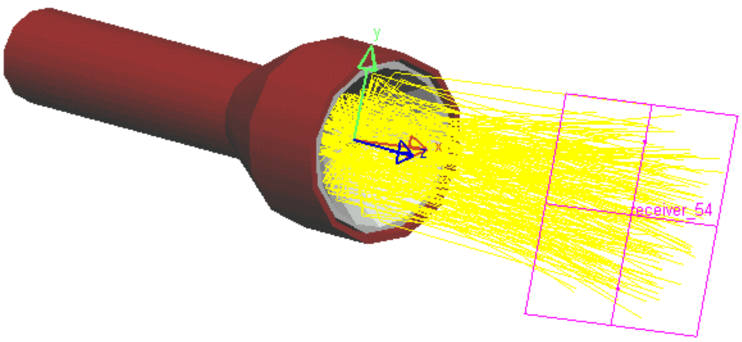
Chapter 4 Starting from Scratch: Build a Virtual Flashlight

This chapter describes the steps needed to create and analyze a model of a wide-angle flashlight. You will start with a smooth parabolic reflector and a point source, introduce a detailed lamp source, and finally use a supplied utility to create a faceted reflector tailored for the desired pattern (i.e., a uniform beam over a 100 mm disk at a distance of 300 mm from the source). You will also learn how to use Boolean operations to build a mechanical model of the flashlight body.

Contents

What is a Wide-Angle Flashlight?

In this example, you will create a virtual prototype of a wide-angle flashlight, using an incandescent lamp and a reflector. What is a wide-angle flashlight? A typical flashlight tries to produce a fairly concentrated beam of light, without much concern for uniformity. A point source at the focus of a perfect parabolic reflector would produce a collimated beam the same size as the reflector, with a Gaussian illuminance profile. In this design example, the flashlight will produce fairly uniform illuminance over a 100 mm diameter circle at a distance of 300 mm from the source.



Because you will be starting from scratch, this example has more steps than the light pipe example. Be sure to follow all the steps and save your work frequently (**File > Save**) so you can get back on track if you should take a wrong turn.

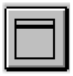
Start with the Reflector

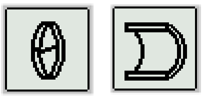
Optically speaking, a flashlight is just a light source and a reflector, so you'll start with the essentials, and then add the shiny metallic flashlight body at the end, mainly for show.

Use the Place Reflector Buttons

First, you'll create a smooth, parabolic reflector.

Note: Be sure that you have set the default preferences as described in Chapter 2.

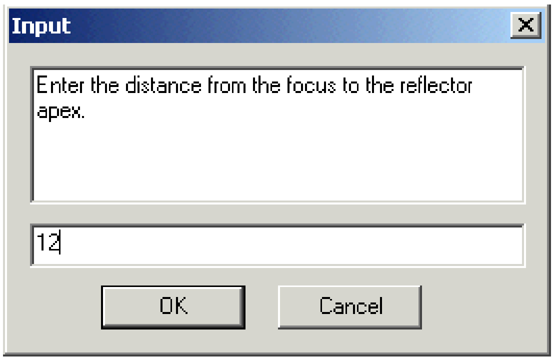
1. Start *LightTools*.
2. With the Console window active, start a new model in a 3D Design view (**File > New Model > 3D Design**).
3. With the Right Side View active (indicated by a red border), click the **One Pane** button on the toolbar: 
4. On the command panel, click the following buttons to display the **Place Reflector** buttons:



The **Place Reflector** buttons are different from other command buttons because they each launch a macro. The only functional difference you should notice, however, is that the prompts for these commands are displayed in small dialog boxes, rather than above the command line.

5. Click the **Spun Parabola** button: 

   The first of three prompting dialog boxes is displayed, shown in the following figure.



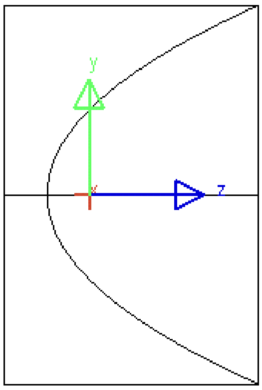
6. Key in **12** for the distance from the focus to the reflector apex and click OK.

   The next dialog box prompts you to enter either the reflector diameter or an input code (-1) to indicate that you want to define the depth of the reflector instead.

7. Key in **-1** and click OK.

8. In the third dialog box, key in **60** for the depth of the parabola and click OK.

   The reflector is displayed, as shown in the following figure.



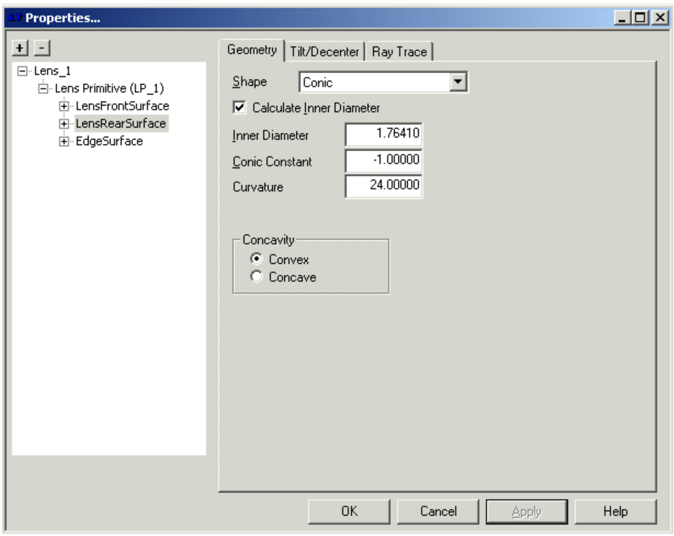
Tip: If you click on the Console window, which logs all events in your session, you can see that the macro issues the ***Wireframe*** rendering and ***FitAll*** scaling commands.

