

Segmented Reflector Design

Automotive reflector design process with ASAP and ReflectorCAD

This technical publication describes the automotive segmented reflector design process from first-order design to standards testing using ReflectorCAD® and the Advanced Systems Analysis Program (ASAP®) from Breault Research Organization (BRO). A brief discussion on some of the important issues in a general design philosophy is included, but the emphasis is on the detailed, modular design steps for segmented reflectors using ReflectorCAD, ASAP, and ELTM (Exterior Lighting Test Module) in ASAP.

This methodology serves as a design road map for segmented headlamp design, but many of its components can also be used for performing general parametric analyses, sensitivity analyses, and tolerance analyses (tolerancing can also be done in ASAP using either the Builder or Editor user interface). The procedure can also be adapted for other headlamp designs, such as projector headlamp systems, and other types of automotive exterior lighting systems.

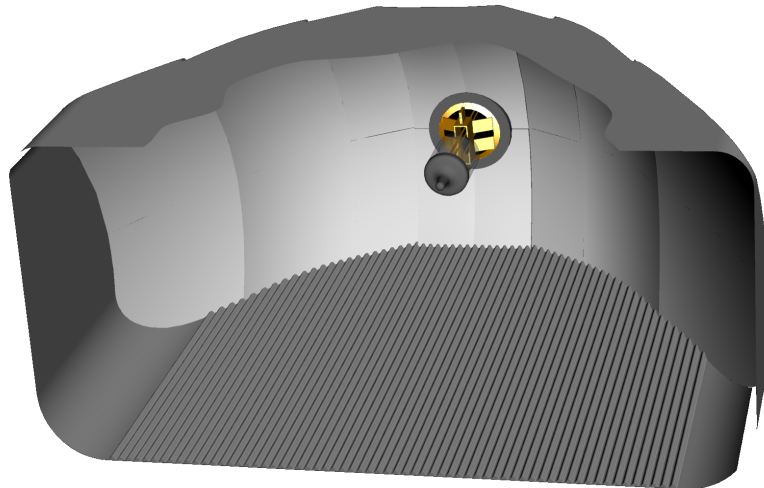


Figure 1 3D view of a segmented reflector

Trend toward segmented reflectors

The traditional, rectangular-shaped, sealed beam headlamps, with parabolic reflectors and cylindrical cover lenses, have almost become a design of the past. They are being replaced with either segmented reflectors containing clear windscreens or projection headlamp systems. A segmented reflector is illustrated in Figure 2.

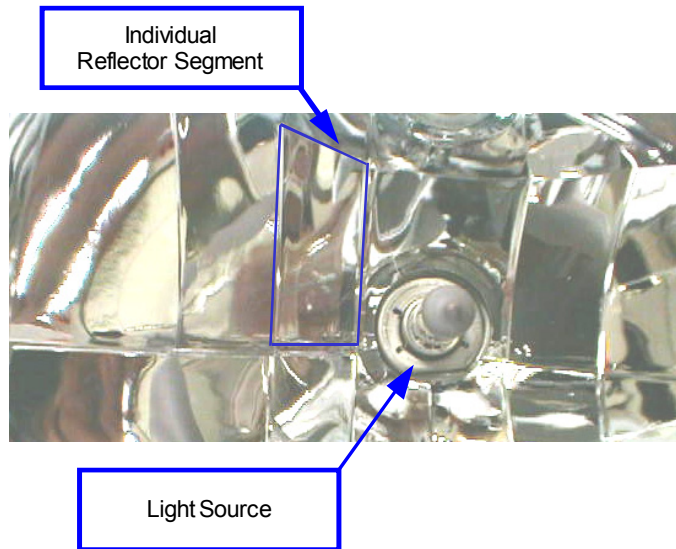


Figure 2 Segmented reflector

Segmented reflectors transfer the optical power or beam-shaping refractive qualities of the cylindrical cover lenses to individually curved and oriented segments comprising the reflector. Each segment contributes to the intensity of the headlamp.

Transferring the lighting control exclusively to the reflector allows headlamp designers additional degrees of freedom to sculpt a clear cover lens or windscreen into the body of the car— something that cannot be done aesthetically or easily with cylindrical lenses. Projection headlamp systems actually use imaging optical lenses to image or project the desired illumination pattern onto the roadway.

In addition to meeting design criteria, segmented headlamp and projection system designs must comply with the strict lighting standards established for all headlamps. The three primary standards used worldwide are:

- Ground Vehicle Lighting Standards, developed by the Society for Automotive Engineering (SAE)
- Federal Motor Vehicle Safety Standards (FMVSS)
- Economic Commission for Europe (ECE) Standards



Prototyping versus computer simulation

Historically, headlamps were engineered as prototypes to the above standards, as well as the manufacturer's specifications. Now, it is no longer cost- or time-effective to create prototype after prototype to design a headlamp. End-to-end computer simulation of headlamps—from first-order to engineering design using Monte-Carlo geometrical ray tracing—is now an accepted, accurate, and precise method for designing and analyzing headlamp performance before, during, and after product development. While many simulation programs cannot support end-to-end computer design and analysis, BRO's ReflectorCAD and ASAP software do.

Seamless design and analysis environment

ReflectorCAD and ASAP include many features that facilitate a complete, smooth, and seamless design and analysis environment. Segmented reflectors can be entirely designed and analyzed in ReflectorCAD and ASAP. Designs can be created within ReflectorCAD and transferred to ASAP for engineering design, and tested against SAE, FMVSS, and ECE standards using their graphical interfaces, without typing any script commands.

