Taking ASAP for a Test Drive

This technical publication describes fundamental concepts of the workflow in the Advanced Systems Analysis Program (ASAP®) from Breault Research Organization (BRO). These concepts are given in the form of an example project or "test drive" to use while you with the program. Steps and graphics apply to Windows 7; if you are using Windows XP or Vista, notation is given if the steps vary significantly. Sample files associated with this document are in the file, bro4318_ASAPTestDrive_Project_Files.zip.

Introduction

ASAP is an industry-standard, optical engineering design and analysis program—recognized as the most sophisticated tool available for virtual prototyping of optical systems and devices. Since ASAP was created 30 years ago, its users have come to rely on the program to design and analyze optical systems, ranging from automotive lighting components to zoom lenses, while spanning the electromagnetic spectrum from the X-ray to the microwave. Users rely on the power, accuracy, and speed of ASAP to solve their most difficult optical engineering problems.

ASAP is well equipped to handle both coherent and incoherent sources of electromagnetic radiation, as well as imaging and non-imaging optical elements and components. Since all optical systems contain both imaging and non-imaging systems, any optical engineering software worthy of consideration must be able to simulate both system types to ensure complete end-to-end, seamless simulation. A software applications program, no matter how apparently easy to use, is by definition impossible to use *effectively* if it does not include the tools that allow you to solve your problem. ASAP gives you the tools to solve your difficult problems.

Unfortunately and unfairly, ASAP is sometimes described, mostly by our competitors, as "difficult to use". They say this is why BRO does not give potential customers trial copies to test drive. They do this as their only defense against the power of ASAP. In fact, it is your problems that are difficult, not ASAP, and a powerful program like ASAP is required to solve your problems! BRO wants to let potential customers like you test drive ASAP so you can determine for yourself just how usable it is. The ASAP Test Drive was created to demonstrate that, in addition to being the most powerful optical engineering software program, ASAP is easy to use.

ASAP is like a fine sports car: it has a streamlined appearance on the surface, but plenty of power "under the hood". So "buckle up" and take ASAP for a spin. Enjoy the ride!

.

ASAP WINDOW

After ASAP is installed, double-click ASAP 2012 on your desktop to launch the program. See Figure 1.
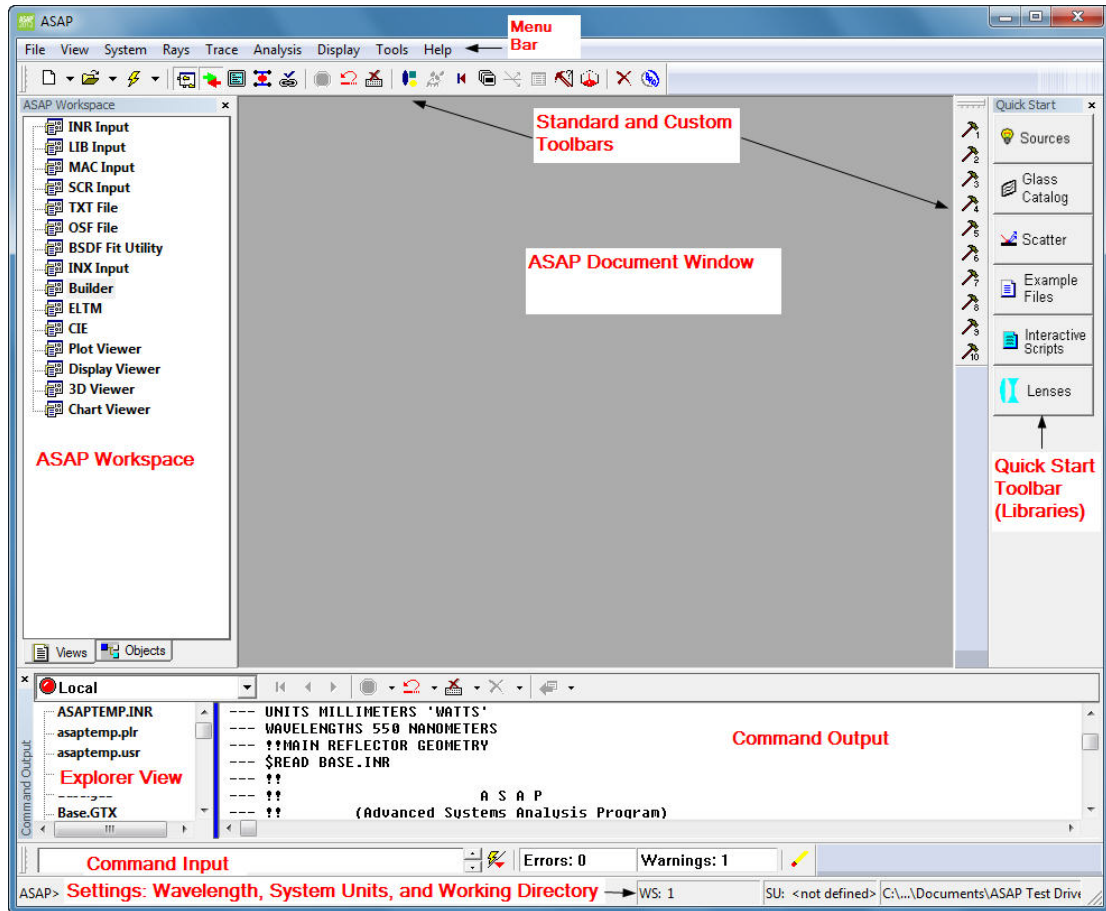

Figure 1 An illustrated main window in ASAP

- ASAP Workspace: open and view lists of document windows, files, and **OBJECT**s.
- ASAP document window: displays the appropriate file associated with file name displayed in ASAP Workspace.
- Command Input window: for entering scripting language and viewing active local or remote machines.
- Command Output window: displays the text and numerical results of ASAP calculations.
- Quick Start toolbar: for easy access of sources, glasses, scatter models, example scripts, interactive scripts, and lenses.
- Status bar: (lower main window) displays current wavelength, system units, and working directory.

TIP *Please see the ASAP Getting Started Guide for information about installation and an overview of the user interface. This publication may be viewed or downloaded from the BRO Knowledge Base, http://www.breault.com/k-base.php?kbaseID=126&catIDhttp://www.breault.com/k-base.php?kbaseID=126&catID.*

SETTING THE WORKING DIRECTORY

The working directory is an important concept in ASAP for organizing your workflow. When doing your work in ASAP, a best practice is to create a working directory and save all your ASAP files in it. You can create a working directory in your Windows Explorer, or in ASAP by clicking File> Set Working Directory on the main toolbar. See Figure 2.
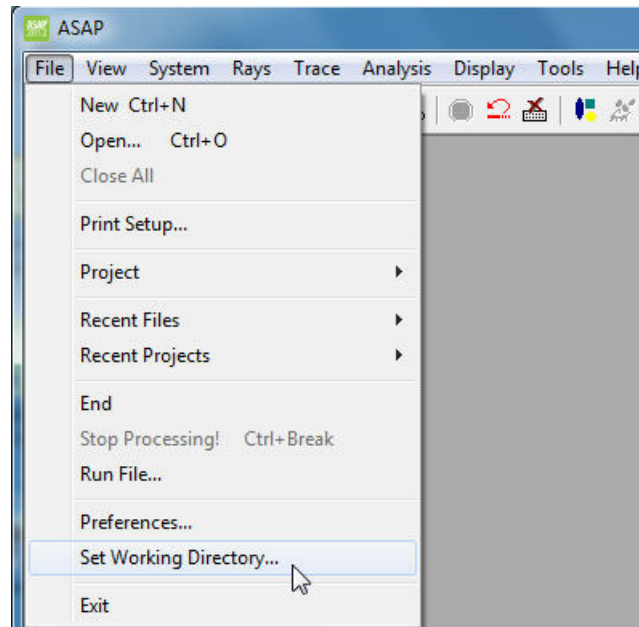