Curve the Back Surface

This step is not optically important, but it makes the reflector look a little more realistic. The reflector has a parabolic front surface (a conic surface with a conic constant of -1), and a flat rear surface. It's easy to give the rear surface the same profile as the front.

1. Select the reflector. You can do this in the design view or the System Navigator.
   – In the design view, click anywhere on the reflector to select it.
   – In the System Navigator window, select **refl**.
2. Open the Properties dialog box by right-clicking and selecting **Properties** on the shortcut menu.
3. In the navigation tree of the dialog box, click on LensFrontSurface.

   On the Geometry tab, you can see that the surface shape is Conic, with Conic Constant of -1 and (vertex) Radius of 24 mm.
4. Click on LensRearSurface, and change the following parameters, as shown in the following figure:



a. For the Shape, select **Conic** from the drop-down list, and click Apply.

b. For the Conic Constant, key **-1**.

c. For the Radius, key in **24.**

The Concavity option is automatically set to Convex.

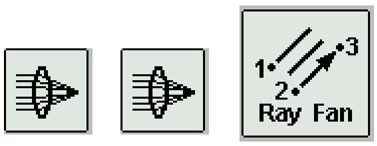
5. Click OK.

The rear surface is redrawn.

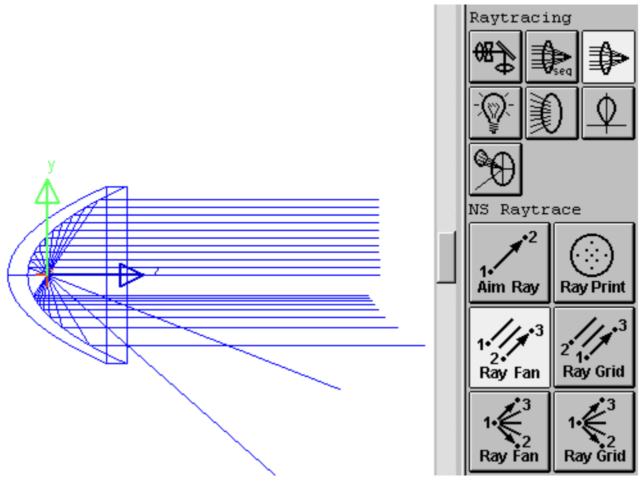
Trace Some Rays

How can you show that this really is a parabolic reflector? If it is, a bundle of point-and-shoot rays traced parallel to the Z-axis should focus to a single point, which, because of the design of the reflector macro, is located at (0,0,0).

1. Pan and zoom the view so that you have about 300 mm of space to the right of the reflector. To do this, use the Shift and Control keys with the right mouse button, with the cursor coordinates in the lower-right corner as a guide.
2. Select the ray fan button (***NSFanAim***):

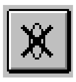


3. Click the points needed to define the ray fan (locations do not have to be exact):
   - Click 1: to the right of the reflector, at approximately Z = **200**, Y = **0**, for example.
   - Click 2: above the first point, at approximately Z = **200**, Y = **45**, to define the width of a half-fan of parallel rays.
   - Click 3: The third click defines the direction. Use a snap operation to make it precise. Right-click anywhere and select **Snap > 90 Degrees**; then, move your cursor to the left to aim the bundle of rays at the reflector and click to trace the rays.



Note that the rays focus precisely at (0,0,0), as expected for an on-axis parallel ray bundle with a parabolic reflector.

If you'd like, select **Edit > Undo** and try creating the fan of rays without using the snap feature to see how a small off-axis angle can lead to big aberrations.

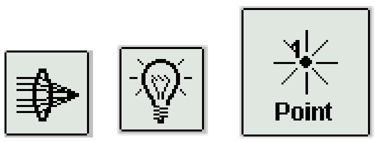
4. Delete the fan of rays. To do this, click on any part of the fan to select it; then, right-click and select **Delete** on the shortcut menu, or click the **Delete** toolbar button, shown here: 

Point Source and Receiver

Place a Point Source

If a parallel fan of rays focuses at (0,0,0), then a point source at this point should produce *collimated light* (parallel rays) after reflection.

1. Click the Point source button (***PtSource***):

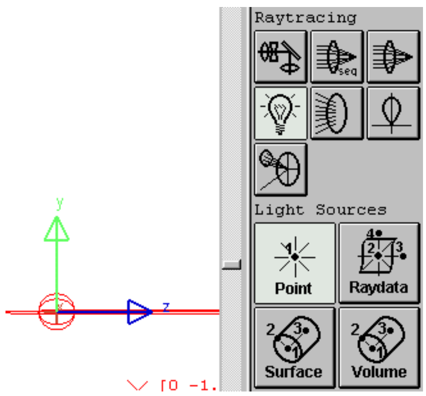


You could zoom in and click at the point (0,0,0), but since you know the coordinates, it's easier to enter them on the command line, following the command ***PtSource***, which is already displayed.

2. In the command line, key in the string **xyz 0,0,0** with spaces and commas as shown, followed by a space.

Note: *LightTools* starts evaluating the command when you input the final space. You do not need to press the <Enter> key.

A 1-lumen photometric point source is created, shown below, after zooming in.



Enlarging (zooming in) for a specific area is described in *The Zoom Tool* on page 17.

