white point* of the monitor. This white point is different for different monitors. Usually the white points are defined in terms of color temperature. (Color temperature represents the color of a glowing object heated to the specified temperature.) Most white points lie between 5000°K, an orangish white, and 9300°K, a bluish white. Most televisions are set to 6500°K, a white that is near the color of daylight. This variation in white is another reason why images on one monitor look different from images on another monitor.

Monitor Gamma

The light from the monitor comes from electron guns exciting the phosphors on the screen. This process is not linear. In order to get light from the monitor that is halfway between zero intensity and full intensity, it is necessary to have the guns fire at above half strength. This nonlinearity of the monitor is called the *gamma* of the monitor. Gamma is also used for similar nonlinearities of other display and recording devices.

This is a problem for Lightscape because when you compute a particular intensity, you want to display that intensity, not the intensity produced by distortions of the system.

Many display programs allow an image to be displayed at a particular gamma. You are strongly encouraged to display images at the correct gamma.

Dynamic Range Mapping

Perhaps the greatest constraint of the monitor is its limited dynamic range. *Dynamic range* is the ratio of the highest intensity the monitor can produce to the lowest intensity.

In a dark room this ratio is around 100 to 1. In a bright room the ratio drops down to around 30 to 1. Real environments have dynamic ranges around 10,000 to 1 or larger. There is currently no good way to compress the dynamic range of a real environment to that of a monitor. This is a subject of ongoing research at Lightscape.

Whiteness Constancy, Adaptation, and Surroundings

The brain wants to perceive white surfaces (those with a white reflectance spectrum) as white. A sheet of white paper under fluorescent or incandescent lights looks white, even though neither of these lights is white. White on a monitor in a dark room looks white, even though the white on two different monitors may look very different if you see them side by side.

When viewing a monitor in a lit room, you have adapted to the illumination of the room, not to the illumination of the model. Even if a model is computed and displayed correctly, it may still be seen as if you are looking into the room from the outside—or, more likely, as if the color of the model is wrong.

Appendix C

Lightscape Utilities

This appendix describes a number of batch programs and utilities.

Introduction

In addition to the main Lightscape GUI program (**lscape**), the Lightscape Visualization System supports a number of utilities that can be run in batch files. Using these batch files, you can set up a series of procedures and have them run sequentially over an extended period. In practice, you will find that although all of the setup and preparation of a model is done in **lscape**, most of the processing and rendering are undertaken using the batch utilities. The following pages give some common examples of batch files that are used with Lightscape. This appendix contains descriptions of these utilities.

Note that throughout this appendix there are examples of command lines that are larger than the width of the page. These commands are shown in two or more consecutive lines.

Example 1

Probably the most common use of batch files is to ray trace images from various Solution files, or from the same file using different views or resolutions.

If you want to make multiple images from a single Solution file and the parameters of an image don't change, you can use the -svf option to specify a list of views, as demonstrated in the last line of the following batch file.

```
lsray -aa 4 -vf view1.vw -x 1280 -y 1024
solution1.ls image1.tif

lsray -aa 4 -vf view2.vw -x 640 -y 512
solution1.ls image2.tif

lsray -aa 4 -x 1280 -y 1024 solution2.ls
image3.tif

lsray -aa 4 -x 1280 -y 1024 -svf view1.vw
view2.vw view3.vw -evf solution1.ls image.tif
```

Example 2 You may want to run a series of tests overnight using different processing parameters to see which parameters result in the best radiosity solution.

In this example four tests are run, each for three hours. Notice that the Preparation file remains the same but the Parameters file, which contains the meshing parameters, is changed and each file is saved to a different Solution file.

```
lsrad -v -termt 180 -df  test1.df -cpt 15 -o
test1.ls test.lp

lsrad -v -termt 180 -df  test2.df -cpt 15 -o
test2.ls test.lp

lsrad -v -termt 180 -df  test3.df -cpt 15 -o
test3.ls test.lp

lsrad -v -termt 180 -df  test4.df -cpt 15 -o
test4.ls test.lp
```

Example 3 You may want to run radiosity solutions exploring various design alternatives that exist on different layers in a Preparation or Solution file. In this case you can save Layer State files and use them to control the geometry and/or lights that you want to include in the solution.

```
lsrad -v -termt 180 -lf  alt1.lay -cpt 15 -o
alt1.ls model.lp

lsrad -v -termt 180 -lf  alt2.lay -cpt 15 -o
alt2.ls model.lp

lsrad -v -termt 180 -lf  alt3.lay -cpt 15 -o
alt3.ls model.lp
```

Example 4 You may want to use a complex batch file to render animation frames of a walkthrough of a complex model that has been split into smaller models, so they can be processed more efficiently. This procedure is discussed in Chapter 12, “Rendering.”

In this example, the project is split into three models and an animation file (path.la) is created that spans all three models. The Preview tool is used to establish at which frame various models come in and out of view and to set up the following table. The images are created in JPEG format.

Table C-1. *Frame Setup*

Segment	Frame	Models in View
1	0	1
2	156	1 and 2
3	387	1 and 2 and 3
4	456	2 and 3
5	694	3

```
lsrender -aa 6 -af path.la -blend -ef 155 -x
640 -y 486 model1.ls frames.jpg

lsmerge -o segment2.ls model1.ls model2.ls

lsrender -aa 6 -af path.la -blend -sf 156 -ef
386 -x 640 -y 486 segment2.ls frames.jpg

lsmerge -o segment3.ls segment2.ls model3.ls

lsrender -aa 6 -af path.la -blend -sf 387 -ef
455 -x 640 -y 486 segment3.ls frames.jpg

lsmerge -o segment4.ls model2.ls model3.ls

lsrender -aa 6 -af path.la -blend -sf 456 -ef
693 -x 640 -y 486 segment4.ls frames.jpg

lsrender -aa 6 -af path.la -blend -sf 694 -x
640 -y 486 model3.ls frames.jpg
```

lid2cibse

This utility reads in a luminous intensity distribution from a photometric file and writes it out in the CIBSE file format. **lid2cibse** currently accepts the following file formats as input:

- CIBSE—The format adopted by the Chartered Institution of Building Services Engineers, as specified in technical memoranda TM14; used in Great Britain.
- IES—The format designed by the Illuminating Engineering Society, as described in report LM-63-1991; used in North America.
- LTLI—The format created by the Danish Illuminating Laboratory, Lysteknisk Laboratorium; used in the Scandinavian countries.

Only the luminous intensity distribution data (photometric web) is converted. All other fields and comments, such as the number of lamps and the luminaire manufacturer, are ignored. Note also that the orientation of the photometric web with respect to the luminaire is not converted either. Therefore, when the output file is associated to a luminaire, manual orientation of the photometric web may be required.

Syntax: **`lid2cibse`** ***`[options] input_file output_file`***

Table C-2. *lid2cibse Utility Parameters*

Option	Extension Control Point	Description
-h		Print a help message.
-v		Turn on verbose mode. Print status information during the conversion process.
input_file		Input photometric file.
output_file		Output CIBSE file.

lid2ies

This utility reads in a luminous intensity distribution from a photometric file and writes it out in the IES file format. **lid2ies** currently accepts the following file formats as input:

- CIBSE—The format adopted by the Chartered Institution of Building Services Engineers, as specified in technical memoranda TM14; used in Great Britain.
- IES—The format designed by the Illuminating Engineering Society, as described in report LM-63-1991; used in North America.
- LTLI—The format created by the Danish Illuminating Laboratory, Lysteknisk Laboratorium; used in the Scandinavian countries.

Only the luminous intensity distribution data (photometric web) is converted. All other fields and comments, such as the number of lamps and the luminaire manufacturer, are ignored. Note also that the orientation of the photometric web with respect to the luminaire is not converted either. Therefore, when the output file is associated to a luminaire, manual orientation of the photometric web may be required.

Syntax: ***`lid2ies [options] input_file output_file`***

Table C-3. *lid2ies Utility Parameters*

Option	Extension	Description
-h		Print a help message.
-v		Turn on verbose mode. Print status information during the conversion process.
input_file		Input photometric file.
output_file		Output IES file.

lid2ltli

This utility reads in a luminous intensity distribution from a photometric file and writes it out in the LTLI file format. **lid2ltli** currently accepts the following file formats as input:

- CIBSE—The format adopted by the Chartered Institution of Building Services Engineers, as specified in technical memoranda TM14; used in Great Britain.
- IES—The format designed by the Illuminating Engineering Society, as described in report LM-63-1991; used in North America.
- LTLI—The format created by the Danish Illuminating Laboratory, Lysteknisk Laboratorium; used in the Scandinavian countries.

Only the luminous intensity distribution data (photometric web) is converted. All other fields and comments, such as the number of lamps and the luminaire manufacturer, are ignored. Note also that the orientation of the photometric web with respect to the luminaire is not converted either. Therefore, when the output file is associated to a luminaire, manual orientation of the photometric web may be required.

Syntax: **`lid2ltli`** ***`[options] input_file output_file`***

Table C-4. *lid2ltli Utility Parameters*

Option	Extension	Description
-h		Print a help message.
-v		Turn on verbose mode. Print status information during the conversion process.
input_file		Input photometric file.
out_file		Output LTLI file.

ls2iv

The **ls2iv** command can be used to convert Lightscape Solution files (.ls) to Silicon Graphics Open Inventor format. The Open Inventor run-time library must be installed to run **ls2iv**.

It is possible to specify coordinate system translation, additional scale factor, and units when exporting to Open Inventor.

Usage: `ls2iv` *[options] solution_file*

Table C-5. *ls2iv Utility Parameters*

Option	Extension	Description
-a		Include active layers only.
-b		Output in binary.
-h		Print help message.
-ldc	*n,n, …*	Level of detail area cutoffs. Multiple areas are separated by commas. This list is ordered from the largest (finest) LOD to the smallest (coarsest) LOD. Areas are in pixels and must be positive integers.
-lg	*n*	Geometric progression of cutoffs where *n* represents the cutoff area (in pixels) of the coarsest level of detail. Successive areas are calculated by *n**4, *n**4*4, and so on. If "-ml *n*" is specified, only the coarsest *n* levels of detail are included.
-lod		Enable the level of detail hierarchy. If this parameter is not specified, the parameters -ldc, -lg, and -ml have no effect and only the finest level of detail is output.
-ml	*n*	Maximum level of detail to convert. *n* is a non-negative integer. Level 0 is the coarsest LOD. When -lg is specified, the default is to output all LOD's. When -ldc is specified, the default is to output *n*+1 LOD's where *n* is the number of cutoffs listed or all LOD's, whichever is less.
-nt		Don't output textures.
-o	*filename*	Output filename. Files without an .iv extensions are given one.