Figure 2 Selecting Set Working Directory on the File menu

For your current test drive, please create a directory on your computer and name it "ASAP Test Drive". See Figure 3.
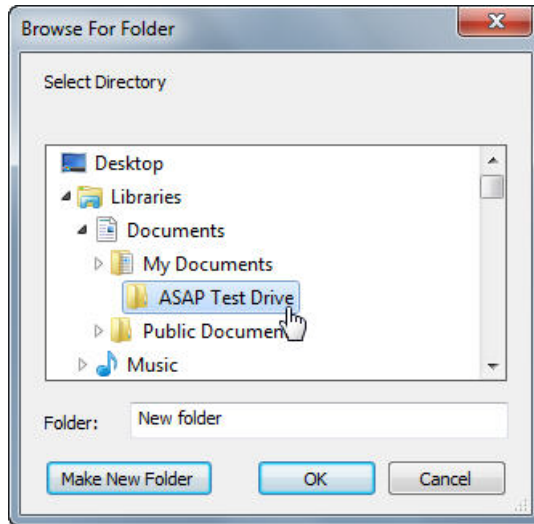

Figure 3 Setting the Working Directory

The path for the ASAP Test Drive directory is now referenced as the working directory in the Command Output window, and in the lower right corner of the status bar. See Figure 4.

Figure 4 Working directory as displayed in Command Output path and on the status bar

• Please unzip the project files, bro4318_ASAPTestDrive_Project_Files.zip, into the ASAP Test Drive directory.

FOUR-STEP PROCESS IN ASAP

Workflow in ASAP is based on a simple, four-step process that is designed into the UI. We will apply this process throughout this test drive so that you become familiar with it:

1 Define and verify the system geometry (System)

2 Define and verify the system sources (Rays)

3 Trace

4 Analyze

You can use ASAP in a variety of ways to help you design and analyze your systems. Communicate with ASAP via its spreadsheet interface, its scripting and macro language, or by importing geometry from a computer-aided design (CAD) program. During your test drive, we touch on each of these communication methods.

The standard menu on the UI contains the ASAP four-step workflow process, as well as other Windows functionality (File, Display, Tools, and Help). See Figure 5.
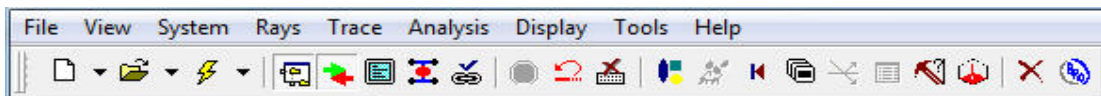

Figure 5 ASAP menus reflect the four-step workflow process: System, Rays, Trace, and Analysis

Step 1: Define and verify the system geometry

ASAP offers a variety of methods for conveniently entering your system geometry. Geometry can be constructed correctly with the aid of the ASAP native language, through its spreadsheet interface or domain-specific scripting and macro language, lens design program translators, or CAD translators. In this test drive, we will import a CAD IGES file from SolidWorks.

TRANSLATING CAD PROGRAM LANGUAGES

All CAD programs have their own languages; that is, their own way of representing geometries. Most use some form of a rational equation, such as a Non-uniform Rational B-Spline (NURB). ASAP also "speaks" these languages, but CAD file formats must be converted into ASAP format to optically simulate the geometry. This is done through a translation process.

Files can be translated in a variety of formats, and one of the most common is IGES (International Graphics Exchange Specification). However, ASAP also translates SolidWorks files saved in a GTX format. If SolidWorks is installed on the same machine as ASAP, it automatically uses the GTX plug-in to translate files from GTX format. The advantage of this translation route is that *native SolidWorks geometrical entities are translated directly into ASAP native geometrical entities without an intermediary process*. Conversely, using IGES or STEP (Standard for the Exchange of Product Model Data) requires that data from a CAD program is translated into one of these standards, and then translated again into your other application software. Every translation process introduces the opportunity for filtering or altering the original data, similar to making a photocopy of a photocopy, which can diminish the fidelity of the model. This should be avoided whenever possible.

BUILDING GEOMETRY IN SOLIDWORKS

NOTE If you do not have nor want to install a CAD program for this test drive, skip to the section, " "Importing CAD geometry into ASAP" on page 10 to translate CAD files directly in to ASAP.

We are now ready to start an ASAP design and analysis task, first by importing reflector geometry created in SolidWorks®, and then performing an analysis with an ASAP LED model.

Although SolidWorks is used for this demonstration, any CAD program that can export IGES files can be used. You can write ASAP geometry files from within SolidWorks, import XML files, or use BRO's proprietary smart-IGES™ system to import system models from any CAD package while maintaining fast, efficient ray trace speed.

1 Click to launch SolidWorks. The SolidWorks splash screen is displayed. See Figure 6.

Figure 6 SolidWorks splash screen

2 On the File menu, click Open (File, Open), and select the file, Base.SLDPRT. See Figure 7.
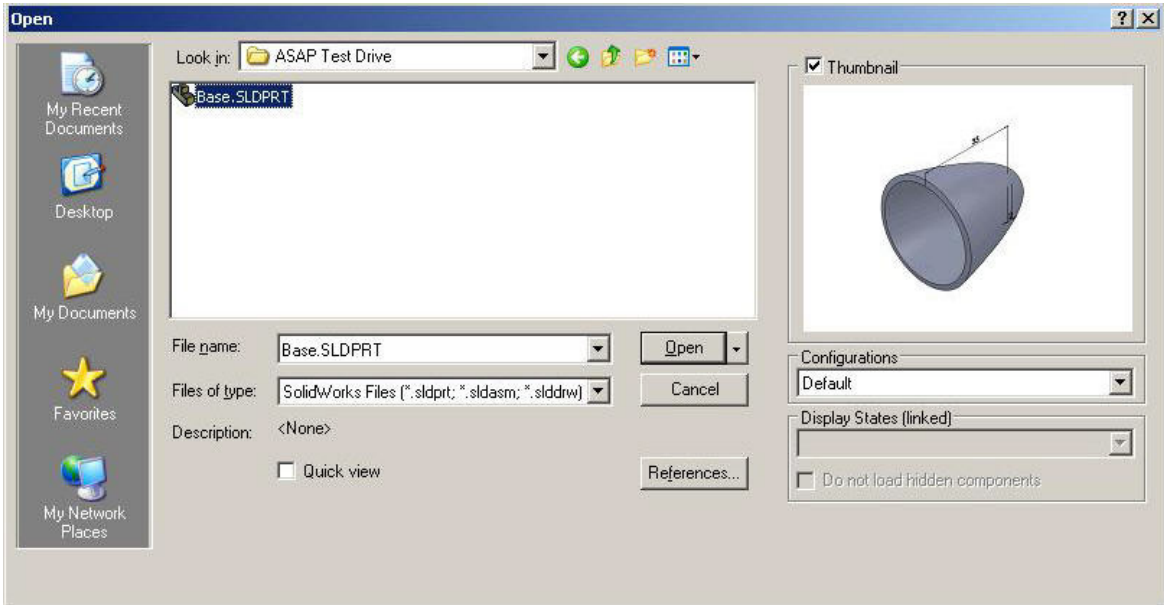