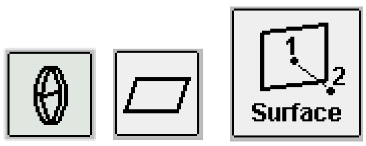
The wireframe sphere represents the aim sphere of the point source. It was created using the default source properties, which are acceptable for now. If you would like to see the properties, click to select the source, then right-click and select **Properties**.

Note: The aim sphere for the source determines the directions in which the source can emit. By default, it's a full sphere. For more details about aim spheres, see *Defining Aim Entities for Importance Sampling* on page 38 in the *LightTools Illumination Module User's Guide*. To display this book, select **Help > Document Library > Illumination Module User's Guide** from the main menu.

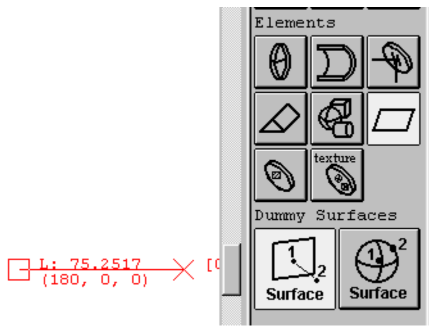
Create a Dummy Surface

Illumination analysis requires at least one receiver, where rays are collected, displayed, and analyzed. A surface receiver can be attached to a surface of any 3D object. (Far-field receivers for angular calculations are not attached to a surface, but it's not necessary to perform angular analysis right now). In many cases, a dummy surface or a flat (no-power) "lens" element is created to hold a receiver, but if there is a convenient real object, you can attach a receiver there. In this case, the goal is a certain illuminance at 300 mm from the source. So, you need to define a square dummy surface with a full-width of 150 mm at a distance Z=300 mm.

1. Select the dummy surface button (***DummyPlane***):



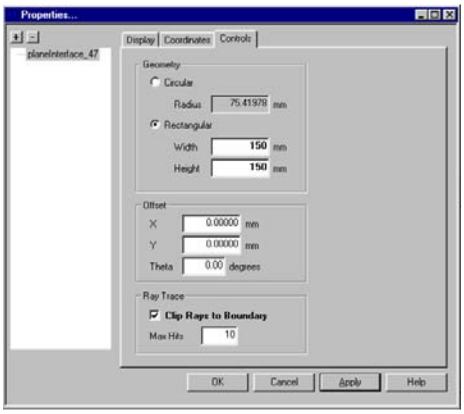
2. Since you know the required coordinates, key in the string **xyz 0,0,300** with spaces and commas as shown, followed by a space.
3. Right-click and select **Snap** > **Z-axis**.
4. Using the rubber band guide (L value for length), click about **75** mm to the left of the starting point.



This point defines the dummy's surface normal vector, and the length of the vector sets the half-size of the dummy. By default, a square dummy plane is created. You can use the Properties dialog box to adjust its (full) width and height to exactly 150 mm.

5. Click to select the new dummy plane, right-click, select **Properties**, and click the Controls tab.

6. Change Width and Height to **150** mm each, as shown.



7. Under Ray Trace, make sure that the box for **Clip Rays to Boundary** is checked.

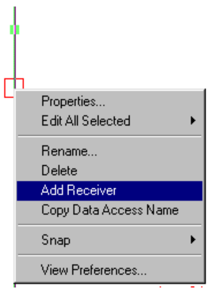
   This is useful when a dummy surface is used to hold a receiver, since you are only interested in the light collected by the receiver. (By default, the receiver is the same size as the dummy surface it is attached to). Note that the default name for the dummy surface is PlaneInterface. You can rename it if you like.

8. Click OK to apply these changes.

Place the Receiver

Now you can attach a surface receiver to the dummy surface. Surface receivers are displayed as a triangular symbol. A right-click menu is the easiest way to do this.

1. Click anywhere on the dummy surface to select it, then right-click and select **Add Receiver**.



A receiver symbol and label are displayed, shown above (after zooming in).

Tip: You can also add a receiver to a surface from the System Navigator. To do this, select the PlaneInterface object listed under the Components heading in the System Navigator, right-click, and select **Add Receiver** on the shortcut menu.

Understanding Surface Receivers and Far Field Receivers

LightTools offers two types of receivers, *surface* and *far field*. Surface receivers can measure illuminance and intensity on a given surface. Therefore, a surface receiver always exists on a given surface and only measures the light that interacts with the surface it is attached to. Surface receivers are always located at a finite distance from the source.

The far field receivers are located at infinity. Therefore, they measure only intensity, and not illuminance. The illuminance on a given surface is given by the following equation:

$$E = \frac{I}{D^2}$$

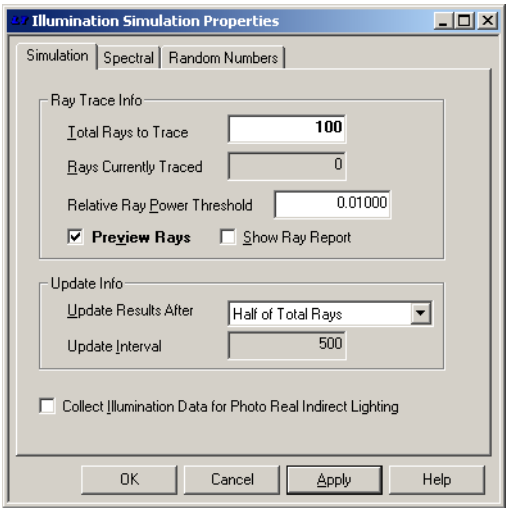
where

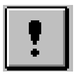
- E – illuminance on a given surface (lumens per unit area)
- I – intensity of the source in the given direction (flux per unit solid angle)
- D – the distance between the source and the receiver

From this, it is clear that the illuminance is inversely proportional to the square of the distance between the source and the receiver. This is known as the *inverse square law*. When the distance becomes infinitely large, the illuminance approaches zero. The intensity, however, is independent of the distance and, therefore, remains unchanged. Most light sources follow the *inverse square law*. However, when the output beam is highly collimated (spot lamps, laser, etc.), they tend to deviate from this behavior.