Table C-5. *ls2iv Utility Parameters (continued)*

Option	Extension	Description
-s	*n*	Scaling factor for dimensions in file.
-t	*conversion*	Target coordinate system (any permutation of XYZ with optional minus signs). Default is X-ZY.
-tem		Embed textures in outfile. Default: reference textures by filename only.
-u	*units*	Length units of model—mm, cm, m, km, in, ft, or mi. Default is m.
-v		Show status messages.
solution_file		Input solution file.

Setting Level of Detail (LOD)

Lightscape models are comprised of surfaces that are geometrically subdivided. For example, a rectangle is subdivided into four smaller portions by dividing itself in half in both directions. Each one of these quadrants can be further subdivided into four smaller quadrants, and so on. A surface that has been subdivided five times can yield up to 1024 smaller subdivisions or mesh elements.

At the coarsest level of detail (LOD), only the largest rectangle, the first one, is displayed. At the next LOD, four elements are displayed and so on until the finest LOD is displayed, which would include all 1024 elements. A word of caution—at coarser LOD's, you can notice a reduced quality of display as well as shifts in the lighting levels. However, finer levels of LOD's may degrade the interactive speed when the model is viewed with an Open Inventor viewer.

By default, the finest level of detail is rendered. You can specify the maximum LOD to render, as well as the *sequence of area cutoffs*. The LOD of a surface actually rendered depends on the number of pixels covered by an approximation of the surface. If the approximation covers *n* pixels and the first (largest) area cutoff is smaller than *n*, then the first (most detailed) LOD is rendered. If *n* is between the first and second cutoffs, then the second finest LOD is rendered and so on. If *n* is smaller than the smallest cutoff, then the coarsest LOD is rendered.

ls2vrml

You can use the ls2vrml command to convert a Lightscape Solution file (.ls) to a VRML version 1.0c file

Usage: **`ls2vrml`** *[options] solution_file*

Table C-6. *ls2vrml Utility Parameters*

Option	Extension	Description
-a		Include active layers only.
-bf	n^1,n^2,n^3 …	Hierarchy subdivision branching factors. Must be integers which are greater than or equal to 2. Lightscape uses the subdivision when creating inline nodes. It initially subdivides the model into a 3D grid $n^1xn^1xn^1$. The system associates surfaces that fall completely within a grid node with that node. Grid nodes themselves can be further subdivided into $n^2xn^2xn^2$ subnodes and so on. The default is one level.
-c		Don't compact file. The default is to compact the VRML file resulting in smaller file sizes at the expense of some precision and readability.
-h		Print help message.
-iw		Don't use WWW Inline nodes. By default, the program writes out many subfiles that are inlined by the main file. Inlining can improve the perceived performance of downloading your model. Subfiles are downloaded only as required by the browser.

Table C-6. ls2vrml Utility Parameters (continued)

Option	Extension	Description
-ldc	n,n, …	Level of detail range cutoffs. The values must be increasing real numbers. Multiple distances are separated by commas and are ordered from the closest (finest) LOD to the farthest (coarsest) LOD. Distances are in scaled model units, i.e. the units of the input model times the scale factor provided with the -s option.
-ml	n	Minimum LOD to convert. n is a non-negative integer. Level 0 (the default) is the coarsest LOD.
-nt		Don't output textures.
-o	filename	Output filename. Files without a .vrl extension are given one. Default is to use the same base name as the input file.
-s	n	Scaling factor for dimensions in file.
-t	coord	Target coordinate system (any permutation of XYZ with optional minus signs). Default is X-ZY.
-u	unit	Length units of model—mm, cm, m, km, in, ft, or mi. Default is m.
-url	name	Prepends name to inline node URLs.
-v		Show status messages. May appear multiple times for increased verbosity.
infile		Input solution file.

Textures are never embedded in the VRML file. Only a reference to the texture file is written. This reference is a filename, not a URL. You may need to edit the VRML file by hand to allow textures to be found across a network.

The -bf option is used to subdivide the model into spatially related submodels. Each of these submodels is placed into its own file and included by the main file using WWWInline nodes. The idea is to group objects of similar size that are near each other into units that a browser can download on an as needed basis. If the model is a room, the main file would include the floor, ceiling, and walls. Subfiles might include a table or chairs. The table subfiles might reference subfiles with books or a telephone. A browser would then be able to quickly download and display the coarse features of the room (for example, walls), while continuing to download the details (for example, the table and books).

lsmerge

The **lsmerge** utility merges different Lightscape files into a single Preparation file or Solution file. The input to **lsmerge**, in addition to options, consists of a list of Lightscape files. Different Lightscape files can be present in the list (see the following list of supported file types), but the first file in the list must be either a Preparation or a Solution file. Preparation files and Solution files cannot be mixed in the same list.

Unless the -o option is specified, the first file in the list is overwritten with the result of merging all subsequent files. This operation is basically equivalent to loading the first Preparation or Solution file into the **lscape** application and then sequentially adding all the other files in the list. Thus the original view, defaults, materials, and so on may be changed as a result of this operation. If other Preparation or Solution files are present, they are merged with the first file. Keep in mind that block and material definitions overwrite existing definitions, and that data on layers with the same names is merged.

The following file types are supported:

- Block Library files (.blk) (only if the first file is a Preparation file)
- Properties files (.df)
- Layer Library files (.lay)
- Material Library files (.atr)
- Preparation files (.lp) (if the first file is a Preparation file)
- Solution files (.ls) (only if the first file is a Solution file)
- View files (.vw)

Syntax: **lsmerge *[options] file1 file2 ...***

Table C-7. *lsmerge Utility Parameters*

Option	Extension	Description
-a		Add active layers only.
-h		Print a help message.
-i		Interactively confirm overwriting of existing files.
-o	*filename*	Alternate output filename. Save the result into this file instead of overwriting.
file1, file2 …		Lightscape files. The first file must be either a Preparation file or a Solution file.

Example 1

```
lsmerge -o room1.lp room.lp mymater.atr
myview.vw mydef.df
```

This command creates a new Preparation file room1.lp from the existing file room.lp, by adding specified material, view, and properties files.

Example 2

```
lsmerge outside.ls inside.ls
```

This command merges the Solution file inside.ls to outside.ls and writes the result to outside.ls.

lsmkray

The **lsmkray** utility can be used for automatic generation of a *ray file*. This ray file can be passed as an option to **lsray**, so that only specified rays in the ray file are traced. You can use this utility to create special effects, such as images suitable for a QuickTimeVR *viewer*.

This utility only generates ray files for panoramic viewing. That is, the observer position is conceptually enclosed by a sphere. The viewer can look at the *region of interest* on the surface of the sphere. This region could be the whole sphere, or it could be enclosed by the meridian and parallel lines. Depending on the provided resolution, rays are shot from the viewer into the region of interest.

The orientation of the sphere is defined by the vertical and horizontal axes, which are set by default to the Z axis and X axis respectively. The vertical axis passes through the poles of the sphere. The intersection of the plane formed by the vertical and horizontal axes with the sphere defines the zero meridian on the sphere.

This utility generates only panoramic ray files, as just described. If you want to create a special effect that doesn't fit into this model, you will have to create a ray file, either manually or by other means.

The format of the ray file is as follows. The first line contains the width and height of the image. The following width times height lines contain the XYZ coordinates of the beginning and end of each ray (six numbers per line).

Syntax: **`lsmkray`** ***`[options] ray_file`***

Table C-8. *lsmkray Utility Parameters*

Option	Extension	Description
-fov	*angle*	Vertical field of view. The observer looks at the region bounded by –angle/2 to angle/2 parallels. Default is 90°.
-h		Print a help message.
-hfov	*angle*	Horizontal field of view. Extends from –angle/2 to angle/2 with respect to the zero meridian. Default is 360°, corresponding to full panorama.
-hor	*x y z*	Horizontal axis. Used for sphere orientation. Horizontal and vertical axes must be orthogonal; if they are not, horizontal axis is adjusted accordingly. Default is (1, 0, 0).
-offset	*n*	Give an offset, in meters, for the eye along the horizontal axis. Default is 0.
-pos	*x y z*	Observer position. Select this position in the center or inside the model. Default is (0, 0, 0).
-rad	*radius*	Radius of the sphere. Default is 1.
-to		Generate rays toward the observer instead. Default: rays are oriented away from the observer.
-ver	*x y z*	Vertical axis. Used for sphere orientation. Vertical axis passes through the poles of the sphere. Default is (0, 0, 1).
-x	*size*	Image width. Default is 500.
-y	*size*	Image height. If only one of width and height is specified, the other dimension is derived from the horizontal to vertical field of view angles ratio.
ray_file		Output ray file name.

Example **lsmkray -hfov 180 -fov 60 -pos 0 0 2 -x 300 rayfile**

This command places the observer at (0, 0, 2) looking at the portion of the eastern hemisphere bounded by –30° and 30° parallels. The size of the image is 300 X 100, corresponding to the ratio of horizontal and vertical field of view angles.

Once the ray file is created, it can be passed to the **lsray** program, for example:

lsray -v -rf rayfile room.ls image.rgb

lsrad

The **lsrad** command is used to process radiosity solutions. Although it is possible to process a radiosity solution in the **lscape** application, **lsrad** is more efficient, because there is no redisplay between each iteration.

The input to the **lsrad** program can be either a Preparation file (.lp) or a Solution (.ls) file. In the case of the Preparation file the data is initiated first and the processing uses the meshing parameters specified in it. In the case of the Solution file, the processing continues from the last iteration completed.

Unless you provide an alternate file name using the -o option, the original Solution file is overwritten with the computed solution. If the Preparation file was provided, a new Solution file is created with the same prefix as the Preparation file, but with an .ls suffix.

By default, the process runs indefinitely. To stop the process, type **Ctrl+C** in the window where the process is running. If you type **Ctrl+C** again, the process terminates without saving any files. Otherwise the process completes the iteration it is working on and outputs a Solution file before stopping.

It is also possible to specify in advance the number of iterations or processing time before exiting, using the -term or -termt option.