Figure 7 Opening the Base.SLDPRT file in SolidWorks

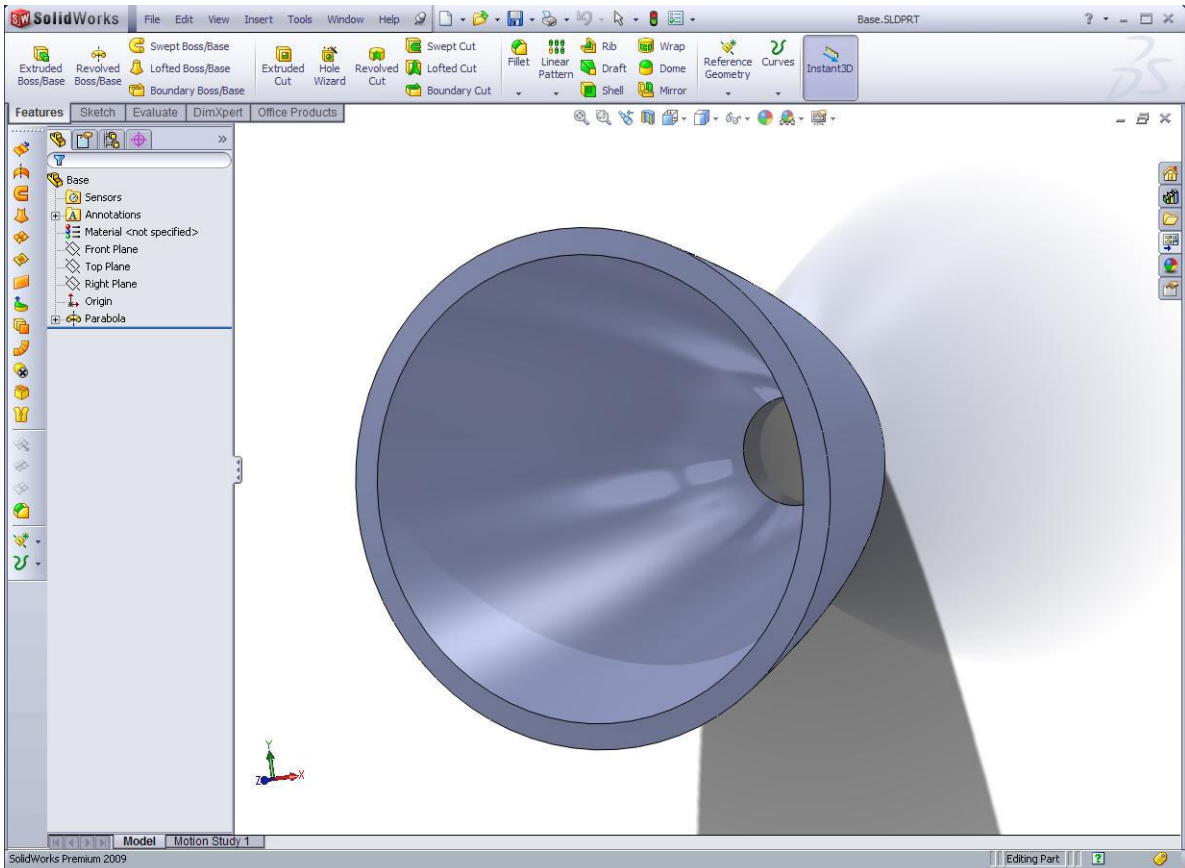**3** Click **Open** to display the file in SolidWorks. See Figure 8.

Figure 8 SolidWorks part displayed in the SolidWorks window

Since the focus of our test drive is ASAP, we will not go through the steps of creating a CAD file for import to ASAP.

4 Click File, Save As. For the **Save as type,** select **ASAP(*.gtx)**. Keep the default name "Base" and save the file as Base.GTX. Exit SolidWorks. See Figure 9.
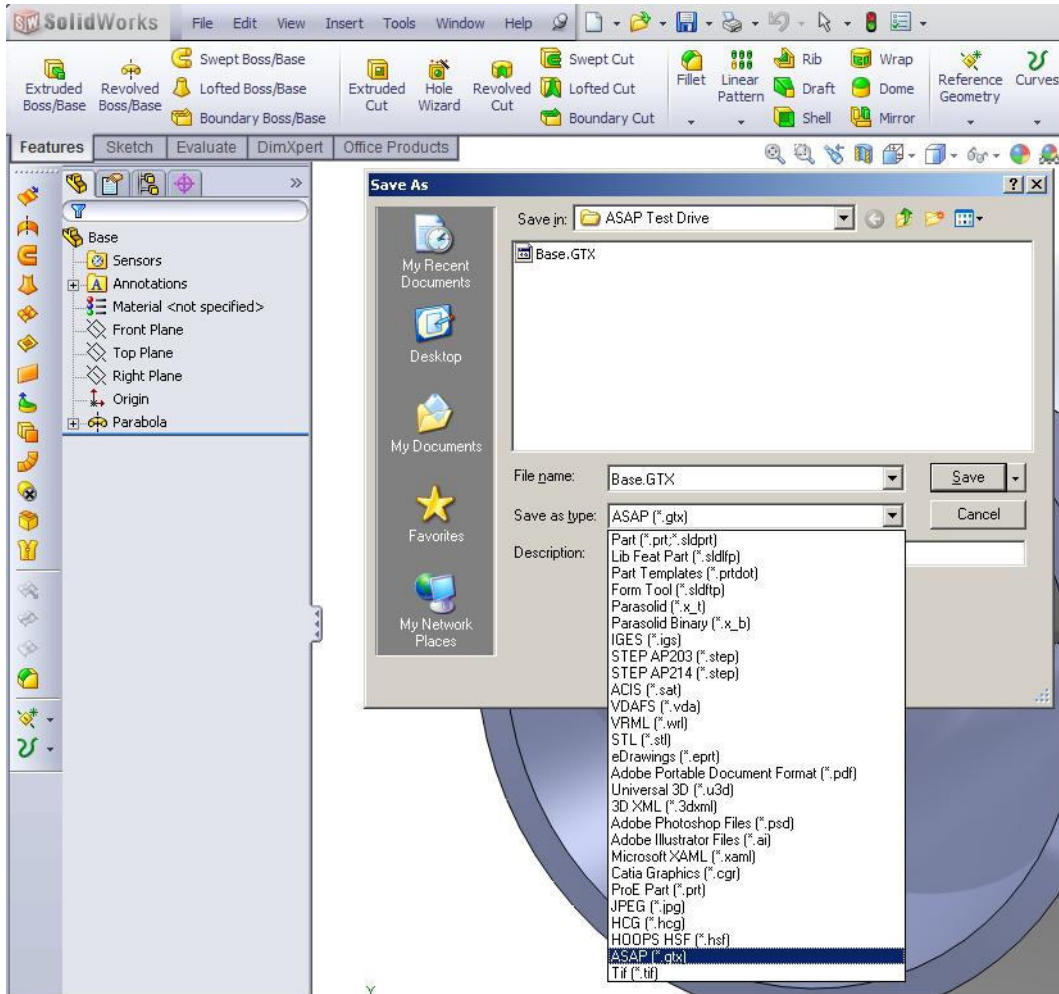

Figure 9 Saving the file in SolidWorks as an ASAP GTX file

IMPORTING CAD GEOMETRY INTO ASAP

The Base.GTX file is now ready to be directly translated into ASAP.