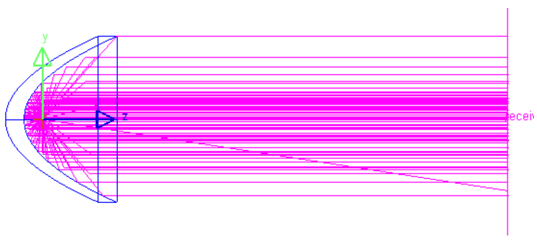
Run a Quick Illumination Simulation

1. Select **Illumination > Simulation Info** to display the Illumination Simulation Properties dialog box.
2. Change Total Rays to Trace to **100**, check the Preview Rays box, and click OK.



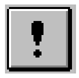
3. Select the Start Simulation button  on the toolbar.

   The results should resemble the figure below, although the colors in your model may be different.



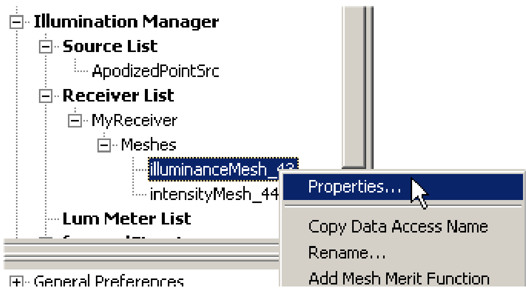
Trace More Rays and Check Results in the LumViewer

The preview rays are collimated, as expected (although you may see some non-parallel direct-radiation rays that don't hit the reflector). Now, you can increase the number of rays and look at an analysis.

1. Select **Illumination > Simulation Info**.
2. Change Total Rays to Trace to **10000** (ten thousand), *un*check the Preview Rays box, and click OK.
3. Click the  button on the toolbar.

   The interrupt and progress dialog boxes are displayed. It will take a few seconds to trace, depending on your CPU speed.
4. Select **Illumination > Illuminance Display > Scatter Chart**.

   The ray density looks good. You could open an illuminance raster chart as well, as in the light pipe chapter, but you can see the properties of the ray data directly from the System Navigator.
5. In the System Navigator, expand the Illumination Manager items as shown in the following figure.



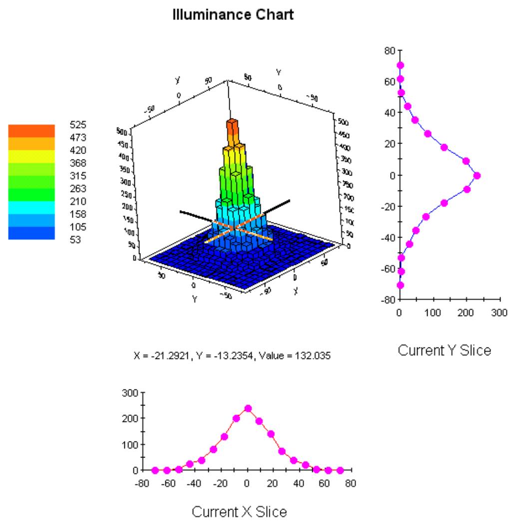
6. Right-click on the illuminanceMesh_*n* item, and select **Properties**.

   The Properties tab should show an auto-sized mesh of 17 x 17 bins. The Results tab should show a peak error of under 5% (the 17 x 17 was chosen to provide this) and total flux of about 0.85 Lumens.

   You can use the LumViewer feature to see how the illuminance distribution looks. LumViewer is interactive and can only display one receiver at a time, so you must select the receiver of interest (the only one, in this case).
7. Click on the receiver to select it. You can either select the name of the receiver in the System Navigator or select the receiver symbol in the design view.

8. Select **Illumination > Illuminance Display > LumViewer**.

   The LumViewer opens an interactive Illuminance Chart, as shown below. This view has been rotated and modified as described in the following steps.



9. Right-click anywhere on the chart to display the Chart Properties dialog box.
10. Check the Grid Lines box on the View tab of the Chart Properties dialog box.

    Note that this tab also gives you graphic smoothing, contour plots, and other controls.
11. Click OK.
12. Place the cursor over the 3D chart, press and hold the right mouse button, and move the mouse to rotate the view.
13. Drag with the left mouse button to move the chart cursor that selects the X and Y slices that are displayed.

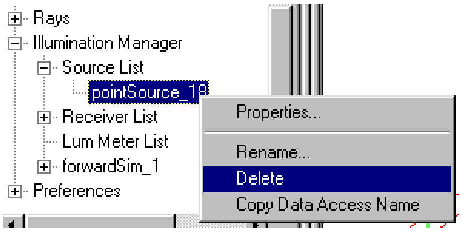
    The illuminance cross-section plots look like Gaussian curves, which they are for a point source at the focus of a parabolic mirror.

Adding a Detailed Source Model

You could experiment with the position of the point source, but you wouldn't be able to get a very uniform distribution. Instead, this section describes how you can replace the point source with a pre-defined lamp model that has been saved as a library element. This lamp model was created for this example and is similar in geometry and illumination properties to actual flashlight bulbs used with larger (3 D-cell) flashlights. It includes a toroidal filament for the actual source, as well as objects to define the bulb and base of the lamp.