Syntax: **`lsrad [`*`options`*`]`** ***`filename`***

Table C-9. *lsrad Utility Parameters*

Option	Extension	Description
-ac		Allow attribute and light-source changes. Default: use Solution file information. See Chapter 8, “Solution,” for more information.
-nac		Don’t allow attribute/light changes for more efficiency. Default: use Solution file information.
-cp	*n*	Iteration-based checkpoint. Output a Solution file every *n* iterations. This is useful when running extended processes—overnight, for example—to ensure that the results are saved periodically to disk in case of power failure or other problems. The output file specified is continuosly overwritten with the latest results. Default is no checkpointing.
-cpt	*n*	Time-based checkpoint. Output a Solution file every *n* minutes. Default is no checkpointing.
-df	*filename*	Load specified Parameters file, overriding those specified in the Solution file.
-do		Process direct light sources only.
-h		Print a help message.
-i		Interactively confirm overwrite of existing files. Default is overwrite existing files without confirmation.
-lf	*filename*	Load specified Layer State file.
-o	*filename*	Output the solution to the filename specified instead of overwriting the original Solution file that was loaded.
-q		Query. Print extra information about the process.
-r		Reset the solution before processing.
-sh	[all|direct|none]	Shadow testing.

Table C-9. *lsrad Utility Parameters (continued)*

Option	Extension	Description
	all	Calculates all shadows (default).
	direct	Calculate direct shadows only.
	none	Don't calculate any shadows.
-term	*n*	Terminate the program and output the Solution file after *n* iterations. Default is no limit.
-termt	*n*	Terminate the program and output the Solution file after *n* minutes. Default is no limit.
-v		Verbose. Print extra information after every iteration.
filename		Input Preparation or Solution file.

Example 1

```
lsrad -cp 20 -v room.lp
```

This command reads a Preparation file room.lp, initializes it, and runs a radiosity process with a checkpoint every 20 iterations. The process can be stopped by typing **`Ctrl+C`** as described earlier. The Solution file will be room.ls.

Example 2

```
lsrad -cpt 3 -termt 15 -o room1.ls -sh none
room.ls
```

This command reads a Solution file and continues processing it for another 15 minutes with no shadow computation, with checkpoints every 3 minutes. The output file will be room1.ls.

lsray

As discussed in Chapter 12, “Rendering,” Lightscape uses a ray tracing postprocess to add global illumination effects such as specular reflections and transparency. The ray tracing can also be used to improve the shadows and lighting effects cast by specific light sources. Although it is possible to ray trace images directly in the **lscape** application, it is faster and sometimes more convenient to produce the images using this batch ray tracer. In addition, more advanced ray tracing options are available in the **lsray** program.

The **lsray** program takes as input any Solution file and generates an appropriate image file. It is also possible to produce a series of image files, corresponding to a list of view files, or an animation file. Textures, if present, are loaded using the current texture path list.

The extension of the image file determines the format into which the image will be saved. The following extensions are supported:

- .bmp (Windows native file format)
- .tga (Targa, True Vision format)
- .tif (TIFF—24-bit or 48-bit)
- .rgb (RGB—24-bit or 48-bit, native Silicon Graphics file format)
- .jpg (JPEG)
- .gif (Compuserve 8-bit graphic format)
- .png (Portable Net Graphics)
- .eps (Encapsulated Postscript)

The program stops when image computation is completed and saved.

Syntax: `lsray [options] solution_file image_file`

Table C-10. *lsray Utility Parameters*

Option	Extension	Description
-aa	1-10	Antialiasing factor. Higher factors result in higher image quality, but take more computation time. Default is 1. See Chapter 12, "Rendering," for more information.
-aaa	*t n r*	Antialiasing threshold, sampling level, and radius. See Chapter 12, "Rendering," for more information.
-af	*filename*	Animation file. Ray trace all frames specified in the animation file. The image filename is used as the base name and a decimal four-digit number, corresponding to the frame number, is appended for each image file—for example, anim0000.rgb, anim0001.rgb, and so on.
-alls		Compute shadows from all layers. Default: as specified in the Solution file.
-alpha		Output alpha channel information in the image file.
-amb	*n*	Ambient level (range from 0 to 200). Default: as specified in the Solution file.
-bd	*n*	Available for .rgb and .tif files only. Choose 24-bit or 48-bit color for the output image.
-bg	*r g b*	Background color (range from 0 to 255). Default: as specified in the Solution file.
-bri	*n*	Brightness (range from 0 to 200). Default: as specified in the Solution file.
-contr	*n*	Contrast level (range from 0 to 100). Default: as specified in the Solution file.
-df	*filename*	Load specified Parameters file.

Table C-10. *lsray Utility Parameters (continued)*

Option	Extension	Description
-ef	*n*	Last frame of the animation desired. -af option must be used. Default: last frame specified in the animation file.
-fogc	*r g b*	Fog color (range from 0 to 255).
-fogd	*n*	Fog density (range from 0 to 1).
-fogf	[none | linear | fog | haze]	Fog function. Default is none. See Chapter 5, "Surfaces," for more information about fog functions.
	none	No fog.
	linear	Linear fog.
	fog	Models natural fog.
	haze	Models natural haze.
-fps	*n*	For animations, number of frames per second. -af option must be used. Default: as specified in the animation file.
-gl		Use OpenGL reflection model. See Appendix D.
-h		Print a help message.
-il		Output interlaced images for animation. -af option must be used. See Chapter 12, "Rendering," for more information about interlacing.
-lf	*filename*	Load specified Layer State file.
-nc		Don't perform backface culling.
-nt		Don't load textures.
-odd		For interlacing, output first frame with odd scanlines. Sets -il option; -af option must be used. Default: output first frame with even scanlines.
-rb	*n*	Number of reflection bounces to trace. Default is 10.
-recover	*filename*	Recover scanlines from unfinished image file. Useful for continuing work in case the processing was interrupted by power failure or other problems.

Table C-10. *lsray Utility Parameters (continued)*

Option	Extension	Description
-rf	*filename*	Custom ray file. Instead of ray tracing the specified view, trace the rays specified in the ray file. This is useful for making panoramic images (see **lsmkray** earlier in this appendix). The format of ray file is that the first line has width and height dimensions. The following width X height lines have beginning and end coordinates of each ray (six numbers per line). If this option is specified, the -x, -y, -af, -vf, and -svf options are ignored.
-roi	$x^1\ y^1\ x^2\ y^2$	Ray trace only the rectangular *region of interest* defined by the lower left and upper right corners.
-sf	*n*	First frame of animation desired. -af option must be used. Default: as specified in the animation file.
-sh		Recompute shadows from sun and other light sources.
-soft		Compute soft shadows. Valid for sun light source only.
-step	*n*	For animations, interval for frame output. -af option must be used. Default is 1.
-svf	*filename... -evf*	List of view files. -evf must be used to terminate the list. Output image files corresponding to the name of each view file in the list. The image filename is combined with the prefix of each view filename. For example, using an image filename of data.rgb and view files pnt1.vw, pnt2.vw, and pnt3.vw results in images named datapnt1.rgb, datapnt2.rgb, and datapnt3.rgb.
-v		Verbose. Print information about the status of the image.
-vf	*filename*	Load specified view file.
-x	*n*	Image width.
-y	*n*	Image height. If only width or height is provided, the other dimension is derived from the aspect ratio of the view. Default is 256.
-w		Display the results interactively in the Graphic Window. This option can only be used when the resolution of the image fits within the resolution of the monitor.

Table C-10. lsray Utility Parameters (continued)

Option	Extension	Description
-wp	*xpos ypos*	Same as above, but place the Graphic Window in the specified location on the monitor. Default: window is placed in the center of the screen.
solution_file		Solution file to run ray tracing on.
image_file		Image file to save the results of ray tracing.

Example

```
lsray -aa 3 -vf view.vw -sh -rb 2 -x 640 -y
512 room.ls image.rgb
```

This command loads the Solution file room.ls and generates a 640 X 512 resolution image called image.rgb using the view specified in view.vw. The image is antialiased (level 3), and two levels of reflections are rendered. Any sunlight or direct light from specified luminaires is also ray traced to produce better shadows.

lsrender

The **lsrender** command creates images that are displayed using OpenGL rendering. The images are not ray traced, and therefore can be generated much faster as compared to the **lsray** utility. Use **lsrender** to rapidly create images that do not require specular reflections and accurate transparency effects.

Although it is possible to generate images in the **lscape** application, it's usually more convenient to use this batch utility. In addition, more advanced options are available in **lsrender**.

The **lsrender** program takes as input any Preparation file or Solution file and generates an appropriate image file. It is also possible to produce a series of image files, corresponding to a list of view files, or an animation file. The resulting images are always displayed in a Graphic Window. Textures, if present, are loaded using the current texture path list.

The extension of the image file determines the format into which the image will be saved. The following extensions are supported:

- .bmp (Windows native file format)
- .tga (Targa, True Vision format)
- .tif (TIFF—24-bit or 48-bit)
- .rgb (RGB—24-bit or 48-bit, native Silicon Graphics file format)
- .jpg (JPEG)
- .gif (Compuserve 8-bit graphic format)
- .png (Portable Net Graphics)
- .eps (Encapsulated Postscript)

The 48-bit color output is available only if your graphics card supports that display mode.

The program stops when image computation is completed and saved.

Syntax: **`lsrender [options] lvs_file image_file`**

Table C-11. *lsrender Utility Parameters*

Option	Extension	Description
-aa	1–10	Antialiasing factor. Higher factors result in higher image quality, but take more computation time. Default is 1. See Chapter 12, "Rendering," for more information.
-af	*filename*	Animation file. Ray trace all frames specified in the animation file. The image filename is used as the base name and a decimal four-digit number, corresponding to the frame number, is appended for each image file—for example, anim0000.rgb, anim0001.rgb, and so on.
-amb	*n*	Ambient level (range from 0 to 200). Default: as specified in the Solution file.
-bd	*n*	Available for .rgb and .tif files only. Choose 24-bit or 48-bit color for the output image.
-bg	*r g b*	Background color (range from 0 to 255). Default: as specified in the Solution file.
-blend		Set blending on.
-bri	*n*	Brightness (range from 0 to 200). Default: as specified in the Solution file.
-contr	*n*	Contrast level (range from 0 to 100). Default: as specified in the Solution file.
-df	*filename*	Load specified Parameters file.
-dm	[hiddenline | hiddenmesh | mesh | shaded | wireframe]	Display mode.
	hiddenline	Display image as hidden lines.
	hiddenmesh	Display image as a mesh with hidden lines removed.
	mesh	Display image as a mesh with all lines shown.
	shaded	Display a shaded image (default).
	wireframe	Display a wireframe image.