The macro (script) language in ASAP is still powerful enough to solve other problems also encountered in headlamp design that other software products cannot solve due to their limited interfaces. The fact that ASAP allows you to go beyond its user interface and solve difficult problems using the macro language is an important feature that distinguishes ASAP from other programs.

Choose the right software for your project

The general system performance requirements dictate the optical configuration from concept to first-order to final design. The primary optical performance requirement for forward lighting is that the headlamp must conform to one of the established SAE, FMVSS, or ECE luminous intensity standards. In addition, the headlamp must conform to mechanical (styling), electrical, manufacturing, and cost requirements.

The relationships between the optical performance requirements and important physical properties are determined based upon knowledge of optics, previous experience, patents, and other external inputs such as benchmarking.

The physical properties derived from the optical performance requirements are used as input into the software design tools. The important point to emphasize is that these inputs dictate the type of software tool needed for the problem and not the reverse. The software tool must have the capabilities to accurately simulate the physical properties that affect performance.

Table 1 applies to the design of a basic headlamp, the first-order optical design parameters they drive, and the software capabilities required to simulate the related physical properties. This summary is not meant to be a comprehensive listing of the relationship between all system requirements and physical properties (software capabilities). It is a guide to help you better understand the relationship between some requirements and the software capabilities and features needed to design a headlamp. Furthermore, the optical properties from one category can affect the performance in other categories.

Table 1 Relationship of requirements, first-order properties, and optical properties

Optical Performance Requirements	First-order Optical Properties	Physical property/Software capability (feature)
Luminous intensity: SAE/FMVSS/ECE	<ul style="list-style-type: none"> •Source radiometry (emission/etendue) •Reflector focal length •Lens focal length 	<ul style="list-style-type: none"> •Source luminous power, near- and far-field emission patterns •Reflector shape/segmentation •Material reflectivity (including surface roughness/scatter) •Lens dispersions and transmissivity (including surface roughness/scatter) •Mechanical housing geometry and material reflectivity (including surface roughness/scatter) •Test point/zone evaluation, pass/fail criteria, isocandela plots •Fresnel effects (reflection/transmission losses), absorption losses
Color: SAE/FMVSS/ECE	<ul style="list-style-type: none"> •Source emission (for example, halogen, filament quartz xenon [HID]) •Source radiometry 	<ul style="list-style-type: none"> •Emission model: surface or volume •Polychromatic sources •Source power spectral density/distribution •Source geometry

Measurements also are a crucial input for the design process—they quantify the physical properties. The accuracy of the simulation is directly influenced by the quality of the measured data. Original equipment manufacturer (OEM) suppliers are sometimes unwilling or unable to supply the data you need for the simulation, so be prepared to measure the physical property yourself.

Measurements are still the best course of action, even though many software packages include material databases for simulating some physical properties. At minimum, check the information from supplied databases used in your simulations against measured data for accuracy. If you do not, you could end up with a bad design. For example, if you use surface reflection coefficients that are the incorrect magnitude, the radiant output from your simulation will be incorrect. If you are using a source model with the wrong emission properties, the luminous intensity of your headlamp will be incorrect.

Optical design and analysis

The optical design process, even using existing user interface programs, means different things to different people. Lens designers designing classical lens systems (such as cameras, telescopes, microscopes), primarily use commercially available “optical design” software programs, which are really lens design programs. They, like illumination designers, start the design process by defining the first-order optical properties of the optical system

from the requirements. Their first-order optical properties are usually different from the illumination designer's optical properties.

Lens designers are interested in first-order optical properties such as image location, magnification, effective focal length, and radiometric/photometric properties. They then use the mathematical tools of first-order optics, such as the y-bar diagram, Gaussian reduction, paraxial imaging, and the power transfer equation, to translate the first-order properties into a preliminary system layout, which serves as a starting point for the actual lens design. Some also use patent applications and experience to generate design starting points. In any case, the first-order design layout is input into a lens design code.

Differences between a lens design code and an optical analysis program

Lens design codes are really geometrical ray trace simulation programs. They use some form of an automated mathematical optimization algorithm to determine an optimum lens design for a given set of conditions. The lens design code automatically changes element radii of curvature, thickness, spacing and refractive indices, while performing and evaluating geometrical ray traces, to force the optical system to conform to a certain merit function. You can think of the automated lens design process as a kind of feedback loop. Lens design codes are very efficient at finding solutions to this type of problem.

Unfortunately, the specific simulation and optimization features of lens design codes usually are impractical to use in illumination systems. For example, it is difficult or impossible for lens design codes to model some of the unusual illumination system geometries, extended sources, and non-sequential ray trace behavior. Furthermore, an extremely large number of rays must be traced to simulate, in a Monte Carlo sense, the extended sources of illumination systems. These ray traces typically take a significant amount of time. Large Monte Carlo ray traces used with lens design optimization routines involve an algorithm trying to solve a non-linear problem, while requiring a minimum of hundreds of iterations to reach a local minimum.