5 Switch your application focus to ASAP. From the main toolbar, click File, Open.

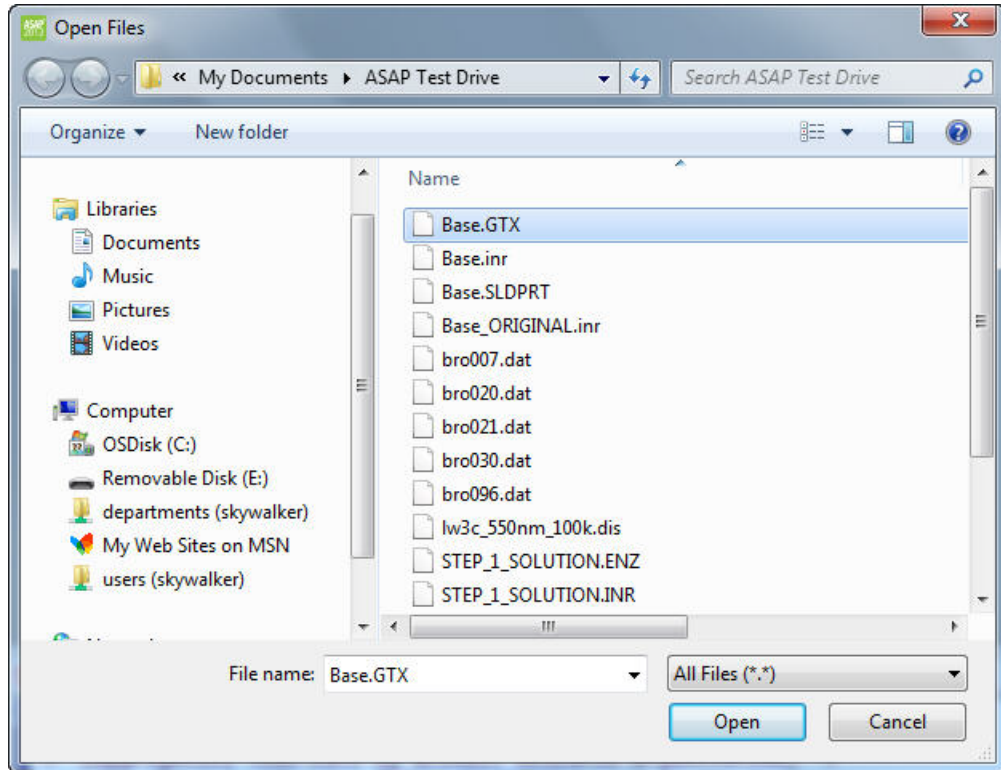**6** Click **Open** and click Base.GTX. See Figure 10.

Figure 10 Opening the SolidWorks file, Base.GTX, in ASAP

NOTE For Windows XP, click **Files of type** *on Open Files, and select* **cad files (*.gtx, *.catpart, *.catproduct)***. Confirm that the* **Open mode** *on Open Files is set to* **Auto** *so that ASAP knows it is opening a GTX file.*

CAD translator in ASAP

After you select the file, Base.GTX and click **Open**, GTX Setup Parameters is displayed first. It is used to set a wide variety of parameters for translation. The default parameters are suitable for our application, so do not change them. See Figure 11.
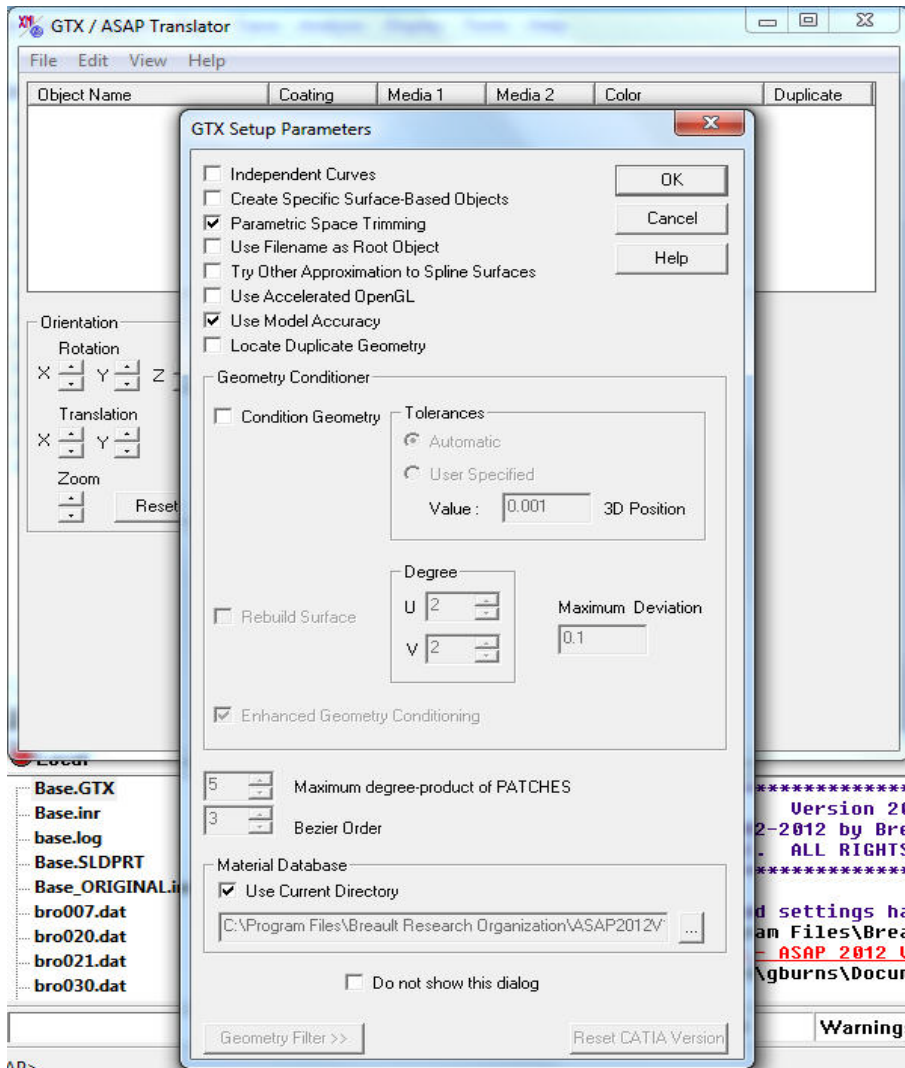

Figure 11 GTX Setup Parameters

> **1** Click **OK** on GTX Setup Parameters.

The translation is made and we can view, translate and rotate the geometry in the CAD/ASAP Translator.