Table C-11. *lsrender Utility Parameters (continued)*

Option	Extension	Description
-ef	*n*	Last frame of the animation desired. -af option must be used. Default is the last frame specified in the animation file.
-enh		Enhanced display mode (available for Preparation files only).
-fogc	*r g b*	Fog color (range from 0 to 255).
-fogd	*n*	Fog density (range from 0 to 1).
-fogf	[none | linear | fog | haze]	Fog function. Default is none. See Chapter 5, "Surfaces," for more information about fog functions.
	none	No fog.
	linear	Linear fog.
	fog	Models natural fog.
	haze	Models natural haze.
-fps	*n*	For animations, number of frames per second. -af option must be used. Default: as specified in the animation file.
-h		Print a help message.
-il		Output interlaced images for animation. -af option must be used. See Chapter 12, "Rendering," for more information about interlacing.
-la		Perform line antialiasing.
-lf	*filename*	Load specified Layer State file.
-nc		Don't perform backface culling.
-nt		Don't load textures.
-odd		For interlacing, output first frame with odd scanlines. Set -il option; -af option must be used. Default: output first frame with even scanlines.
-sf	*n*	First frame of animation desired. -af option must be used. Default: as specified in the animation file.
-step	*n*	For animations, interval for frame output. -af option must be used. Default is 1.

Table C-11. lsrender Utility Parameters (continued)

Option	Extension	Description
-svf	*filename… -evf*	List of view files. -evf must be used to terminate the list. Output image files corresponding to the name of each view file in the list. The image filename is combined with the prefix of each view filename. For example, using an image filename of data .rgb and view files pnt1.vw, pnt2.vw, and pnt3.vw results in images named datapnt1.rgb, datapnt2.rgb, and datapnt3.rgb.
-v		Verbose. Print information about status of the image.
-vf	*filename*	Load specified view file.
-x	*n*	Image width.
-y	*n*	Image height. If only width or height are provided, the other dimension is derived from the aspect ratio of the view. Default is 256.
-w		Display the results interactively in the Graphic Window. This option can only be used when the resolution of the image fits within the resolution of the monitor.
-wp	*xpos ypos*	Same as above, but place the Graphic Window in the specified location on the monitor. Default: window is placed in the center of the screen.
lvs_file		Solution or Preparation file for image generation.
image_file		Image file to save the results.

Example

```
lsrender -bg 0 0 255 -dm wireframe -svf v1.vw
v2.vw v3.vw -evf -v room.lp image.rgb
```

This command loads Preparation file room.lp, sets the background color to blue, and generates wireframe images corresponding to the view files v1.vw, v2.vw, and v3.vw in the current directory.

soft2lp

The **soft2lp** command converts a SOFTIMAGE 3D scene to a Lightscape Preparation file (.lp) and an Animation file (.la).

Lightscape supports most of the texture mapping options in SOFTIMAGE. Texture mapping options not supported in the imported file include bump mapped textures, textures with clipping regions, and multiple textures applied to a single object.

Lightscape ignores clipping regions on textures in the SOFTIMAGE file when importing. It simply imports the entire texture. If textures must be clipped in SOFTIMAGE, you need to clip them by hand.

Lightscape ignores multiple textures applied to a single object. It imports the last texture applied to the object. This is the last texture in the texture list in SOFTIMAGE.

In Lightscape, several block creation options are available when you import SOFTIMAGE files. The block options organize all entities into blocks based on the following options:

- None—No blocks are created.
- Single—All entities in the file are organized into a single block. The block name is based on the scene filename if you do not specify a different filename.
- Top Level—A block is created for each top level object at the root of the model hierarchy.
- Per Model— A block is created for every object in the scene. This option preserves the instancing of objects in the SOFTIMAGE scene.

Loading a block with the same name as an existing block replaces the existing block definition. The default block creation mode is per model.

Lightscape places all of the imported objects on one layer. The default name for this layer is SOFTIMAGE. You can specify a different name for the layer created when you import the SOFTIMAGE file.

Camera animation in the SOFTIMAGE file is imported only if the camera path is specified as a position and/or an interest path. Lightscape does not import camera constraints.

The radiosity algorithm does not allow geometric animation. Thus, animated objects won't be animated on import. They will be imported as static objects.

Lightscape will import most lights set in a SOFTIMAGE file. The system, however, cannot import settings for mental ray extended or area lights.

Lights will be imported with an intensity determined by the intensity multiplier set by the -li option.

Sunlight (daylight) settings in SOFTIMAGE are also supported.

Usage: **`soft2lp`** *[options] scene-name*

Table C-12. *soft2lp Utility Parameters*

Option	Extension	Description
-a	file	Animation path output filename.
-bc	[none | single | model | top]	Block creation.
	none	No blocks created.
	single	One block created for the entire object.
	model	One block per model. This is the default.
	top	One block per top-level model.
-h		Print a help message.
-D	dbDir	Database directory. Default is [`$SI_DBDIR`].
-db	name	Name of single block. Default is full scene name.
-dl	name	Name of layer. Default is [`SOFTIMAGE`].
-li	n	Maximum light intensity scale. Default is 2500.0. This option scales the intensities of lights imported from SOFTIMAGE without losing their relative intensities. For example, if the maximum lighting intensity maps to 2500 candelas (cd), then the system maps a light that is 0.5 maximum intensity to 1250 cd.
-nt		Ignore all texture data.
-nc		Don't convert texture images. Leaves filenames and UV coordinates in place but modifies filenames to have .rgb extension. Ignored if the -nt option is used.
-o	file	Output filename. Files without an .lp extension are automatically given one. Default is to use the same base name as the input file.
-ow		Overwrite existing texture images in the texture output directory.
-R	rsrcDir	Resource directory. Default is [`$SI_LOCATION/3D/bin/rsrc`].

Table C-12. *soft2lp Utility Parameters (continued)*

Option	Extension	Description
-rn	n	Scene version number. Must be greater than or equal to 1. Default is the latest version.The version number tells Lightscape which version of the SOFTIMAGE scene to import. If a version number is not specified, the system imports the latest version of the scene.
-s	n	Scaling factor for dimensions in file.
-T	textDir	Texture output directory. This option is ignored if -nt or -nc is specified.
-t	coord	Target coordinate system (any permutation of XYZ with optional minus signs). Default is X-ZY.
-ta	n	Patch tessellation accuracy (range from 0 and 1 inclusive).Use this option to set a multiplier to the step parameter you set in SOFTIMAGE. This option determines how finely the system tessellates spline based objects.
-u	unit	Length units of model—mm, cm, m, km, in, ft, or mi. Default is m.
-v		Increase verbosity (may appear multiple times).

Appendix D

Reflection Models

This appendix describes the reflection models you can use to create highly-realistic images with Lightscape. To aid you in understanding these reflection models, it also introduces you to how light interacts with surfaces.

The physical behavior of light interacting with surfaces is approximated by a variety of *reflection models,* which make different approximations and are useful in different situations. Lightscape uses reflection models during three processes—during the radiosity computation, during OpenGL rendering, and during ray tracing.

The reflection model for radiosity processing is never seen directly. It is simply used by the radiosity algorithm to determine how much light is reflected from the surfaces in the environment.

Radiosity and OpenGL use very similar lighting models and have similar restrictions.

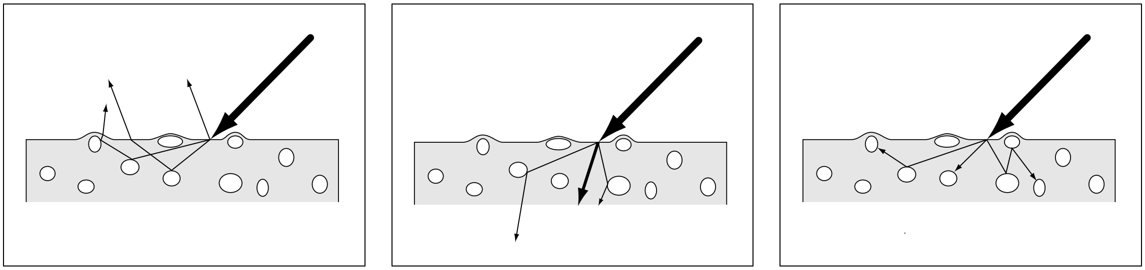
With the ray tracer, you have the option of using two different lighting models. One has the same set of restrictions as the OpenGL display to allow compatibility between these two renderers. The other has fewer restrictions and can be used to produce images of very high realism.

Light and Materials

In Lightscape, you use a material's color and reflectance characteristics, as well as whether or not it is a metal, to describe its scattering appearance. Setting these properties of the material is as important as placing the lights to appropriately model light using Lightscape.

Reflection, Transmission, and Absorption of Light

Light interacting with a material can behave in various ways, as shown in Figure D-1.



Reflection **Transmission** **Absorption**

Figure D-1. *Reflection, transmission, and absorption of light*

The light can be *reflected*, *transmitted*, or *absorbed* by the material:

- *Reflected light* is all the photons that bounce back from the material and continue to move through the environment. Light can be reflected both from where the material meets the air (their interface) and from within the material. Some of this light is reflected specularly and some diffusely. "Interactions at the Interface" on page D-5 discusses specular and diffuse reflection.
- *Transmitted light* is all the photons that pass completely through to the other side of the material. Lightscape only models the specular component of transmitted light. "Transmitted Light" on page D-9 discusses this in more detail.

- *Absorbed light* is light that passes into the body of the material and stays there. This light neither passes through nor is reflected back. The fact that photons of a particular wavelength are absorbed while others are not determines the *color* of the material.

At any given point on a surface, photons may have arrived directly from a light source (*direct illumination*) or else indirectly through one or more bounces off other surfaces (*indirect illumination*). The combination of direct and indirect illumination is the *incident light* at that point.

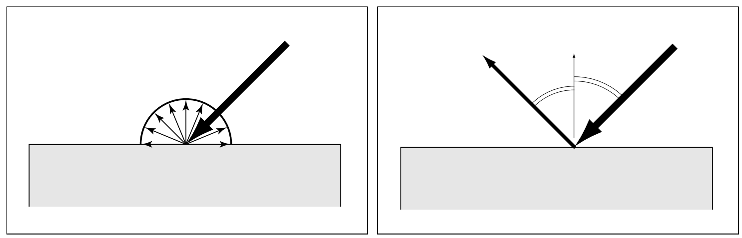
The final illumination of a space is determined by interaction between the surfaces in the space and incident light in the space. When you turn on a light in a room, some of the emitted photons are absorbed by the first surface they reach. Others reflect off many surfaces before being absorbed. Some of the reflection happens at the interface between the surface and the air and some happens below this interface.