Illumination engineers who design automotive headlamps primarily use computer-aided design software (CAD) and optical analysis packages. Optical analysis codes are used to analyze those phenomena that cannot be easily simulated in lens design codes such as the extended sources of illumination systems. Some optical analysis programs contain a CAD interface or CAD translators to facilitate the illumination design procedure. Their programs are CAD interfaces wrapped around ray tracing engines. The interface allows you to input geometry and drag and drop it with the click of a mouse. These programs may allow you to draw the segments of the reflector and then trace rays to determine the radiometric output, but their processes are not automated. You must still supply the design algorithm for achieving the desired performance. That is, you must change the optical prescription of the segments yourself to achieve a different performance, and then trace rays to see the result. Moreover, there is no graphical interaction between the input geometry and the radiometric output.

Optical engineering, and particularly optical design, are inherently not graphical disciplines. You cannot design an optical system by simply “drawing” optical component shapes. You must take into account the spatial interaction between light, and optical materials and shapes that change the behavior of the light after it interacts with the optical components. This is an important point because the required illumination pattern is already known and what you really want is to have the software create an optical prescription that produces that pattern. ReflectorCAD was developed with these issues in mind.

ReflectorCAD for segmented reflector design

ReflectorCAD is a true user interface, providing long needed illumination design for segmented reflectors. An application-specific program (ASP) with a user interface, ReflectorCAD was designed from the ground up specifically for designing segmented reflectors. The goal of an ASP is to increase user productivity by substantially

reducing learning curves and development times. With ASPs, you can make repetitive and often complex work-place tasks less expensive as well as easier for users to learn and use. This is accomplished by automating specific steps and presenting them with a logical, unambiguous methodology.

The intuitive interface in ReflectorCAD allows you to graphically lay out reflector segments, to quickly compute the approximate intensity from individual segments or all of the segments and a source, and most importantly, to graphically adjust the radiometric output (intensity) from each segment. ReflectorCAD automatically computes the segment geometry and aim to achieve that output. ReflectorCAD also automatically fills in the spaces between segments with intersegment fillers, and allows you to visualize in 3D the reflector, source, segment shadowing, and intersegment fillers. ReflectorCAD accomplishes this with a design algorithm that is a radical departure from commercially available user interface programs. ReflectorCAD is a truly integrated user interface!

In ReflectorCAD, the design process works from the performance end. Instead of altering the segments, you graphically alter the aim and spread of the radiometric output (intensity) from a segment. ReflectorCAD automatically adjusts the optical prescription and aim of that segment to achieve that radiometric performance. The reflector segmentation geometry, including intersegment fillers, can be directly output to ASAP for further detailed engineering design and analysis or to an IGES file for use with a CAD package.

You first define a base reflector surface in ReflectorCAD, which approximates the desired shape of the finished reflector. The surface may be a general conic or a sampled (discrete) surface from ASAP. You next select a source type. In ReflectorCAD, sources are collections of rays, and do not include the actual source geometry. Several source models of common automotive bulbs are available.

Radiant Imaging's Radiant Sources™ can also be used in ReflectorCAD. In fact, virtually any ASAP source can be converted into a ReflectorCAD source, including HID or discharge sources. Once the base surface and source are defined, you then create segments, which forms the reflector surface. This is done graphically in the ReflectorCAD Segment/Reflector View window. The entire graphical interface is one window with two views. The

other view is the Output View, which is an isocandela plot (intensity) as a function of angle. See Figure 3.

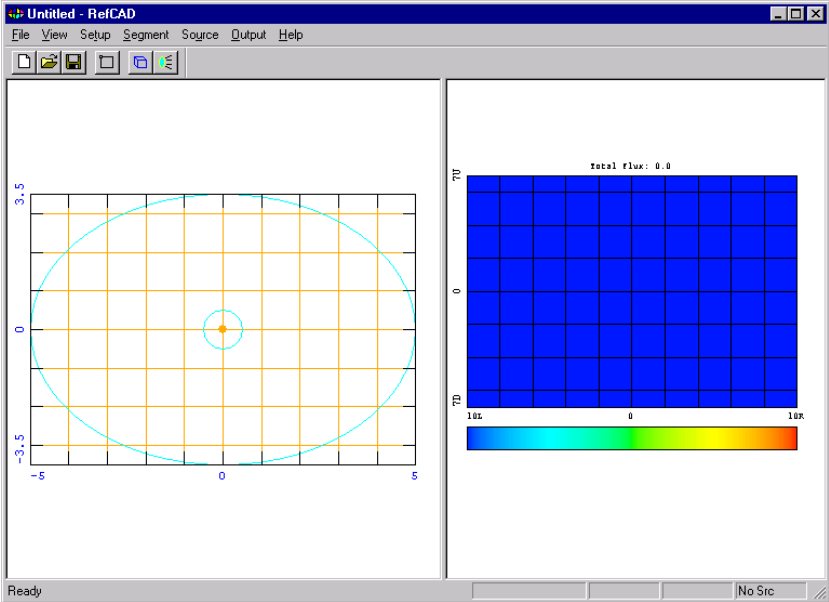


Figure 3 ReflectorCAD main window

ReflectorCAD uses the source to set segment aiming and to compute the system output. ReflectorCAD quickly calculates the output due to a particular segment. It is so fast, in fact, that an individual segment's output is computed almost instantaneously. The fast ray trace engine is key for allowing you to rapidly adjust the output from individual segments to assess their contribution to the required intensity pattern. By default, each segment is adjusted to produce an intensity pattern with a specific direction and spread in angle space.