2 To see all translated geometry, click the plus sign next to **Default**, and then the plus sign next to **Parabola** under the **Object Name** column. See Figure 12.
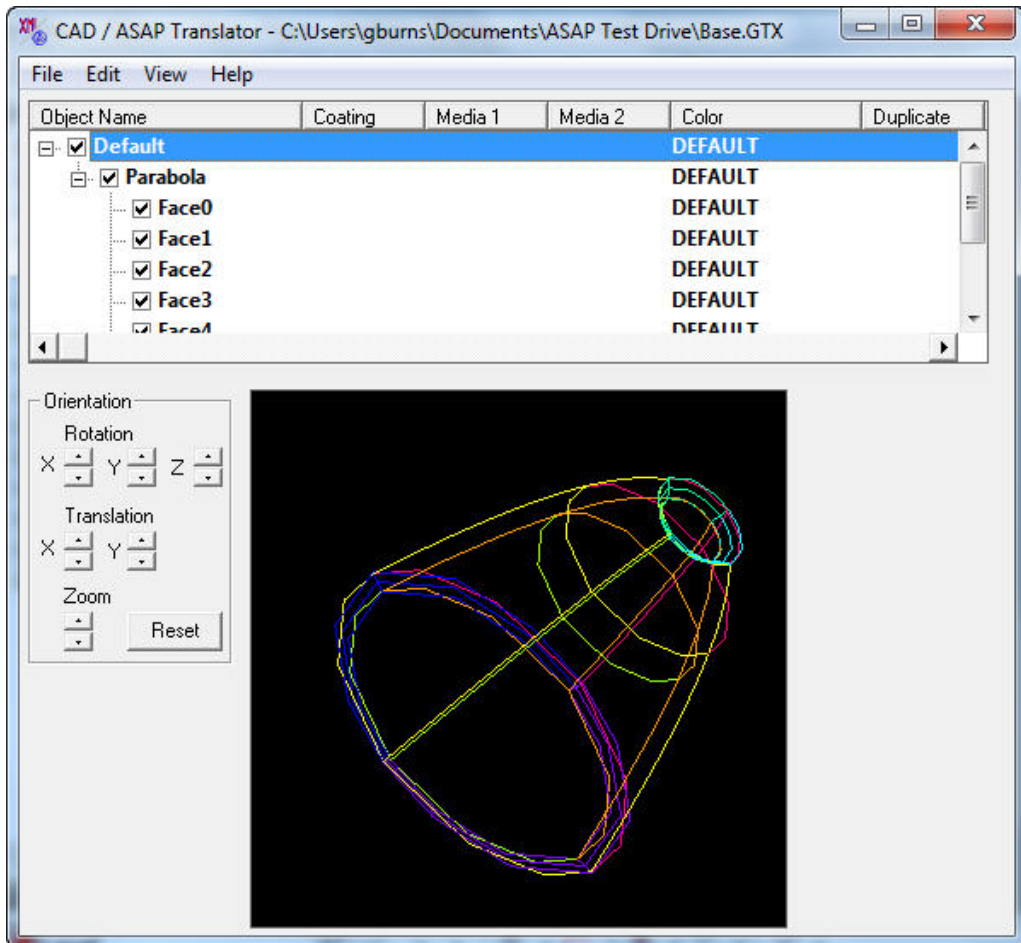

Figure 12 CAD/ASAP Translator

To view various angles of the geometry in the preview window, ASAP gives us two ways of moving the objects.

- Use the controls in the **Orientation** area to the left of the geometry window, or
- Simply place your cursor anywhere within the geometry window, right-click, hold down the right mouse button, and move the cursor.

By moving the cursor in any direction, you can manipulate the viewing angle for the geometry. This should give you a visual method of quickly inspecting and verifying the geometry is correct.

ASSIGNING OPTICAL PROPERTIES

Translating the geometry is only part of the process of setting up the optical system model. We must now assign optical properties to the geometry to account for phenomena-like reflection and refraction. This, too, is done through the CAD/ASAP Translator.

Note that in addition to an **Object Name** column, below the menu bar, there are columns for assigning coatings and refractive media on the sides of the objects. Within the translator, we can assign optical properties individually or as a whole to different components of the geometry.

NOTE An OBJECT in ASAP is a bounded physical geometrical entity with optical properties. We capitalize OBJECT here because it is an ASAP command, which are case-sensitive.

    3 Right-mouse click the first object name at the top of the geometry tree on the translator window to see a short-cut menu with the options shown in Figure 13.
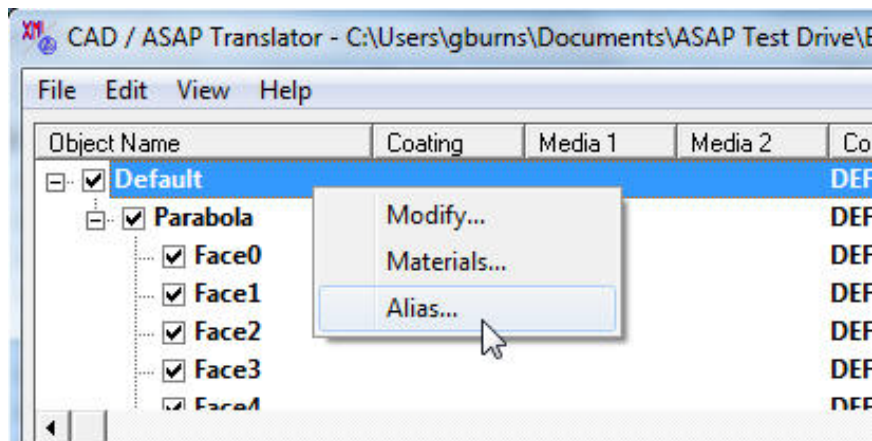

Figure 13 Shortcut menu for object names

The first menu option, **Modify** is for changing the optical properties of our geometry. The second option, **Materials** is for creating optical properties not found in the default optical properties database. The third option, **Alias** is for changing the name of the particular OBJECT. Changes such as optical properties or alias names made at a particular level are automatically applied to the branches in the lower levels.

4 Select **Alias** on the shortcut menu, change its name, "Default" to "REFLECTOR", as shown in Figure 14, and click **OK**.

Figure 14 Changing the object name using the Alias dialog box

The object name changes to REFLECTOR on the CAD/ASAP Translator.

5 Press the Ctrl key and select only **Face0** and then **Face6**.

In the CAD/ASAP Translator window, selected entities retain their original color, while unchecked entities are white. Note that you can highlight nodes on the tree or on the viewing window. The selected entities are the reflective interior of the parabola.

6 Right-click REFLECTOR, and this time click **Modify** on the shortcut menu.

7 On Modify Object, change the properties as follows: set reflectivity (REFLECT) of the OBJECTS to 100% with AIR as the refractive indices on either side of the interface, set the color of the

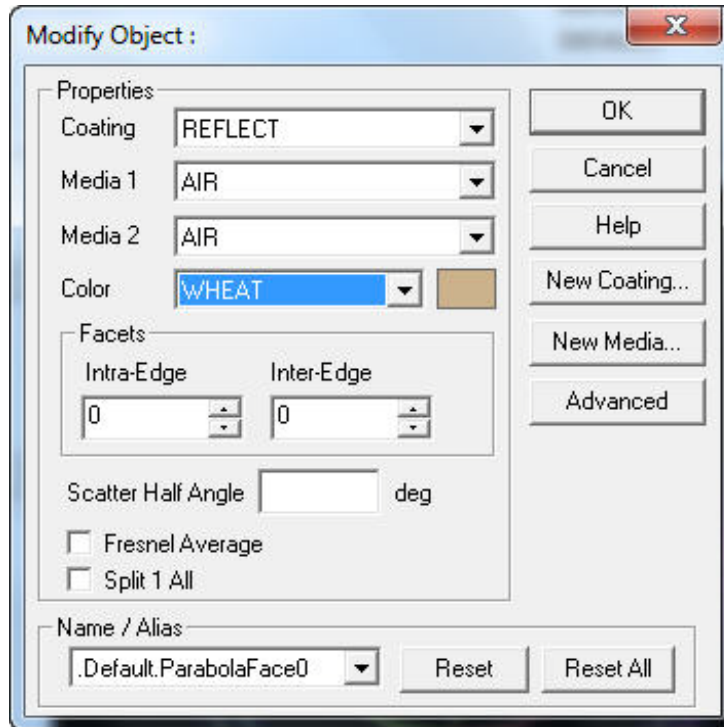reflective **OBJECT**s to WHEAT. See Figure 15, and click **OK**.