When you specify the properties of the materials used on the surfaces of a room, you're in effect specifying where and how photons are reflected, transmitted, and absorbed. These properties affect how the system models interactions between the material and light at the material to air interface, within the material, and coming out the far side of the material. The following sections describe what happens in these areas, to help you to effectively specify these properties.

Interactions at the Interface

Where light hits a material is the *interface* between that material and the air. At the interface, some light continues into the interior of the material and some reflects off the interface. The next section describes how Lightscape determines how much light gets into the interior.

Light reflected at the interface has components of both *specular reflection* and *diffuse reflection*, as shown in Figure D-2. These components are responsible for different lighting effects.



Diffuse **Specular**

Figure D-2. *Diffuse and Specular Reflection*

In specular reflection, the light being reflected leaves the interface at the same angle at which it arrived. Specular reflections provide clear reflections off shiny surfaces, such as seeing a face reflected off a wall. A mirror is a perfectly specular surface; that is, all of the light reflected at the interface is reflected in the specular direction.

There are two types of diffuse reflection—uniform diffuse and directional diffuse. *Uniform diffuse reflection* accounts for light that is scattered uniformly in all directions. In *directional diffuse reflection*, sometimes called *specular highlight*, the light leaves the surface at various angles. Directional diffuse reflections do not provide clear reflections. Instead, they provide highlights, such as the bar of shininess on a door knob where the light hits it at the right angle.

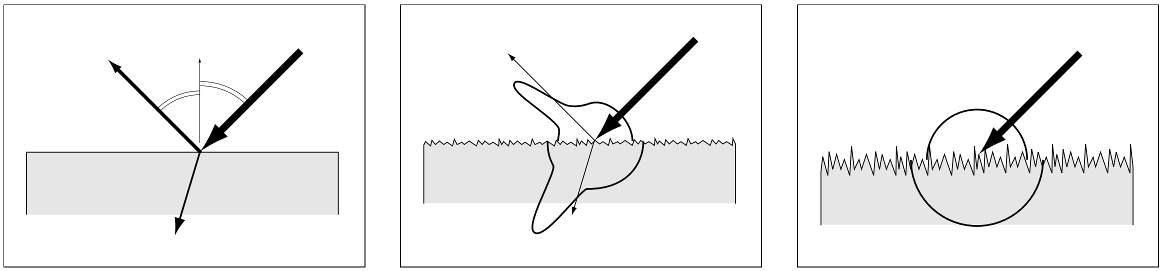
For most rendering techniques, you do not need to understand the consequences of directional diffuse reflection. It can only be calculated when ray tracing and refining shadows using the high-quality reflection model. (See "High-Quality Reflection Model" on page D-12 for more information.)

Lightscape uses the smoothness of the material, the angle of the incident light hitting it, and the index of refraction to determine the proportions of specular and diffuse reflection. (The angle of the light, of course, isn't a property of the material itself, but of the geometry of the object using the material.) The smoother the material or the closer the angle comes to grazing the material, the larger the component of specular reflection.

Note that in Lightscape, *smoothness* refers to smoothness only as it relates to light. Many surfaces that feel smooth (such as a piece of paper) are not smooth when magnified. Conversely, many surfaces that seem rough (such as an orange) have bumps, each of which is actually very smooth.

If the material is perfectly smooth, all light is reflected and transmitted specularly. The specular component is responsible for the clear reflections off shiny materials, as well as the images seen through transparent materials. A smooth material has more of a specular reflection when ray traced and has a sharper highlight when shadows are refined. (See Chapter 12, "Rendering.")

As a material becomes rougher, more of the energy is reflected and transmitted in the non-specular directions, until the material becomes very rough and most of the energy is reflected and transmitted diffusely (uniformly in all directions). Figure D-3 shows the proportion of diffuse reflection as a bubble of light.



Smooth (Glossy paint) **Medium rough (Semigloss paint)** **Very rough (Matte paint)**

Figure D-3. *Reflection and Transmission from Materials of Different Roughness*

For nonmetals, the color of the reflection at the interface is the same as the color of the original light. For metals, the reflection takes on the color of the metal. In addition, as the angle of the light gets closer to grazing, the reflection takes on less of this color and more of the color of the light. In general, a surface looks plastic if it has white highlights and metallic if it has colored highlights.

Scattering within the Material

For metals, all light is reflected off the material to air interface. Lightscape doesn't need to model light entering a metal.

For nonmetals, how much of the light reaches the interior of the material depends on the index of refraction of the material and on the angle at which the light hits the material.

The higher the refractive index, the less light goes into the interior of the material. If the index of refraction is 1, the material and the air appear the same to the light and all of the light is transmitted into the material. Most materials have an index of refraction between 1 and 1.5, the index for glass. By contrast, diamonds have an index of refraction of 2.5.

When the incident light hits the interface at a perpendicular angle, more light is transmitted into the material. When it just grazes the surface of the material, most of the light is reflected off the interface.

As light passes through the material, some wavelengths are absorbed more than others. As it hits small particles inside the material, the light is scattered in different directions. This is *subsurface scattering*.

Some of the scattered light leaves through the surface to air interface, some passes through the material, and some is absorbed in the material, as shown in Figure D-4.

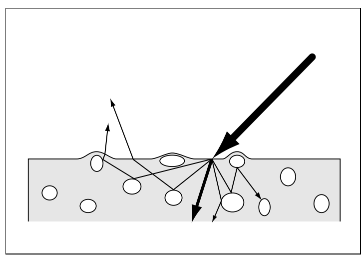


Figure D-4. *Subsurface Scattering*

Lightscape approximates the light that scatters out of the front of the surface as ideal diffuse—that is, uniform in all directions.

Transmitted Light

For metals, all light is reflected off the surface. Consequently, Lightscape doesn't need to model light going through a metal. For nonmetals, Lightscape uses the transparency of the material to determine how much light comes out the far side.

In reality, how much light and how it is transmitted out the far side is quite complicated, having the same components as does reflected light. Lightscape does not account for the diffuse components, only for the specular component. (Specular transmission, like specular reflection, looks like what it's transmitting and goes all in one direction.) Any diffuse aspects are lost.

As a result, in Lightscape you can accurately model transmission of light through a stained glass window, since that is primarily specular transmission. However, you can't accurately model transmission through tissue paper, since much of that transmission is diffuse.

Reflection Model for Radiosity

Lightscape uses this reflection model only for radiosity processing, not for displaying an image. This model has the following capabilities:

- Diffuse reflection
- Specular transmission
- Correct texture handling

For radiosity computations, Lightscape assumes the surface is an ideal diffuse (lambertian) surface. If the surface is transparent, light makes it through the surface and is attenuated by the color of the surface. This results in colored shadows being cast by transparent surfaces.

This reflection model has the following limitations:

- No refraction of transmitted light
- No specular reflection from shiny objects
- No diffuse transmission

Transparent surfaces do not refract the transmitted light. It is not possible with the radiosity process to create a lens and have it focus the transmitted light into a bright spot. It is also not possible to have a mirror reflect a bright spot of light onto another surface. (These effects are sometimes referred to as caustics.)

Reflection Model for OpenGL Display

The reflection model used during OpenGL display is very similar to the one used during radiosity processing. It has the following capabilities:

- Diffuse reflection
- Specular transmission

All surfaces are displayed as diffuse and anything seen through a transparent surface is attenuated by the color of the surface.

This reflection model has the following limitations:

- No refraction of transmitted light
- No specular reflection from shiny objects
- No diffuse transmission
- Incorrect display of intensity of textured surfaces

Transparent surfaces do not refract light. For example, there is no distortion when looking through a curved piece of glass.

OpenGL was designed to take advantage of hardware acceleration, causing two further limitations. The mapping from physical units to the limited range of values used by the hardware can only be done before applying the texture. Consequently, textures are not displayed at the correct intensity during interactive display. In general, this causes texture-mapped surfaces to appear too dark during OpenGL display.

The other limitation is that the OpenGL libraries use blending to handle transparency. For this reason, there can be significant loss of precision if several transparent surfaces overlap. These limitations are not significant if interactivity is desired.

Ray Tracing Reflection Models

Ray tracing works by tracing rays from the eye into the environment. This allows it to handle reflections and refraction through transparent surfaces. Ray tracing is described fully in Chapter 2, “The Basics,” of the *Lightscape Visualization System Getting Started* guide. There are two reflection models you can use with the ray tracer:

- OpenGL-compatible reflection model
- High-quality reflection model

OpenGL-Compatible Reflection Model

The ray tracer uses the OpenGL-compatible reflection model to create images that are very similar to the OpenGL images. It has the following capabilities:

- Diffuse reflection
- Specular transmission
- Simple specular reflection

One difference between this reflection model and the OpenGL reflection model for an environment containing only rough (diffuse) surfaces is that transparency is not limited by the precision problems caused by the blending in the OpenGL libraries.

In addition, if there are surfaces that are somewhat smooth, they are treated as reflective. Reflections are seen on these surfaces, but no highlights. The reflection model has the following limitations:

- No refraction of transmitted light
- Less accurate specular reflection from shiny objects
- No diffuse transmission
- Incorrect display of intensity of textured surfaces

Transmitted rays are not refracted because this reflection model ignores the index of refraction. Use this reflection model if you need to match a ray traced image with an image generated using the interactive OpenGL renderer.

High-Quality Reflection Model

This reflection model is based on some of the most physically accurate reflection models in the field of computer graphics. A physically valid model is crucial to achieving good results with a physically based reflection model.

Objects should not have holes that allow light into the inside of them. Transparent objects should have both a front and a back. This is not the same as making a transparent surface a two-sided surface—the two sides must be separated from each other. (See "Orientation" on page 5-16.)

This reflection model has the following capabilities:

- Diffuse reflection
- Specular reflection
- Highlights on nondiffuse surfaces
- Specular transmission with refraction
- Correct display of intensity on textured surfaces

This reflection model accounts for reflections and highlights from the interface between the surface and the air, as well as specular transmission and diffuse reflection. Refraction effects such as the distortion that comes from looking through wavy or angled glass are also present.