After a segment is created, there are several ways to alter a segment's aiming and spread and thus its intensity pattern. ReflectorCAD allows you to drag any set of aim points, corresponding to a specific segment's intensity pattern, to any desired angular location in the Output View, by simply using the mouse. This operation essentially tips and tilts the segment that corresponds to the intensity pattern you have graphically indicated with the mouse in the Output View. Any one of the four aim points may also be individually dragged to alter the spread of the segment's intensity pattern.

An aim balance point allows you to distribute the peak of the intensity distribution within any of the four aim points of the aim region. If the aim balance point is located in the center of the aim region, ReflectorCAD attempts to spread the light evenly throughout the aim region. Both of these operations essentially change the optical prescription of the segment to achieve the radiometric performance that you have graphically indicated with the mouse in the intensity plot of the Output View. See Figure 4 and Figure 5.

Segmented Reflector Design

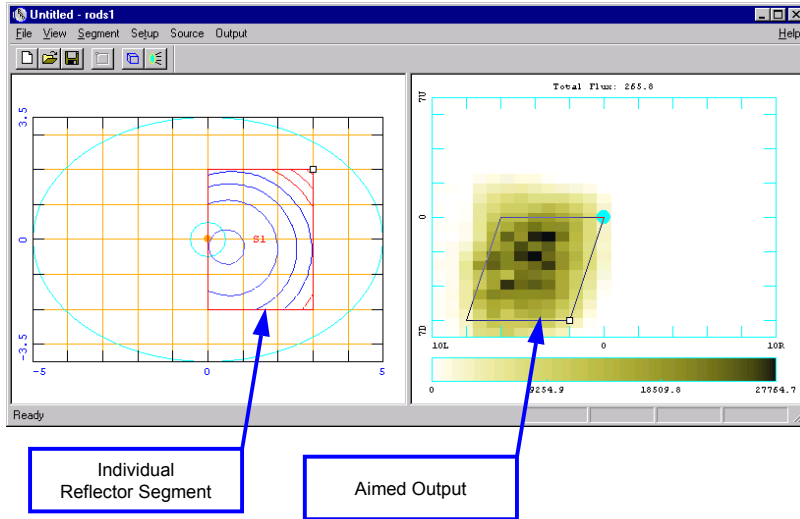


Figure 4 Segment aiming

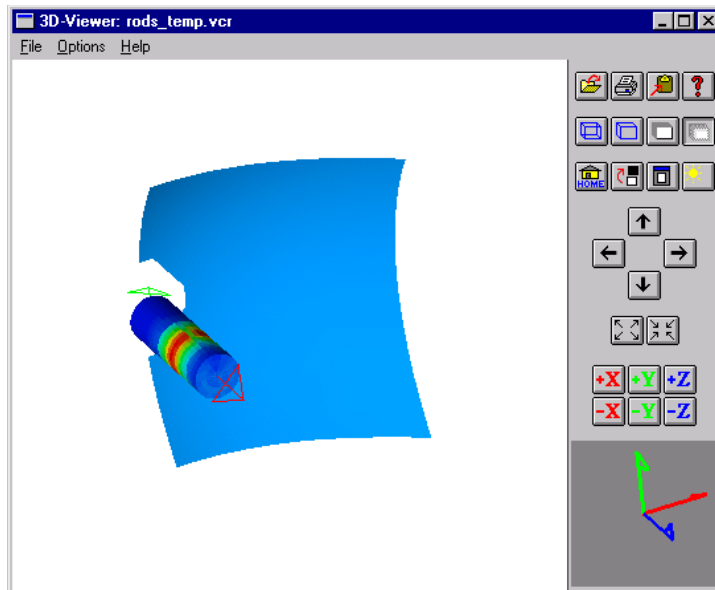




Figure 5 Resulting segment with 3D visualization

ReflectorCAD supports 3D visualization, which allows you to view the segmented geometry and source at any time during the design. You can view segments and the source after they are individually created, or after you have created several segments. The 3D visualization feature is especially useful for examining segmentation and intersegment fillers, which connect the edges between adjacent segments. Two adjacent segments are often discontinuous at their common edge, because a different radiometric performance or segment prescription is re-

Segmented Reflector Design

quired from the segments. See Figure 6, Figure 7, and Figure 8.