Figure 15 Changing the reflective interior of the reflector

8 Select only **Face1** through **Face5** and **Face7.** Right-click, select **Modify** on the short-cut menu to change properties on Modify Object.

The selected objects are the absorptive exterior of the reflector.

9 On Modify Object, set reflectivity for the selected objects to be totally absorbing (ABSORB) with AIR as the refractive indices on either side of the **INTERFACE**. Change the color of the

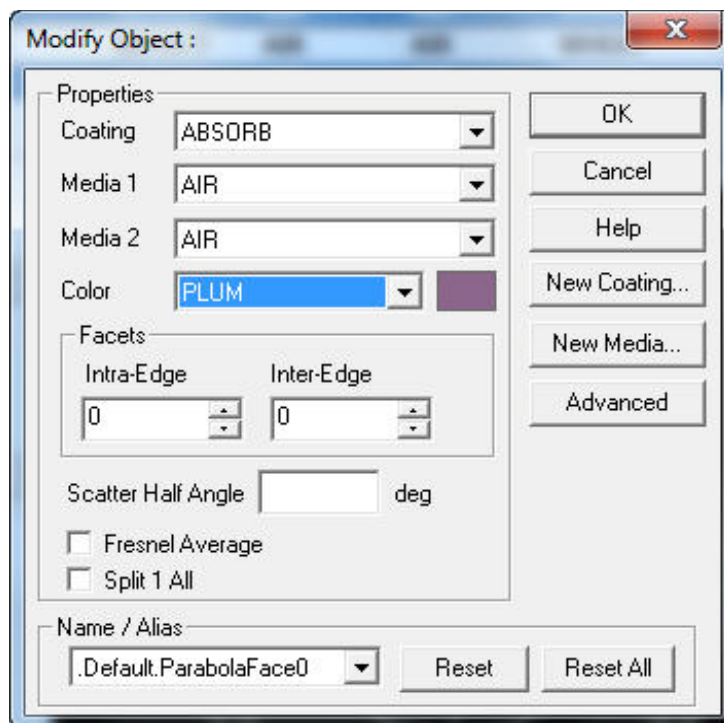absorbing OBJECTs to PLUM, and click **OK**. See Figure 16 and Figure 17.

Figure 16 Modifying the absorbing OBJECTs

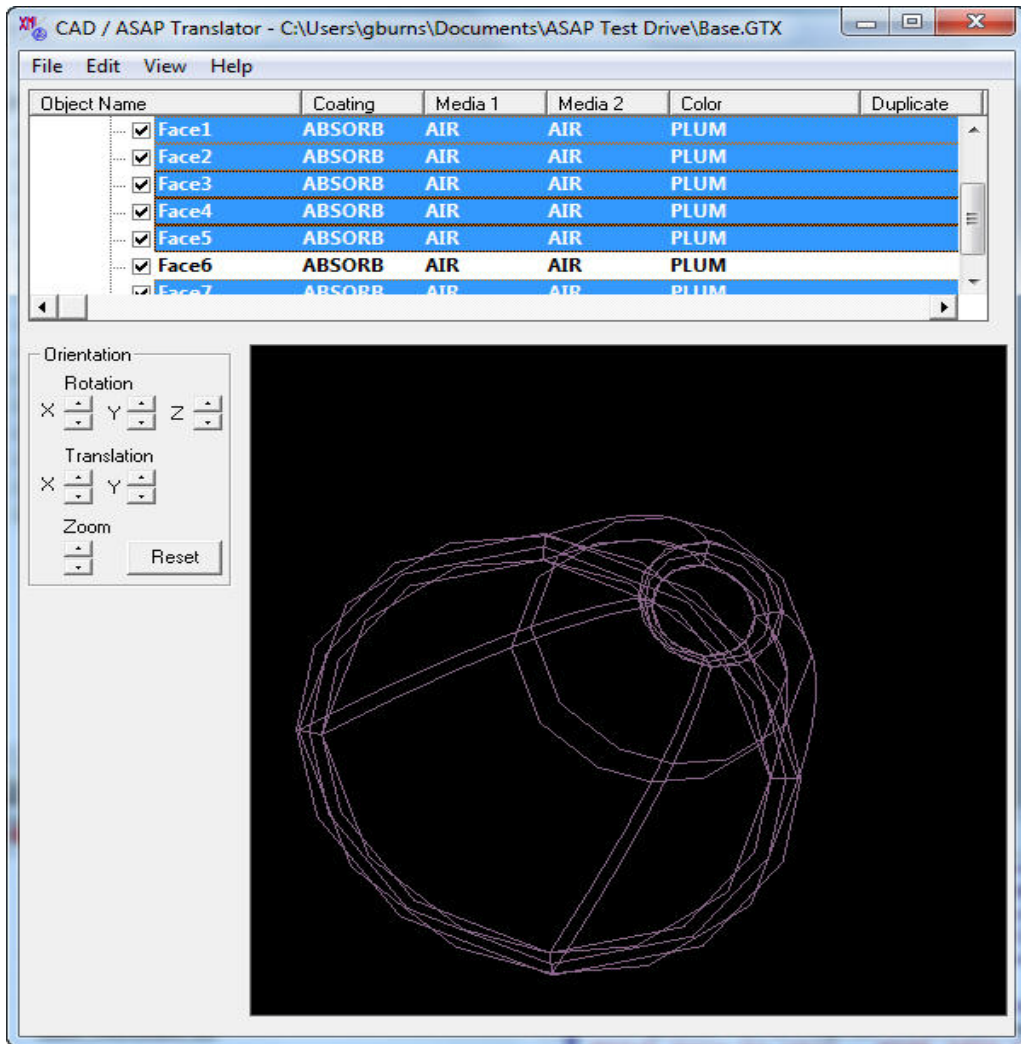Figure 17 Modified objects showing plum coloring

10 On the CAD/ASAP Translator, click File, Finish to complete the translation.
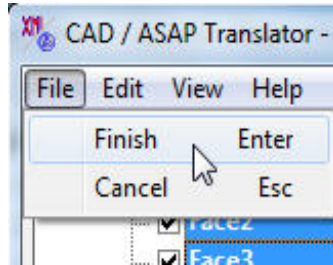

Figure 18 The last step to finish the translation

The CAD file is translated into ASAP, and the ASAP INR script file is displayed in the ASAP document window. The ASAP INR file name assumes the original CAD file name by default, and has an *.inr extension (for example, Base.inr). The file is listed on the ASAP Workspace window on the **Views** tab. The ASAP script is displayed in the document window, and is the imported translation result in ASAP of the geometry created earlier. See Figure 19.

Figure 19 ASAP INR file of the translated geometry (partial)

TIP Actual ASAP commands, but not comments, are capitalized. ASAP is a case-sensitive code whose default state recognizes uppercase command names. You can set the case sensitivity, but uppercase is a best practice in ASAP when referencing command names.

COMPLETING STEP 1 WITH THE ASAP BUILDER

The ASAP UI includes a spreadsheet-like Builder that can be used throughout its four-step workflow process. The Builder offers a simple approach to building your system model. Designed for new and experienced ASAP users, the Builder introduces you to the concepts of using ASAP to create your geometry, including sources. The entire system geometry is created in one window, using most standard spreadsheet navigation features, and saved with the file extension, ENX (an XML file) or ENZ (a compressed XML file). You can also use the Builder for tracing rays and analyzing your system. Please see the topic, "Builder Window Overview", in ASAP Help for more detailed information.