Highlights on surfaces are a function of both the viewing direction and the direction toward the luminaire. To render highlights, the Ray Trace Direct Illumination option must be turned on when ray tracing (see Chapter 12, “Rendering”) and the luminaires from which you want the highlights must have their Ray Trace Direct Illumination processing option turned on. (See Chapter 7, “Lighting”).

Textures are displayed correctly using this reflection model. This model does not handle diffuse transmission.

Appendix E

IES Standard File Format

This appendix briefly describes the IES LM-63-1991 standard file format for photometric data.

This appendix describes how you create a photometric data file in the IES format. Only the information relevant to Lightscape is described. For a complete description of the format, see *IES Standard File Format for Electronic Transfer of Photometric Data and Related Information,* prepared by the IES Computer Committee.

The luminous intensity distribution of a luminaire is measured at the nodes of a photometric web for a fixed set of horizontal and vertical angles. The poles of the web lie along the vertical axis, with the nadir corresponding to a vertical angle of zero degrees. The horizontal axis corresponds to a horizontal angle of zero degrees and is oriented parallel to the length of the luminaire. This type of photometric web is generated by a Type C goniometer and is the most popular in use in North America; other types of goniometry are supported by the IES standard file format, but are not discussed here.

The photometric data is stored in an ASCII file. Each line in the file must be less than 132 characters long and must be terminated by a carriage-return/line-feed character sequence. Longer lines can be continued by inserting a carriage return/line-feed character sequence. Each field in the file must begin on a new line and must appear exactly in the following sequence:

1. IESNA91
2. [TEST] the test report number of your data
3. [MANUFAC] the manufacturer of the luminaire
4. TILT=NONE
5. 1
6. The initial rated lumens for the lamp used in the test or -1 if absolute photometry is used and the intensity values do not depend on different lamp ratings.

7. A multiplying factor for all the candela values in the file. This makes it possible to easily scale all the candela values in the file when the measuring device operates in unusual units—for example, when you obtain the photometric values off a catalog using a ruler on a goniometric diagram. Normally the multiplying factor is 1.
8. The number of vertical angles in the photometric web.
9. The number of horizontal angles in the photometric web.
10. 1
11. The type of unit used to measure the dimensions of the luminous opening. Use 1 for feet or 2 for meters.
12. The width, length, and height of the luminous opening. Currently, Lightscape ignores these dimensions because you are free to associate a given luminous intensity distribution with any of the luminaire geometric entities supported by Lightscape. It is normally given as 0 0 0.
13. 1.0 1.0 0.0
14. The set of vertical angles, listed in increasing order. If the distribution lies completely in the bottom hemisphere, the first and last angles must be 0° and 90°, respectively. If the distribution lies completely in the top hemisphere, the first and last angles must be 90° and 180°, respectively. Otherwise, they must be 0° and 180°, respectively.
15. The set of horizontal angles, listed in increasing order. The first angle must be 0°. The last angle determines the degree of lateral symmetry displayed by the intensity distribution. If it is 0°, the distribution is axially symmetric. If it is 90°, the distribution is symmetric in each quadrant. If it is 180°, the distribution is symmetric about a vertical plane. If it is

greater than 180° and less than or equal to 360°, the distribution exhibits no lateral symmetries. All other values are invalid.

16. The set of candela values. First all the candela values corresponding to the first horizontal angle are listed, starting with the value corresponding to the smallest vertical angle and moving up the associated vertical plane. Then the candela values corresponding to the vertical plane through the second horizontal angle are listed, and so on until the last horizontal angle. Each vertical slice of values must start on a new line. Long lines may be broken between values as needed by following the instructions given earlier.

The following is an example of a photometric data file accepted by Lightscape:

```
IESNA91
[TEST] Simple demo intensity distribution
[MANUFAC] Lightscape Technologies, Inc.
TILT=NONE
1
-1
1
8
1
1
2
0.0 0.0 0.0
1.0 1.0 0.0
0.0 5.0 10.0 20.0 30.0 45.0 65.0 90.0
0.0
1000.0 1100.0 1300.0 1150.0 930.0 650.0
350.0 0.0
```

Appendix F

Lightscape File Types

This appendix describes the file types and filename extensions used in Lightscape.

Animation file (.la)
Stores the animation keyframes and motion data defined in the animation menus.

Block Library file (.blk)
Stores a collection of Lightscape blocks. The blocks may represent geometric objects or luminaires. You can import these blocks and luminaires into other Lightscape models.

Layer State file (.lay)
Stores the state (on, off, or current) of each layer in a model. You can load this file to reset the layers to the saved states.

Material Library file (.atr)
Stores a collection of Lightscape materials. You can import these materials into other Lightscape models.

Material Map file (.mm)
Stores a mapping (correspondence) between the 256 colors supported by DXF and Lightscape materials. You can specify a material map file when loading a DXF file. If you do so, Lightscape automatically assigns to surfaces the Lightscape material associated with their color index.

Parameters file (.df)
Stores the parameters that control the processing of a radiosity solution and the display of the results. You can load this file to reset the saved parameter values. Preparation and Solution files also preserve these parameter values.

Preparation file (.lp)
Stores all the basic geometric, material, and lighting data required to run a radiosity solution in ASCII format. The file structure is very similar to the DXF format, but is Lightscape proprietary.

Solution file (.ls)	Stores the radiosity solution of the model in binary format. This solution file contains the geometric information together with the photometric sample points (mesh) for each surface.
View file (.vw)	Stores the camera parameters for a specific view. You can load this file to reset the Graphic Window display to the saved view.

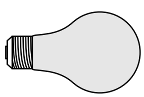
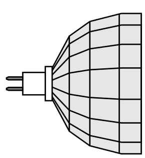
Appendix G

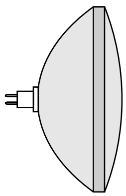
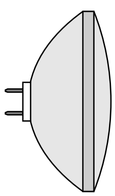
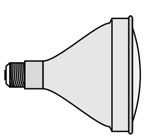
Common Lamp Values

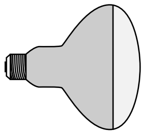
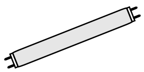
This appendix describes some common lamp values you can use as a guide for defining luminaires in Lightscape.

The following table lists some commonly used lamps. The information in the table is approximate. More precise photometric data for these lamps can be obtained from their manufacturers. This information is usually in an IES file.

You can approximate the intensity for a fluorescent luminaire with a diffusing panel by multiplying the number of lamps by the intensity of each lamp. For example a 2′ X 4′ luminaire may contain (4) 4′ tubes. This is equal to an intensity of 8,000 to 12,000 lumens. Again, precise measurements can be obtained using photometric data provided by the manufacturer, describing the luminous intensity distribution of the luminaire.

Lamps	Classification	Watts	Type	Intensity	Beam	Field
	**General Purpose**			**Candelas**		
	A-19/Med	60	Point	70		
	A-19/Med	75	Point	95		
	A-19/Med	100	Point	139		
	**M—16 Low Voltage**			**Candelas**		
	Narrow Beam	20	Spot	3300	6	12
	Narrow Beam	50	Spot	9150	12	25
	Medium Beam	50	Spot	3000	25	50
	Wide Beam	20	Spot	460	38	75
	Wide Beam	50	Spot	1500	38	75

Lamps	Classification	Watts	Type	Intensity	Beam	Field
	**Par—36 Low Voltage**			**Candelas**		
	Narrow Beam	25	Spot	4200	9	15
	Narrow Beam	50	Spot	8900	10	15
	Medium Beam	50	Spot	1300	30	60
	Wide Beam	25	Spot	250	36	75
	Wide Beam	50	Spot	600	39	75
	**Par—56 Line Voltage**			**Candelas**		
	Narrow Beam	300	Spot	68000	9	15
	Narrow Beam	500	Spot	95000	9	15
	Medium Beam	300	Spot	24000	18	36
	Medium Beam	500	Spot	47500	18	36
	Wide Beam	300	Spot	10000	30	60
	Wide Beam	500	Spot	18000	30	60
	**Par—38 Line Voltage**			**Candelas**		
	Narrow Beam	45	Spot	4700	14	28
	Narrow Beam	75	Spot	5200	12	25
	Narrow Beam	150	Spot	10500	14	28
	Medium Beam	45	Spot	1700	28	60
	Medium Beam	75	Spot	1860	30	60
	Medium Beam	150	Spot	4000	30	60

Lamps	Classification	Watts	Type	Intensity	Beam	Field
	**R—40 Line Voltage**			**Candelas**		
	Narrow Beam	150	Spot	5400	22	50
	Wide Beam	150	Spot	1040	76	130
	Wide Beam	300	Spot	1950	76	130
	**Fluorescent Tube—4H**			**Lumens**		
		32–40	Area	2000–3000		

Appendix H

Sample Luminaires

Lightscape provides examples of different types of luminaires that you can use as is or modify. This appendix describes the luminaires in lvs_lum.blk.

The luminaires that use IES intensity distribution files are not related to any particular manufacturer or fixture design. You can easily modify these fixtures to your own specifications but should save them under another luminaire name to preserve the original fixture. You can save the modified library to a new library by choosing Save As in the Luminaires Table context menu.

Lightscape provides DXF files of all the luminaires so that you can use these blocks in your original model if desired. If you import a DXF file, you can load a luminaire library to provide automatic substitution of the block.