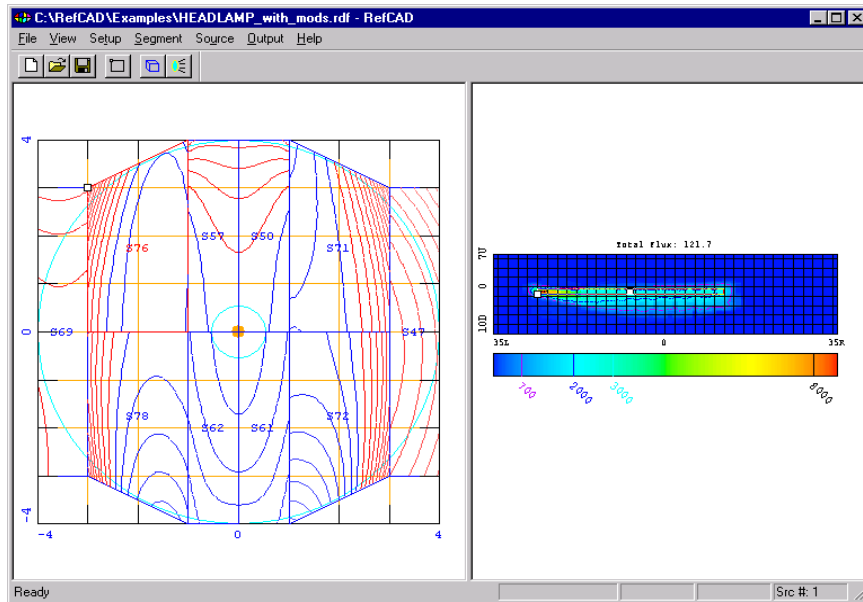


Figure 6 Individual and total segment contributions, part 1 of 3

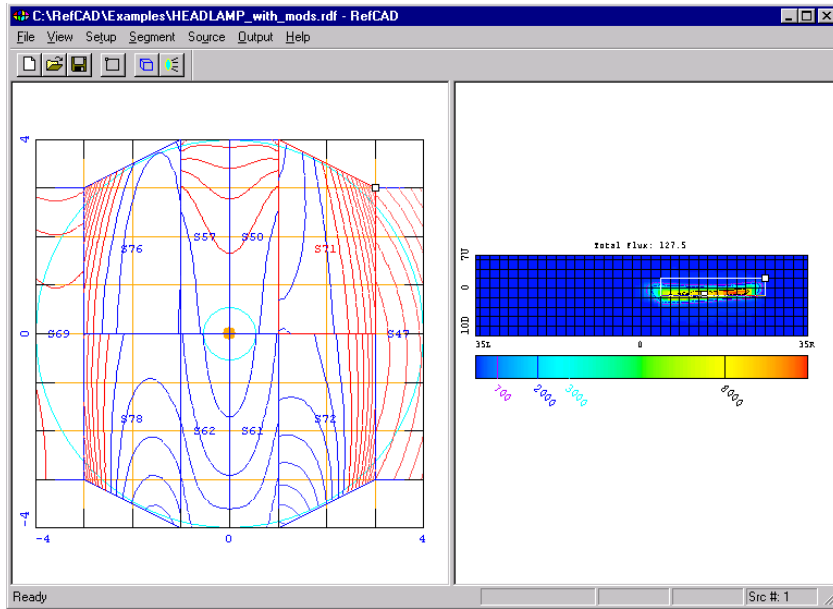


Figure 7 Individual and total segment contributions, part 2 of 3

Segmented Reflector Design

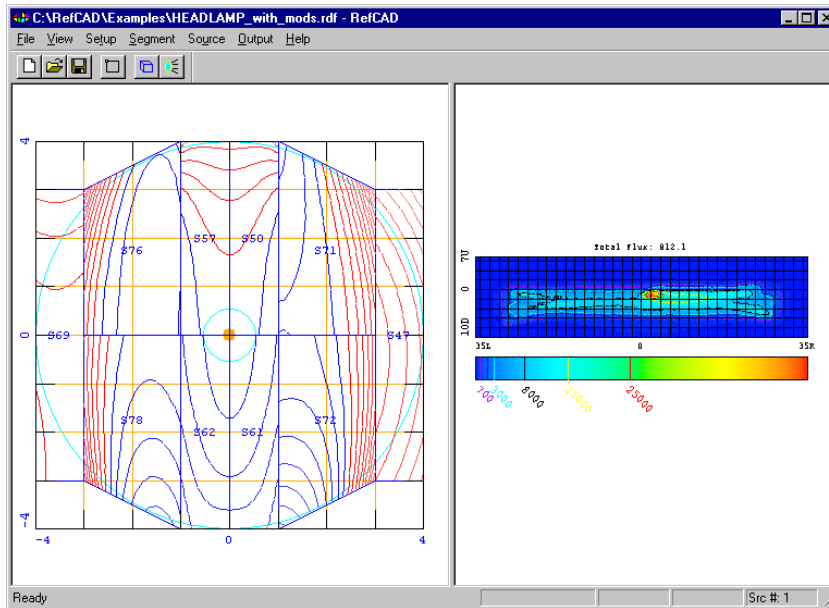


Figure 8 Individual and total segment contributions, part 3 of 3

A planar surface that fills this gap is the “intersegment filler”. If the intersegment filler is illuminated by the source, light reflected from this plane can cause undesired illumination in the output. It is usually necessary to hide these intersegment fillers by forcing the near part of the segment furthest from the source to lie in the shadow of the adjacent segment closest to the source.

ReflectorCAD not only automatically creates intersegment fillers, it also provides a numerical and graphical method for checking whether the intersegment filler is hidden in a segment’s shadow. Figure 9 illustrates graphically that a segment is properly hidden; that is, the intersegment filler lies behind and is shadowed by the segment closest to the source.