Delete the Point Source

1. In the System Navigator, locate and expand the Illumination Manager section as shown below.

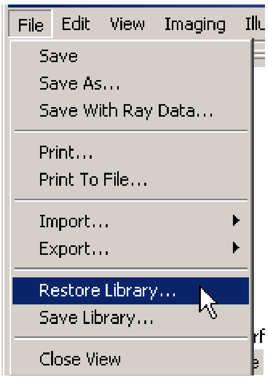


2. Right-click on the pointSource_*n* item under Sources, and select **Delete**.

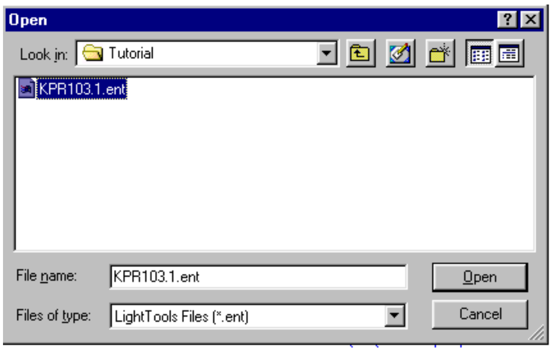
Insert the Library Element

Most sources are placed from the source palette, but since this source has been saved as a multi-object library entity (an .ent file), you load it using the **File > Restore Library** menu.

1. Select **File > Restore Library**.

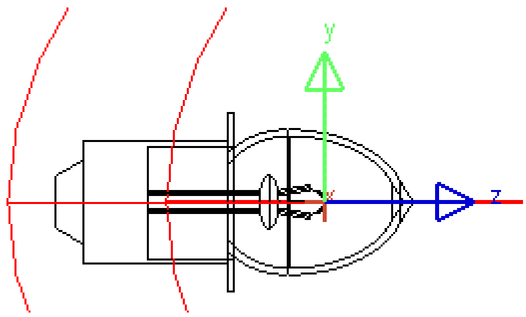


2. In the Open dialog box, navigate to the \Tutorial folder in the *LightTools* installation directory. Click on the file name KPR103.1.ent and click Open.



Watch the command line text prompts (noted below) for the following steps, entering the values or clicking as required.

3. *Enter scale factor for element:* Enter **1** for the scale factor, followed by a space.
4. *Indicate position:* Enter **xyz 0,0,0**, followed by a space.
5. *Indicate Z axis direction:* Right-click and select **Snap** > **Z-axis**, then click anywhere to the right of the source position.
6. *Indicate Y axis direction:* Right-click and select **Snap** > **90 degrees**, then click anywhere above the point just entered.



LightTools .ent files are similar to .lts files, in that they can contain multiple objects of various types (e.g., sources, optical parts, mechanical parts), but they are meant to be inserted into an existing model. Inserting the lamp file takes a few seconds.

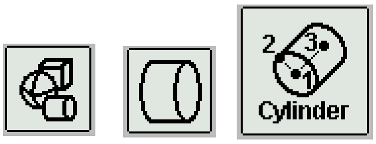
Make a Hole for the Lamp Base

Right now, the lamp base occupies the same space as a portion of the reflector. *LightTools* has no problem with this, but it's not physically correct. It's easy to use a Boolean operation to “drill a hole” in the lamp base. Boolean operations are used to combine simple 3D objects into more complex objects. To make the hole, you will create a mechanical cylinder and subtract it from the mirror.

You can be as precise as necessary with dimensions, but in this case, you can visually estimate the size.

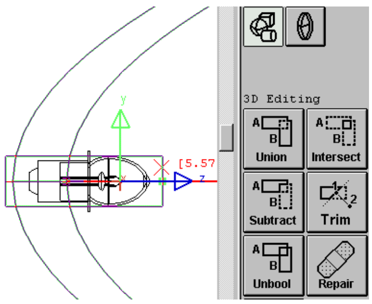
Before you begin these steps, you may want to zoom in on the lightbulb because you will be drawing a cylinder around it. And you can use the **Y-Z Plane** button to adjust the view if you had rotated it earlier.

1. Select the mechanical Cylinder button (***Mcylinder***):

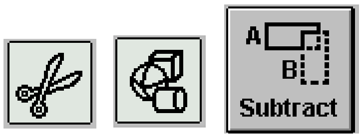


2. Right-click anywhere in the design view and select **Snap > Z-axis**.
3. Click somewhere to the right of the front surface of the reflector (over the lamp is OK, around z = **0** is fine).
4. Use the lamp base and displayed radius (R, in a red font on the screen) as a guide and click to define the radius of the cylinder. A value of **6** for example, is bigger than the lamp base but smaller than the flange.
5. Right-click and select **Snap > Z-axis**, then click somewhere to the *left* of the *rear* surface of the reflector (a Z of about **-25** will work).

   The cylinder is created, as shown in the following figure.

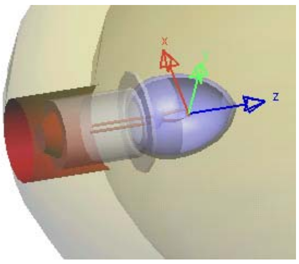


6. In the System Navigator, select the **refl** object. Then, control-click on the name of the cylinder to add it to the selection. **The order in which you do this is important.**
7. Select the Boolean **Subtract** button:



Note that this command operates immediately on the selected objects; no input points or data are required.

Try switching to translucent rendering mode and rotating the view with the right mouse button to see the hole. By default, mechanical objects are rendered in red, optical absorbers are yellow, reflectors are silver, and refracting objects are translucent blue. In the picture below, absorbing and reflecting objects have been made translucent for illustration. They are opaque by default.