We are now going to use the Builder to complete the reflector geometry and verify that the geometry is correct.

11 On ASAP Workspace, right-click the **Builder** and click **New** to create a new Builder window. See Figure 20.
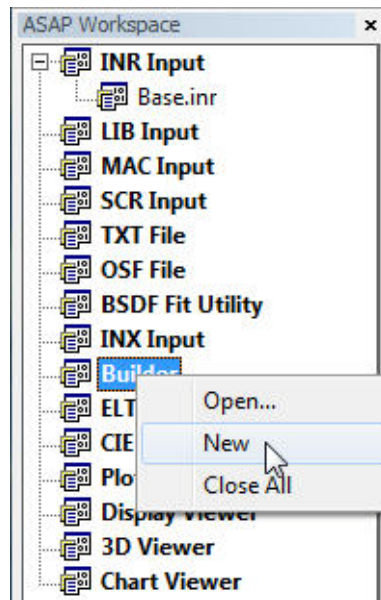

Figure 20 Creating a new Builder window on ASAP Workspace

A Builder window is displayed in the ASAP document window with its own toolbar.

12 Double-click a cell under the Type column to display a shortcut menu, and use your cursor to hover over specific options on the menu. For example, under **System** select Geometry> Surfaces> Biconic, as shown in Figure 21.

We can clearly see the four-step process listed on the initial shortcut menu on the Type column: **System**, **Rays**, **Trace**, and **Analysis**.
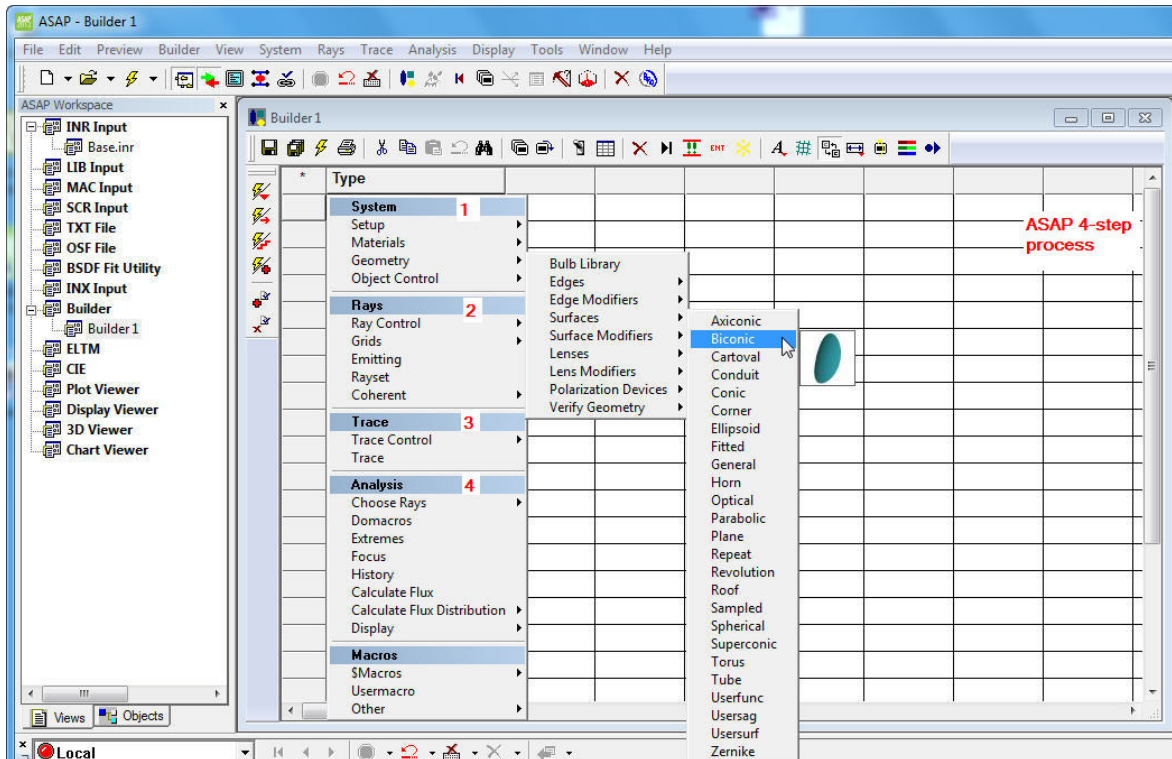
Figure 21 Selecting commands on the Builder window menu

We start our analysis by commenting our file, to record what is in the file. Clear, concise comments are essential to any work, and ASAP gives us an easy way to comment our files for future reference. ASAP commands in the Builder or script files that are proceeded by double exclamation marks (!!) are interpreted as a comment in ASAP.

13 Double- click the first cell in the Type column to display the shortcut menu, move the cursor down to **Macros** and click Other> Comment. See Figure 22.
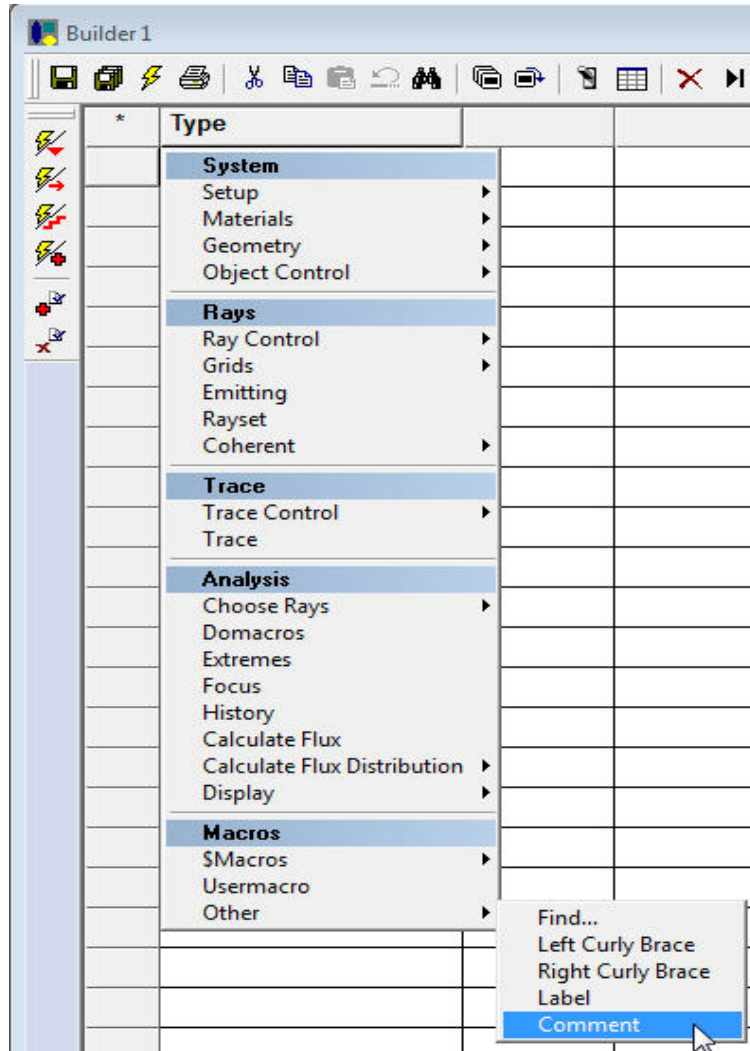
Figure 22 Setting up a comment using the a macro on the Builder shortcut menu

The cell in the Text column now displays "!!This is a comment".

14 To edit the comment, double-click the cell in the Text column to display the Command Editor, type comments to add to your file, and click **OK**. See Figure 23.