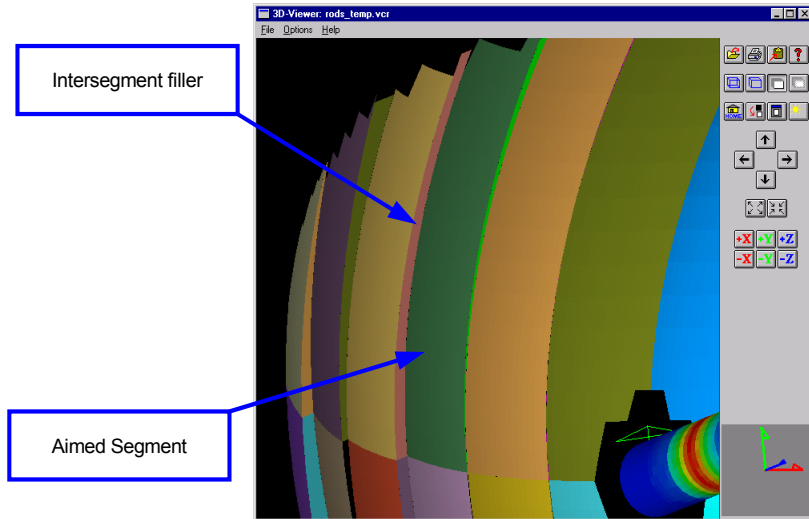


Figure 9 Complete reflector and associated intersegment fillers in 3D Viewer

Exporting your design for further analysis

After the initial design is complete, you can translate the reflector segmentation geometry, including intersegment fillers, directly to ASAP and ELTM for further detailed analysis, to an IGES file for use with a CAD package, or to other optical analysis programs that support the IGES format. Further detailed, high-accuracy analysis is required because ReflectorCAD is a layout tool that does not include analysis capabilities, like surface roughness and variable reflectivity. It also does not model other mechanical structures that might be associated with the system design such as reflector top and bottom “shelves”, bulb shields, and additional mounting structures.

See Figure 10.

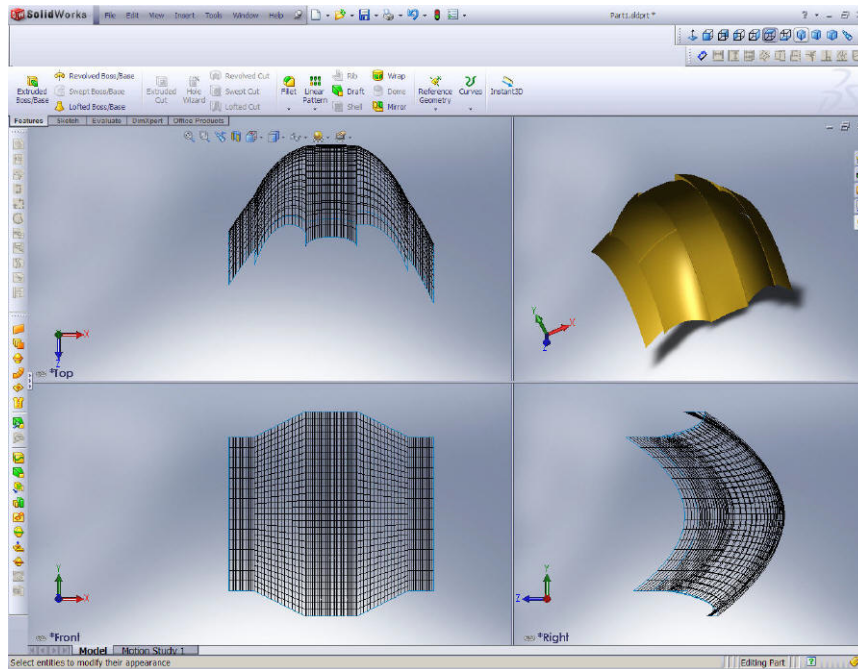


Figure 10 Exported IGES file, viewed in SolidWorks®

Engineering design

Once your design is created in ReflectorCAD, you are ready to export it to ASAP for engineering design, analysis and testing. Why do you now need ASAP for engineering design? Good final designs start from a good basis. Accurate analyses and testing require integrating the headlamp's physical (optical) properties into the design, especially those that contribute directly to the required performance. These optical properties, such as reflection and transmission behavior, including scatter, first must be quantified by measurement and converted into modeling parameters of the simulation program. ReflectorCAD does not model these properties, which can influence the headlamp performance—ASAP does.

Analyzing your design with ASAP

You have two choices for exporting your segmented reflector from ReflectorCAD into ASAP:

- 1 directly into the ASAP command language, or
- 2 into an IGES file.

When you choose to export the file to an IGES format, you can then work with the user interface in the ASAP to IGES translator to add optical properties directly to the headlamp. The IGES translator is integrated directly

into ASAP, so you only need to double-click an IGES file to automatically import it into the translator. See Figure 11. The IGES translator allows highlighting of objects of interest, either in the object list or a graphic on the plot screen. You can then modify the object's interface through a dialog box. Use this dialog box to enter your own materials, reflection and transmission coefficients.