Table H-1. *Ceiling Lamps*

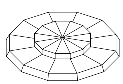
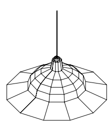
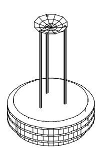
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_cl1.dxf	lvs_cl01	isotropic	95 cd	75W D65 White		
	lvs_cl2.dxf	lvs_cl02	spot	1050 cd	150W D65 White	80	138
	lvs_cl3.dxf	lvs_cl03	isotropic	140 cd	100W D65 White		

Table H-1. Ceiling Lamps (continued)

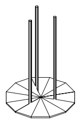
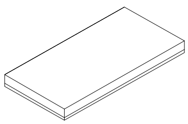
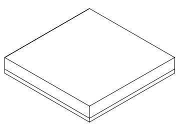
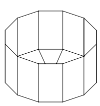
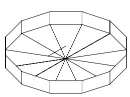
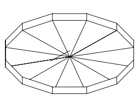
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_cl4.dxf	lvs_cl04	spot	140 cd	100W D65 White	150	180
	lvs_cl5.dxf	lvs_cl05	area-IES (parabolic)	8500 lm	30 W fluorescent		
	lvs_cl6.dxf	lvs_cl06	area-IES (parabolic)	4000 lm	30W fluorescent		
	lvs_cl7.dxf	lvs_cl07	spot	1100 cd	150W D65 White	76	130
	lvs_cl8.dxf	lvs_cl08	spot	4000 cd	300W halogen	30	90
	lvs_cl9.dxf	lvs_cl09	spot	12000 cd	300W D65 White	30	70

Table H-1. *Ceiling Lamps (continued)*

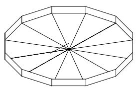
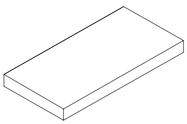
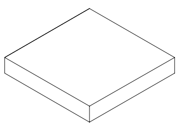
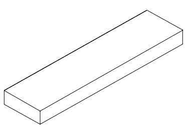
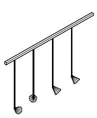
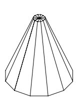
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_cl10.dxf	lvs_cl10	spot	6000 cd	150W incandescent	30	70
	lvs_cl11.dxf	lvs_cl11	area	6000 lm	40W daylight fluorescent		
	lvs_cl12.dxf	lvs_cl12	area	3000 lm	30W white fluorescent		
	lvs_cl13.dxf	lvs_cl13	area	3000 lm	30W daylight fluorescent		
	lvs_cl14.dxf	lvs_cl14	4x spot				
	lvs_cl14a.dxf	lvs_cl14a	spot	3000 cd	50W halogen	38	75

Table H-1. *Ceiling Lamps (continued)*

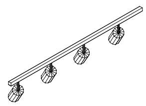
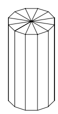
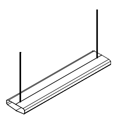
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_cl15.dxf	lvs_cl15	4x spot				
	lvs_cl15a.dxf	lvs_cl15a	spot	9150 cd	50W halogen	30	60
	lvs_cl16.dxf	lvs_cl16	area	6000 lm	40W cool white		
	lvs_17.dxf	lvs_cl17	isotropic	159 cd	100W D65 White		

Table H-2. *Floor Lamps*

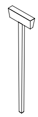
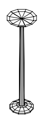
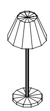
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_flr1.dxf	lvs_flr1	area	6000 lm	250W halogen		
	lvs_flr2.dxf	lvs_flr2	spot	2000 cd	D65 White	160	176
	lvs_flr3.dxf	lvs_flr3	point-IES	240 cd	100W incandescent		

Table H-3. *Exterior Lamps*

Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_st1a.dxf	lvs_st1a	4x spot	4x115000 cd	mercury vapor	60	120
	lvs_st2.dxf	lvs_st2	spot	25000 cd	low pressure sodium	70.2	135.2

Table H-3. *Exterior Lamps (continued)*

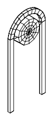
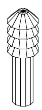
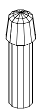
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_st3.dxf	lvs_st3	point	200 cd	warm white		
	lvs_st4.dxf	lvs_st4	spot	800 cd	deluxe cool white	120	169
	lvs_st5.dxf	lvs_st5	spot	1000 cd	D65 White	120	172

Table H-4. *Table Lamps*

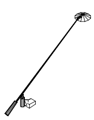
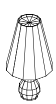
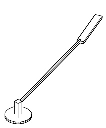
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_tbl1.dxf	lvs_tbl1	spot	1040 cd	50W halogen	40	120
	lvs_tbl2.dxf	lvs_tbl2	point-IES	250 cd	100W D65 White		
	lvs_tbl3.dxf	lvs_tbl3	spot	500 cd	daylight fluorescent	90	140

Table H-5. *Wall Lamps*

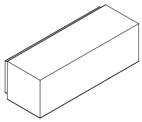
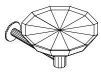
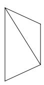
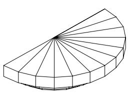
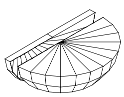
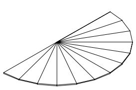
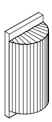
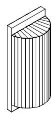
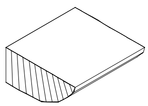
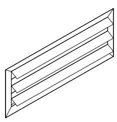
Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_wl1.dxf	lvs_wl1	isotropic	140 cd	D65 White		
	lvs_wl2.dxf	lvs_wl2	spot	1500 cd	D65 White	110	154
	lvs_wl3.dxf	lvs_wl3	spot	1600 cd	D65 White	104	160
	lvs_wl4.dxf	lvs_wl4	spot	1200 cd	D65 White	150	170
	lvs_wl5.dxf	lvs_wl5	spot	1500 cd	D65 White	150	170
	lvs_wl6.dxf	lvs_wl6	point	1500 cd	D65 White	150	170
	lvs_wl7.dxf	lvs_wl7	point-IES	400 cd	D65 White		

Table H-5. Wall Lamps (continued)

Preview	DXF name	LVS name	Type	Intensity	Source	Beam	Field
	lvs_wl8.dxf	lvs_wl8	point-IES	400 cd	D65 White		
	lvs_wl9.dxf	lvs_wl9	area	3000 lm	D65 White		
	lvs_wl10.dxf	lvs_wl10	area	3000 lm	incandescent		

Appendix I

References

This appendix provides you with a list of reading materials if you want more information about the technology used in Lightscape.

Ashdown, Ian. 1994. *Radiosity—A Programmer's Perspective*. John Wiley & Sons, Inc., New York, New York.

CIBSE (The Chartered Institution of Building Services Engineers), 1988. *CIBSE Standard File Format for the Electronic Transfer of Luminaire Photometric Data*, Technical Memoranda TM14.

Cohen, Michael F. and John R. Wallace. 1993. *Radiosity and Realistic Image Synthesis*. Academic Press Professional, Cambridge, Massachusetts.

Foley, John D., Andries van Dam, Steven K. Feiner, and John F. Hughes. 1990. *Computer Graphics, Principles and Practice*. Addison-Wesley Publishing Company, Reading, Massachusetts, 2nd Edition.

Illuminating Engineering Society of North America. 1991. *An Introduction to Light and Lighting*, IES ED-50-1991.

Illuminating Engineering Society of North America. 1984. *Calculation of Daylight Availability*, IES RP-21-1984.

Illuminating Engineering Society of North America. 1990. *Color and Illumination*, IES DG-1-1990.

Illuminating Engineering Society of North America. 1991. *IES Standard File Format for Electronic Transfer of Photometric Data and Related information*, IES LM-63-1991.

Illuminating Engineering Society of North America. 1986. *Nomenclature and Definitions for Illuminating Engineering*, ANSI/IES RP-16-1986.

Illuminating Engineering Society of North America. 1979. *Recommended Practice of Daylighting*, IES RP-5-1979.

Illuminating Engineering Society of North America. 1989. *IES Recommended Practice for the Lumen Method of Daylight Calculations*, IES RP-23-1989.

Glassner, Andrew S., Editor. 1989. *An Introduction to Ray Tracing*. Academic Press, Inc., San Diego, California.

Hall, Roy. 1989. *Illumination and Color in Computer Generated Imagery*. Springer-Verlag, New York, New York.

Rea, Mark S., Editor. 1993. Illuminating Engineering Society of North America. *Lighting Handbook, Reference and Application*, 8th Edition.

Sillion, François X and Claude Puech. 1994. *Radiosity & Global Illumination*. Morgan Kaufmann Publishers, Inc., San Francisco, California.

Glossary

This glossary contains industry-wide lighting terms as well as terms used in the Lightscape Visualization System Manuals.

3DS file format
The file format standard used by Autodesk's 3D Studio application to store the description of the scene.

accumulation buffer
An offscreen buffer where several images are summed (accumulated). The resulting image is scaled and displayed. Lightscape uses the accumulation buffer for antialiasing.

adaptation
The process by which the eye adjusts to the intensity levels and colors in a scene.

adaptive subdivision
The process of subdividing a radiosity mesh into smaller mesh elements during the computation of the illumination from a source to a receiving surface.

ambient lighting
A constant amount of light added to every surface in an environment as an approximation of the effects of undistributed energy.

antialiasing
The process of reducing artifacts caused by undersampling small, sharp details in an image. The elimination of "jaggies."

area light
A polygonal light source of finite area.

AS units
American System of Units.

beam angle
The angle of the spotlight aim axis at which the luminous intensity drops to 50% of its maximum. See *spotlight distribution*.

blending
A rendering technique in which two colors are combined into one, usually by linear interpolation. Lightscape uses OpenGL blending to render partially transparent objects.

bump mapping
Randomly displacing the surface normal on a surface to make the surface appear bumpy.

candela (cd) The SI unit for luminous intensity.

checkpoint A solution file containing a snapshot of the lighting simulation at a given instant in time. Checkpoints can be saved at regular intervals during the lighting simulation to ensure that the result of the computation are not lost in case of abnormal termination.

chromatic adaptation The process by which the eye becomes accustomed to strong color shifts in an image, causing them to appear less severe.

CIBSE file format The standard file format adopted by the Chartered Institution of Building Services Engineers for the electronic transfer of luminaire photometric data—used in Great Britain.

clear sky See *sky conditions*.

cloudy sky See *sky conditions*.

color The sensation produced by light entering the eye and being perceived by the brain.

color matching The process of mixing a set of colored lights to create a color that appears exactly like a test color.

color space A representation for color. All colors are defined with respect to some particular color space—typically HSV and RGB.

contrast The relative difference in illumination between two adjacent regions.

criterion rating The fraction of the area of a surface that satisfies or exceeds a specific criterion.

culling
Lightscape provides control over backface culling, which discards from the display all surfaces that face away from the viewer. View frustum culling, which is always performed when rendering the model, quickly discards all surfaces totally outside the field of view (view frustum).

daylight
Availability of the luminous flux from the sun and sky for a given time, location, and sky conditions.

diffuse distribution
An axially symmetric luminous intensity distribution such that the emitted light varies as the cosine of the emission angle, measured from the axis of the distribution.

diffuse reflection
Incident light reflected by a surface so that the reflected luminance is the same in all directions.

direct illumination
Illumination resulting from light reaching a surface directly from a direct light source.

direct light
A self-luminous light source.

dolly
A camera motion toward or away from the focus point.

double-buffer
A rendering technique to provide smooth interactive display. Lightscape renders into the hidden "back" buffer while displaying the "front" buffer. When finished, the buffers are swapped so that the back buffer becomes the (visible) front buffer.