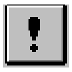


You can set these properties on the Surface Color and Surface Rendering tabs on the Preferences dialog box. To display this dialog box, right-click anywhere in the design view and select **View Preferences** on the shortcut menu.

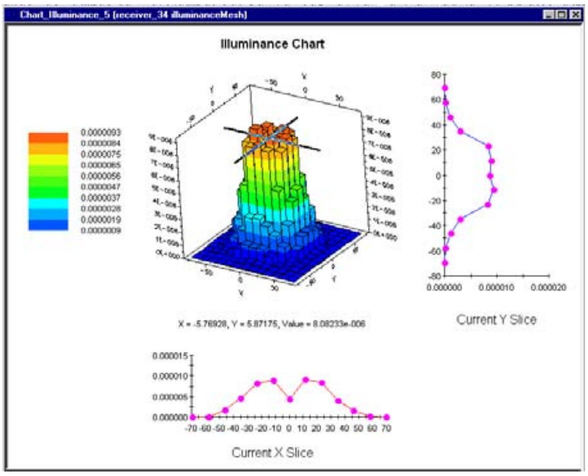
Tip: Commands are useful shortcuts for some tasks. Try typing ***Wireframe*** (or just ***Wir***) and ***Translucent*** (or just ***Tra***) in the command line of the design view to change the rendering mode. ***Fit*** is the same command as is used by the Fit toolbar button; it puts all of the components into view. The **View** menu is always available for these tasks as well.

Re-run the Simulation

Now you have a more complex source. Re-running the simulation run is simple. If you would like, you can first run a small simulation (100 rays) with Ray Preview turned on, as shown earlier. (Use **Illumination > Simulation Info** to change the settings.) Or, you can just re-run with the most recent settings.

1. Click  on the toolbar.
2. Check the results in the LumViewer chart. If you still have a chart window open, click the name in the Window Navigator to bring it to the front. Otherwise, select the receiver again (in the System Navigator or the design view) and select **Illumination > Illuminance Display > LumViewer**.

   The following chart was created with 20,000 rays.

   Note that the distribution is somewhat flattened out because of the extended source, and there is a dip in the middle due to the lamp base and hole in the reflector. It's still not very uniform.

Save Your Work

If you haven't done so recently, save your work before moving on to the next steps.

To do this, select **File > Save As**, enter a new name such as SmoothReflLamp.1.lts, and click Save.

Create a Faceted Reflector

One solution to the problem of uniformity is to use a faceted reflector. Provided with *LightTools* is a powerful Faceted Reflector utility that you can use to create faceted reflectors with specified properties.

LightTools provides a number of utility programs and macros that make certain tasks easier. Although they are written using the same programming tools available to you (the COM/Visual Basic interface and the earlier *LightTools* Basic macro language), no programming is required to use these utilities. They generally run in a separate window through a user interface that is different from that of *LightTools*, but still very easy to use.

Delete the Smooth Reflector

To prepare for the next step (using a faceted reflector) first delete the smooth reflector.

Tip: To make sure nothing else is selected for deletion, click a blank area to clear the selection buffer, and then select the reflector.

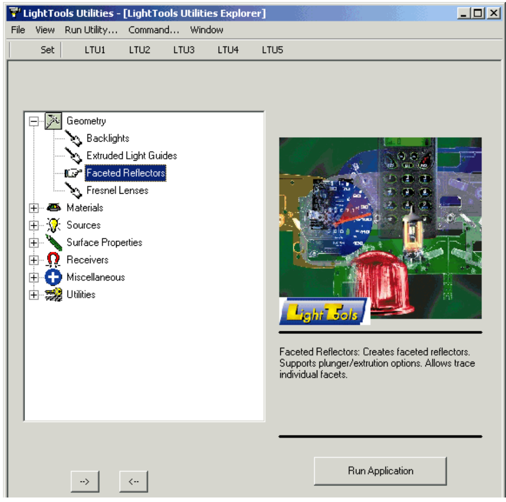
1. Select the reflector.
2. Right-click and select **Delete** on the shortcut menu.

Running Utilities

Although utilities are actually separate programs that communicate with *LightTools* when they run, you can launch the Utility Library from the **Tools** menu.

1. Select **Tools > Utility Library**.

   This launches a new window from which you can browse, select, and run utilities.



You can access the supplied utilities from this window using the **Run Utility** menu at the top or the navigation tree on the left.

2. Expand the **Geometry** category in the browser panel.

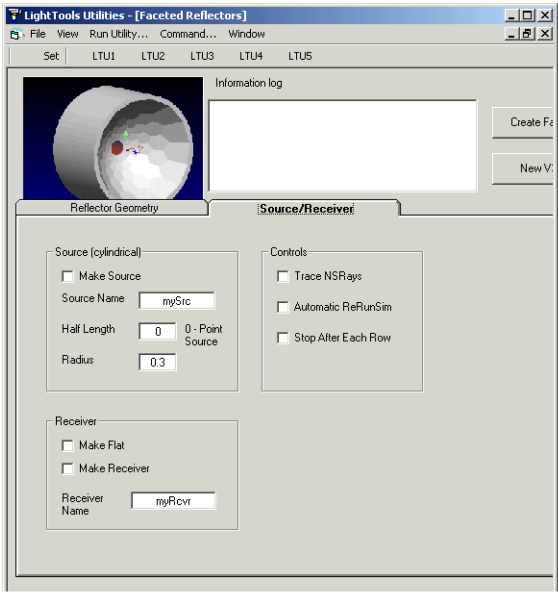
3. To launch the Faceted Reflector utility, you can either:

   – Double-click on the **Faceted Reflectors** item.