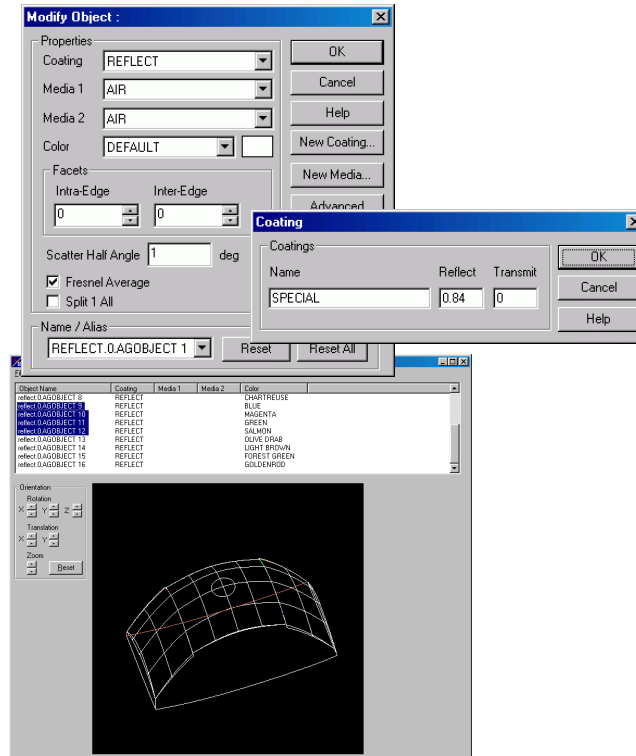


Figure 11 ASAP to IGES translator

ELTM for compliance testing

Once you have added optical properties to your geometry and created an ASAP file, you can enter this file and any files describing other opto-mechanical geometries into ELTM, which was created specifically for the automotive industry at the request of our automotive clients. Three ELTM input dialog boxes lead you through ASAP simulated SAE, FMVSS, and ECE tests, and then displays the test results graphically in an isocandela plot and a numerical table. The numerical results are also displayed as pass/fail. You do not have to enter a single ASAP command to do this.

Comparing Radiant Imaging and BRO Light Source Library sources

Part of the ELTM process, in addition to entering the reflector/mechanical housing geometry, is to enter a source. The ray data may actually be from the same data used in ReflectorCAD. The sources can also be from Radiant

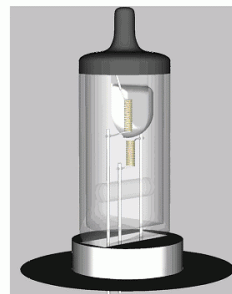
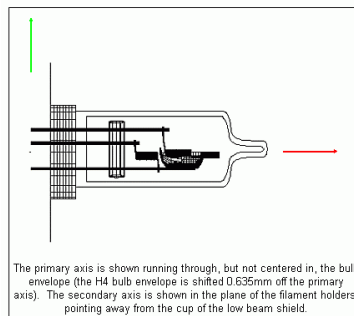
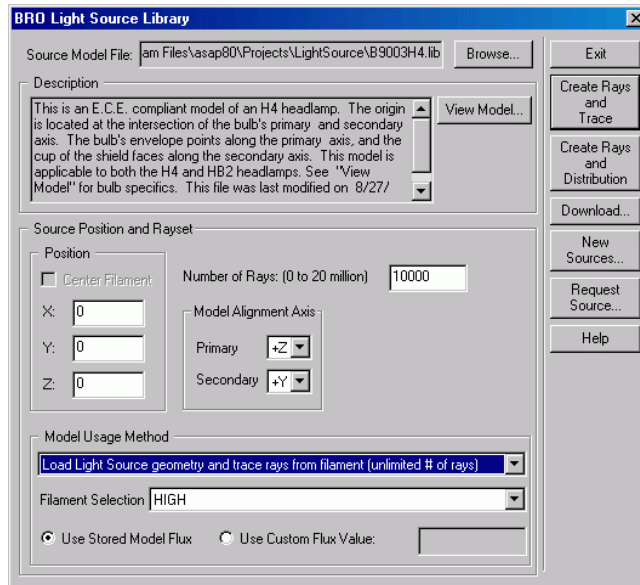
Imaging measured data (Radiant Imaging's ProSource® must be installed), or from the BRO Light Source Library. We refer to Radiant Imaging's models as "sources" instead of "bulbs" for the following reason. The ASAP Light Source Library models contain the actual optical and mechanical geometry from the bulb, which is necessary for accurate and thorough analysis. A Radiant Imaging source does not contain any geometrical information.

Radiant Imaging sources can be used for analyses. However, light reflecting off the reflector and mechanical housings, which normally passes back through the source model, passes through empty space with a Radiant Imaging source. In fact, the light can be reflected, refracted, or scattered by the source model geometry. The ASAP source models with bulb geometries properly account for these optical interactions, which many times are the cause of, or lead to, stray light in automotive optical systems. This is an important consideration in automotive lighting. A method exists for using Radiant Sources with the BRO source geometry to compensate for this.

Light Source Library in ASAP

The BRO Light Source Library has all common U.S. and European automotive source models, with detailed source geometry. A complete list of available sources is at <http://www.breault.com/software/asap-lightsources-lib.php>.

A sample of one of the many ASAP automotive source models is shown in Figure 12. Note the geometric detail required for accurate analysis. You also have your choice of how the source geometry and rays are used in your analysis. You can load the source model geometry and trace an unlimited number of rays. Alternatively, you can load the source model geometry and stored ray data, and trace rays from the outer shell of the source model. While this method is faster than the previous method, you are limited in the total number of rays you can trace. Lastly, you can use the stored ray data without the source model geometry. This is the fastest method, but it does not include the source model geometry, and you are again limited in the total number of rays you can trace.



Notes:

- There is intentional tilting of the low beam shield, it is rotated 7.5 degrees about the primary axis.
- The center of the low beam filament is 31.25mm above zero, on the primary axis.
- In addition to an offset in the primary axis, the source's filament has another offset(s). With the primary axis set to +Z, and the secondary axis set to +Y, the true center point of the high beam filament is at (0.00,-0.80,24.55), and the low beam filament is at (0.00,-0.07,31.25) in (X, Y, Z).
- This bulb has an output of 1468 Lumens (high beam) and 890 Lumens (low beam), while running at 60 and 55 Watts,

Figure 12 Sample source model from the ASAP Light Source Library

Once you have chosen your source model, you then add this to ELTM, which guides you through the ray tracing and automotive testing. After adding your geometry and source models, you then choose from a menu of available tests.