DXF file format
The file format standard used by Autodesk's AutoCAD package for exchange of drawing data among CAD applications. Currently the de facto industry standard.

dynamic range — The ratio of the highest intensity in an image or environment to the lowest intensity. The dynamic range of real scenes is very large. The dynamic range that most display devices are capable of reproducing is low.

field angle — The angle of the spotlight aim axis at which the luminous intensity drops to 0. See *spotlight distribution*.

filter — A device that changes the spectral composition of light transmitted through it.

footcandle (fc) — The AS unit of illuminance, equal to 1 lumen per square foot.

form factor — The fraction of the energy leaving a light source that actually arrives at a receiving surface.

gamma — The nonlinear change in light intensity caused by a particular display device. Gamma is often used as the process of compensation for this nonlinearity.

global illumination — The effect of all possible types of light transport (direct illumination, indirect illumination) throughout an environment.

GON file format — See *TBT file format*.

hue — One of three parameters in the HSV (Hue Saturation Value) color space. It describes the dominant wavelength of the color such as red, yellow, or green.

identity transformation — A function that transforms a point to itself. A geometric transformation that has no effect.

IES file format The standard file format adopted by the Illuminating Engineering Society for the electronic transfer of photometric data and related information—used in North America.

illuminance The luminous flux incident on a surface of unit area.

Illuminating Engineering Society The technical authority for the illumination field in North America.

indirect illumination Illumination that results from light reaching a surface after being reflected by one or more other surfaces in the environment.

indirect light A surface that reflects light into the environment and is thus acting as a light source.

initiation The Lightscape operation that converts the initial description of a model into data structures suitable for the radiosity processing.

intensity magnitude The intensity of a light in photometric units. This plus a color can be used to determine the radiometric quantities needed for the simulation.

intensity mapping A type of procedural texture mapping used to vary the intensity over a surface to make it appear less perfect and more like a real surface.

interlacing A technique of displaying every other scan line when updating a video image. First the even scan lines are displayed, then the odd ones. This allows the entire screen to be updated only every thirtieth of a second rather than every sixtieth.

interreflection — The reflection of light between two surfaces in the environment.

Inventor file format — The file format used by Silicon Graphics Open Inventor to describe the 3D scene.

inverse square law — The law stating that the illuminance measured at a point on a surface is directly proportional to the luminous intensity of a point light source in the direction of the receiving point and inversely proportional to the square of the distance between the source and the point.

isotropic distribution — A constant luminous intensity distribution.

jittering — A small, random change in a position or direction used to prevent aliasing artifacts.

lambertian surface — A surface that reflects the same luminance in all directions. See *diffuse reflection*.

lamp — An artificial source of light. Normally used to denote a light bulb.

level of detail — A technique to improve rendering performance by dropping detail from complex objects that only cover a small area on the screen. Because the object appears small, any detail is unlikely to be visible anyway.

light — Radiant energy capable of producing a visual sensation in a human observer.

linear light — A light source that can be approximated as a straight line segment.

LTLI file format	The luminaire photometric file format implemented by the Danish Illuminating Laboratory, Lysteknisk Laboratorium, in the early 1970s—used in the Scandinavian countries.
lumen	The SI unit of luminous flux.
luminaire	A light fixture complete with one or more lamps and housing.
luminance	The photometric quantity that describes light leaving a surface in a particular direction.
luminance contrast	The relative difference between luminance values of adjacent regions.
luminous exitance	The luminous flux leaving a surface of unit area.
luminous flux	The quantity of light energy per unit time arriving, leaving, or going through a surface.
luminous intensity	The light energy per unit time emitted by a point source in a particular direction.
luminous intensity distribution	The function that describes the directional distribution of luminous intensity of a point source.
lux	The SI unit of illuminance, equal to 1 lumen per square meter.
magnify	A filtering operation used by texture mapping techniques to determine the color of an area that covers less than one pixel in image texture space.
material	The set of parameters that are used by the reflection model to determine how light interacts with a surface.

material properties — See *material*.

matte surface — A surface that scatters light uniformly in all directions. It appears equally bright at any angle.

mesh — The data structure that describes the light distribution over a receiving surface. It breaks down the original surface into a set of smaller polygonal pieces called *mesh elements*. The corners of these elements, called *mesh vertices,* are shared among adjacent elements and are used to store the illumination data collected during the lighting simulation.

minimize — A filtering operation used by texture mapping techniques to determine the color of an area that covers more than one pixel in image texture space.

nanometer (nm) — One billionth of a meter. A common unit for describing the wavelength of light.

normal — See *surface normal*.

OpenGL — An industry-standard application programming interface for drawing 3D graphics.

orbit — A camera motion around the focus point, keeping the same distance.

orientation — See *surface orientation*.

pan — A camera motion parallel to the screen. The focus point moves the same amount in the same direction as the camera.

partly cloudy sky — See *sky conditions*.

penumbra — The transition region at the boundary of a shadow where light shining from a source partly reaches the receiving surface and is partly occluded by some other obstacle in the environment.

photometric web — A regular grid of luminous intensity samples that describes the luminous intensity distribution of a light source.

photometry — The measurement of light taking into account the psychophysical aspects of the human eye/brain perceptual system.

point light — A light source so small compared to its distance from the observer or receiving surface that its radiation can be assumed to come from a dimensionless luminous point.

procedural texture mapping — A more general form of texture mapping that is usually not based on images and that can affect more than just the material color.

progressive refinement — A technique for computing radiosity solutions that starts with the direct illumination and then computes more and more of the indirect illumination until the solution converges.

radiosity — A technique for solving the global illumination problem for diffuse environments.

ray offset — The displacement measured from the origin of a shadow ray. Intersections between a surface and a shadow ray closer to the origin than the ray offset amount are discarded.

ray tracing — A way of computing an image based on tracing paths of light from the eye back to the luminaires.

reflectance The ratio of the luminous flux reflected off a surface to the luminous flux incident on it.

reflection Light incident on one side of a surface leaving it from the same side.

reflection model A description of how light interacts with a surface.

refractive index Ratio of the speed of light in a vacuum to the speed of light in a material. Determines the amount of light reflected and transmitted at the interface between them.

refraction The bending of light rays as they pass from one material, such as air, into another material, such as glass.

refraction index See *index of refraction*.

rotate The rotation of the camera about its center.

saturation (of a color) One of three parameters in the HSV (Hue Saturation Value) color space. It describes how pure the color is. A color with a low saturation is very close to gray.

scroll A camera motion parallel to the screen. In an orthographic view, the focus point moves with the camera. In perspective view, the focus point remains the same but the screen is tilted with respect to the view direction.

self-emitted luminance Luminance emitted from a surface that is not due to reflection of incoming light off that surface.

shadow ray — A line cast between a point on a light source and a point on a receiving surface to determine the possible presence of occluders that would prevent light from the source from reaching the receiving surface.

SI units — International System of Units.

sky conditions — The conditions of the sky at a given time and location; described as the fraction of the sky covered by clouds or as clear, partly cloudy, or cloudy sky.

sky light — Light energy from the sun that reaches the scene after scattering through the atmosphere.

smoothing angle — The angular threshold used during automatic computation of vertex normals. Polygons incident on a vertex share a vertex normal only if their respective surface normals form an angle that is less than the given threshold.

soft shadow — A shadow with an area of penumbra along its boundary.

solar altitude — The angular distance from the plane of the horizon to the sun.

solar azimuth — The angular distance from true south to the vertical plane that contains the sun.

source accuracy — The accuracy of the calculation that computes the light contribution from a source to a receiving surface.

spectral curve — A representation of a spectrum that gives the intensity of light at each wavelength in the visible spectrum.

spectral quantity — Any quantity that varies with the wavelength of light.

spectrum

See *visible spectrum*.

specular reflection

A perfect reflection off a surface in the mirror direction. A mirror has a very large amount of specular reflection.

specular transmission

An ideal transmission of light through the surface in the direction determined by the angle at which the light strikes the surface and the index of refraction of the surface.

spotlight distribution

A luminous intensity distribution that is axially symmetric, that has maximum luminous intensity along its axis, and whose intensity drops smoothly away from this axis. The angle off the axis at which the luminous intensity drops to 50 percent of its maximum is called the *beam angle*. The angle off the axis at which the luminous intensity is cut off to zero is called *field angle*.

sunlight

Direct illumination from the sun.

surface normal

The direction that is perpendicular to a surface at a point on the surface. Sometimes surface normal is simply referred to as "normal."

surface orientation

The direction of the front of the surface as determined by the surface normal. The front of the surface is illuminated by the lights, the back is not.

TBT file format

The file format used by Integra's Turbo Beam Tracing to describe its light sources and associated photometric data—used in Japan. Also referred to as GON file format.

tessellation

The process of subdividing a surface into smaller pieces. It is often used to approximate a curved surface with a set of planar polygons.

texture filter methods Ways of blurring a texture as it is applied to a surface so that aliasing artifacts do not appear on the texture-mapped objects.

texture mapping The changing of material properties such as color based on an image or procedure.

transmission Light incident on one side of a surface leaving it from the opposite side.

transmittance The ratio of the luminous flux transmitted by a surface to the luminous flux incident on it.

transparency The property of a material that determines how much light is transmitted through the surface.

value One of three parameters in the HSV (Hue Saturation Value) color space. It describes how dark or light the color is.

view dependence In a view-dependent global illumination algorithm, moving the camera requires recomputing most of the solution.

view frustum The region of 3D space visible from a given camera or observer. This region is a rectangular pyramid with the apex at the observer's eye. The near and far clipping planes cap the top and bottom of the pyramid respectively.

view independence In a view-independent global illumination algorithm, the camera can be moved with minimal computation.

visibility The process of determining if there are any objects between two points in an environment. Used by the radiosity system to determine how much light gets from one surface to another and by the ray tracer to determine whether a point on a surfaces is in the shadow of a luminaire.

visible spectrum
The range of electromagnetic radiation (380 nm to 780 nm) to which the eye is sensitive. Often referred to as light.

white point
The brightest white a monitor is capable of. The white points of different monitors can vary widely in color.

whiteness constancy
The tendency of the eye to perceive white surfaces as white even under lights of different colors.

workplane
A surface in the scene used to collect illumination samples for lighting analysis.

zoom
A change in the camera's field of view (or focal length). The camera does not move.

Index

Symbols

Numerics

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W