   – Click once on the **Faceted Reflectors** item to select it click on the **Run Application** button.

   This utility has a picture of a typical faceted reflector, an information area, several buttons, and two tabbed pages.

4. Click on the Source/Receiver tab and *un*check all the check boxes that are checked by default, as shown below.



This utility can create sources and receivers and do other special tasks. Since your model already has a source and receiver, you don't need these features now.

5. Click on the Reflector Geometry tab.

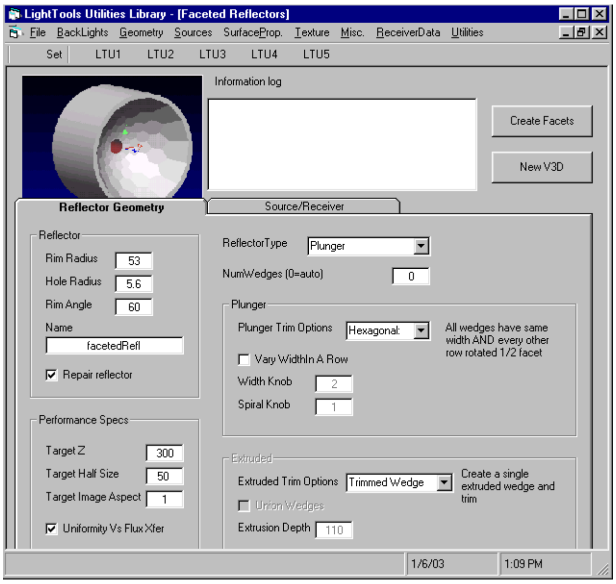
6. Enter **53** for the Rim Radius, **5.6** for the Hole Radius, and **60** degrees (default) for the Rim Angle.

   These values approximate the geometry of the original smooth parabola. The program automatically places the focus of the reflector at the point (0,0,0), which is the location of the source filament.

   This program has many options for the type of reflector, facet geometry, etc., but the default settings are adequate for this example. You need to provide performance specifications, however (in the lower left corner), based on the goals originally stated (i.e., uniformity over a 100 mm diameter disk at a distance of Z=300 mm). The utility uses the target information (plus default values) to determine the necessary facet geometry.

7. Enter **300** for the Target Z and **50** for the Target Half Size.

8. Click the Create Facets button.

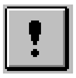


The facet creation operation can take a little time (maybe a minute or so), and you can switch over to *LightTools* and watch as the command panel flashes and the geometry is created. The utility is actually a separate program that runs *LightTools* by “remote control,” sending commands to it via the Windows COM facility. When the Interrupt dialog box is no longer displayed, the creation process is complete.

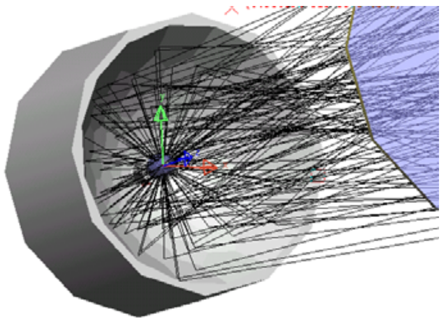
Run an Illumination Simulation

The source and receiver are still in place from before, so it's easy to run another simulation on this system. But since the geometry has changed quite a bit, you should do a small run with Ray Preview turned on to prove that the rays are doing what they should.

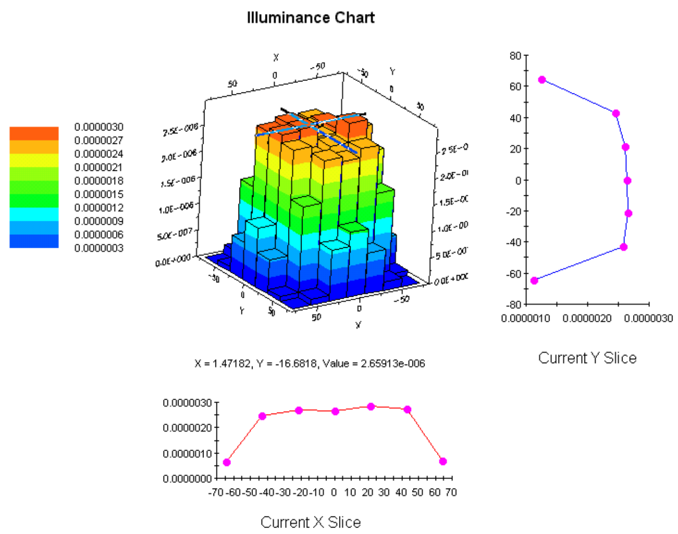
1. Select **Illumination > Simulation Info**, change Total Rays to Trace to **200**, check the Preview Rays box, and click OK.

2. Click the  button on the toolbar.

   Your model should resemble the following figure (after right-click rotation).



If it looks OK (did you delete the smooth reflector?), go ahead and select **Illumination > Simulation Info** to define more rays (10,000) and turn off the Ray Preview. Re-run the simulation, then select the receiver and open a LumViewer chart to examine the illuminance. Open an Illuminance Raster Chart too, and look at the numerical results. The LumViewer output shown below is for 20,000 rays.



The uniformity is much improved over the smooth reflector case. Try running with even more rays (50,000 or so) to get better spatial resolution.