Currently SAE, FMVSS, and ECE tests are available with ELTM. Figure 13 illustrates this process.

After selecting your test, you then proceed to the Output Settings dialog box, where you set up the ray trace and other analysis options. You then click the trace button and wait for the results. An optional isocandela plot is displayed, and a table lists the test points (see Figure 14), and indicates whether or not the system passed or failed the tests. Highlighting a test point in the table also highlights the test point graphically. You can also output the numerical intensity data in various ASCII formats, which you can then import into visualization programs.

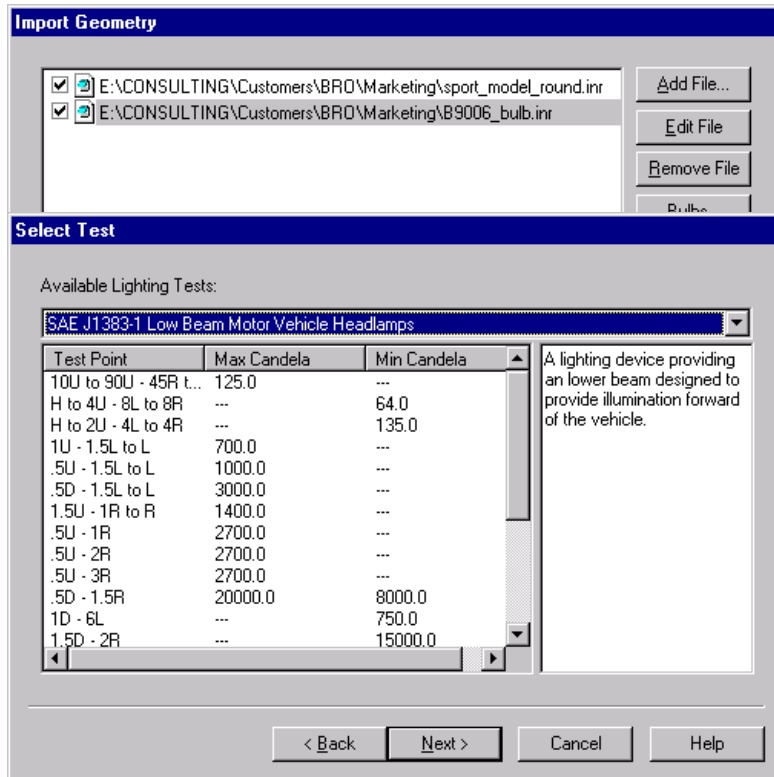


Figure 13 Dialog boxes in ELTM

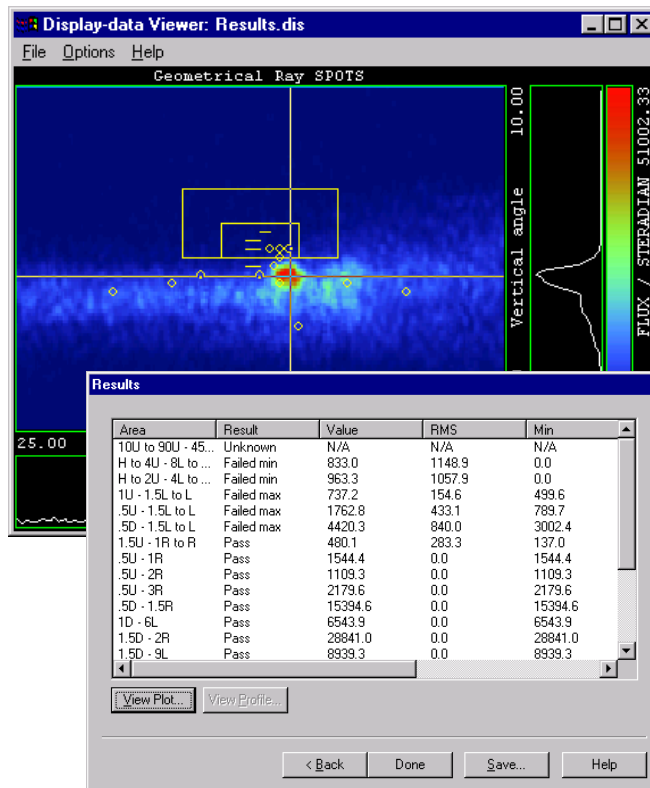


Figure 14 Isocandela plot and test point evaluates for an SAE test, using ELTM

Summary

BRO's ReflectorCAD and ASAP software tools include many capabilities and features that facilitate a complete, smooth, seamless segmented reflector design and analysis environment. In fact segmented reflectors can be entirely designed and analyzed in ReflectorCAD and ASAP. First-order designs can be created within ReflectorCAD and transferred to ASAP for engineering design and tested against SAE, FMVSS, and ECE standards using their user interfaces. The entire design and analysis can be done without entering a single ASAP command.

At this time, no other commercially available software product has a wizard that walks you through an automotive analysis as easily as this. The macro (script) language in ASAP is powerful enough to solve problems encountered in headlamp design that other software products cannot because of their limited interfaces. The ability to go beyond the ASAP graphical interface to solve complicated problems sets ASAP apart from other programs.