3. Select **File > Save As** and save this faceted reflector version with a new name.

Create the Flashlight Body

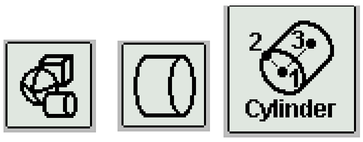
Although it doesn't affect the optical performance, adding a simple mechanical model for the flashlight body will make the model look more realistic and complete. This can be done with a few mechanical cylinders (one tapered) and a few Boolean subtract and union operations.

Since the objective is to present a quick visual impression, you don't have to be too precise about measurements, but it's still a good idea to use the Snap feature to be sure things line up properly. Some steps (e.g., Boolean subtract) are described just briefly because they are similar to the procedure for making a hole in the smooth reflector a few pages back.

Collar

You could certainly trim the reflector to be a little smaller, but you don't need to bother in this case. Just make the collar big enough to house the reflector. Make sure you are in wireframe mode in the right side view.

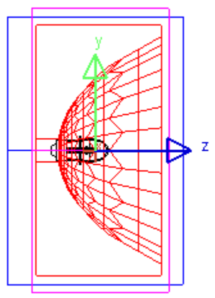
1. Select the mechanical Cylinder button (***Mcylinder***) as follows:



2. Right-click and select **Snap > Z-axis**; then, click a little to the left of the rear (left) surface of the reflector (around Z = **-30**).
3. Click to make the radius of the cylinder about **66** mm.
4. Right-click and select **Snap > Z-axis**; then, click just to the right of the reflector (around Z = **34**).

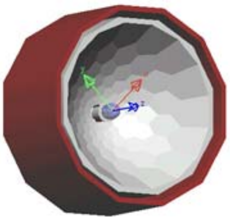
   This is the outer ring of the collar.
5. Click the cylinder tool again and repeat steps 2, 3, and 4 with the following values: left Z = **-40**, radius **62** mm, right Z = **+40**.

   The model should resemble the following example.



6. Select the larger-radius (66mm) cylinder; then, shift-click the inner cylinder. The order in which you select the cylinders is important.
7. Select the Boolean **Subtract** button (or type the command ***Sub*** for ***Subtract*** on the command line and press Enter).

   The collar now surrounds the reflector.



Create the Battery Compartment

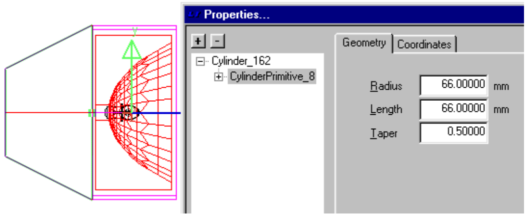
Next, you need to make a tapered cylinder to connect the collar to the battery compartment of the flashlight.

1. Select the ***Mcylinder*** button again (or, type ***Mcyl*** in the command line and press Enter).
2. Right-click and select **Snap > Object**; then, place the mouse near the axis at the left side of the collar assembly. **z = -30** is good.

   Snap-to-object finds the nearest object vertex, helping you to line up the next cylinder with the collar assembly.
3. Click to define the axis of the new cylinder. Make its radius (R) **66** mm, using **Snap-to-Z** (z will be around **-97**) to make its length (L) **66** mm.

4. Select the new cylinder, right-click, and select **Properties**.
5. In the Object Navigator, select the CylinderPrimitive, and enter **0.5** for the Taper parameter.

   Taper is the ratio of the end diameters. (It would be 2.0 if you had started the cylinder from the other end.) It should look like this:



   Before you create the final cylinder, you may want to zoom out.

6. Type ***Mcyl***, then start at the left end of the tapered cylinder (Snap-object), radius **33**, length **300** (snap-Z).

   If you are off on any of the measurements, use the Properties dialog box to correct the values. Visually speaking, you are finished, but, because the flashlight body can be considered one unit, you can use a Boolean union to join all the body parts.

Join the Parts Using a Boolean Operation

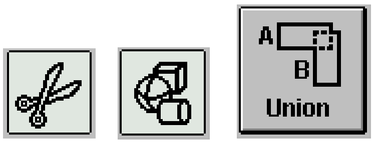
1. Change to translucent mode (**View > Render Mode > Translucent**).

   This makes it easier to select the body parts. Note that, using default surface rendering settings, mechanical parts are not translucent, even in translucent mode.

2. Click on the collar assembly, then Shift-click on the tapered cylinder, then Shift-click the long cylinder.

   Note that, in shaded views, selected objects highlight with both wire frame color and surface shading. (The default color is purple.) Check the System Navigator to be sure that all mechanical cylinders are selected (and that the reflector assembly and lamp are not selected).

3. Select the ***Union*** button:



4. Try this simple test to verify that the objects were unioned:

   a. Select the flashlight body, type ***Move*** in the command line, and click somewhere in the design view to specify a new location.

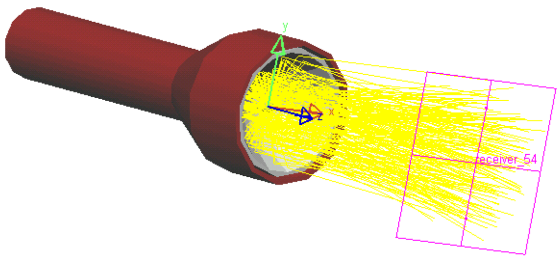
      The flashlight body should move as a unit.

   b. Select **Edit > Undo** to return the flashlight body to its original position.

5. Save your work!

Conclusions

The final model should resemble the picture below. For this example, a small simulation was run with the Ray Preview turned on, and the color of the rays was changed to yellow in the Properties dialog box.



In this chapter, you learned even more about how systems are built and analyzed in *LightTools*. The final chapter introduces the concepts of backlight design using another powerful utility that is supplied with *LightTools*.