Figure 23 Typing a comment in the Command Editor dialog box

As with any process, you start by defining initial settings for your analysis. The most common starting point for settings is for the system **UNITS**.

15 In the Type column, double-click an empty cell, and under **System** click Setup> Units. See Figure 24.
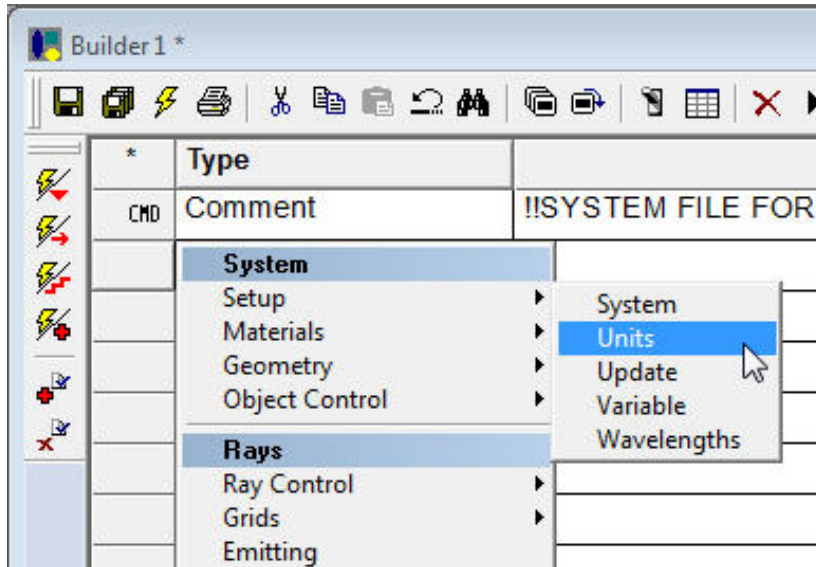
Figure 24 Setting system units

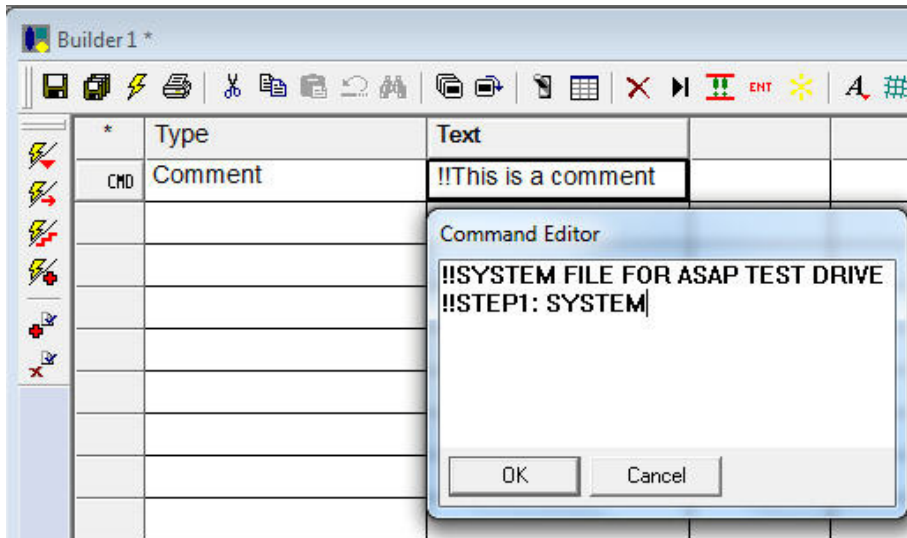The default for units is millimeters, which we want for our current system. We can also add a label identifier in the Flux Label column to indicate the flux or power units we are using for our analysis. This label is displayed only on plots and has no other computational significance.

16 Type WATTS in the Flux Label column. See Figure 25.

Figure 25 Setting Units as Millimeters, and adding a Flux Label of WATTS

Commands may be added by typing the command in the Type column.

17 To define a **WAVELENGTH** command, listing the wavelengths of interest for your analysis, double-click a cell under Units in the Type column, and under **System**, click Setup> Wavelength.

The **WAVELENGTH** command, along with **MEDIA** and **COATING** commands, define the specular dispersive optical properties of our optical system model. The **MEDI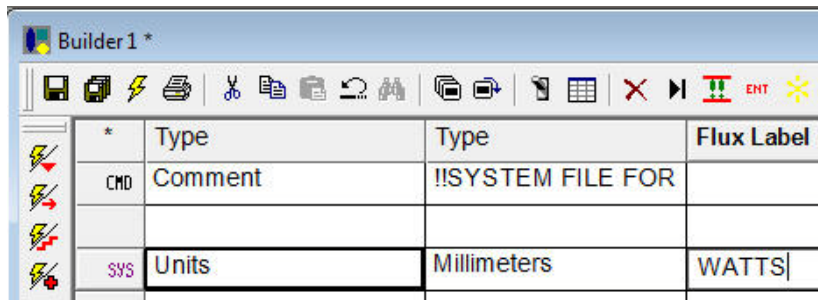A** and **COATING** information follows the **WAVELENGTH** command on the next row. For each **WAVELENGTH**, a corresponding refractive index exists in the **MEDIA** database, and reflective/transmissive pairs of coatings in the **COATING** database. For our example:

18 Assume a single wavelength of 550 nanometers. Following our standard procedure, double-click the cell in the row below Units, and under **System** click Setup> Wavelengths. Accept the default 550 nanometer for the wavelength.

Note that the column headings are dynamic. When the Comment cell is highlighted, the heading displays "Text". When "Units" is highlighted, the headings display "Type" and "Flux Label". When "Wavelength" is highlighted, the heading displays "Wave 1", "Wave 2", and so on. See Figure 26.

*	Type		Wave 1	Wave 2	Wave 3	Wave 4	Wave 5	Units
CMD	Comment	!!SYSTEM FILE						
SYS	Units	Millimeters	WATTS					
SYS	Wavelength		550					Nanometer

Figure 26 Column headings change dynamically to correspond to the command

Our main geometrical feature is the reflector. We already translated this into ASAP, and can now make it part of the Builder file by importing the file we created with the CAD/ASAP Translator (described in "Step 1: Define and verify the system geometry" on page 5).

19 Double-click the cell below Wavelength, and under **Macros**, click $Macros> I/O Control> $READ. See Figure 27.

*	Type			
CMD	Comment	!!SYSTEM FILE		
SYS	Units	Millimeters	WATTS	
SYS	Wavelength		550	

System
Setup ▶
Materials ▶
Geometry ▶
Object Control ▶

Rays
Ray Control ▶
Grids ▶
Emitting
Rayset
Coherent ▶

Trace
Trace Control ▶
Trace

Analysis
Choose Rays ▶
Domacros
Extremes
Focus
History
Calculate Flux
Calculate Flux Distribution ▶
Display ▶

Macros
$Macros ▶   Flow Control ▶
Usermacro   I/O Control ▶
Other ▶   Other Macros ▶

$Ask
$Copy
$Fast
$Ff
$Grab
$Io
$Move
$Plot
$Rcl
$Read
$Reg
$Scr
$Sto
$Sys
$View

Figure 27 Using the macro, $READ to import (read) the translated CAD geometry

20 Type the file name BASE.INR, which is the file we created during the CAD translation. See Figure 28.

*	Type	File Name							
CMD	Comment	!!SYSTEM FILE							
SYS	Units	Millimeters	WATTS						
SYS	Wavelength		550						Nanometer
CMD	$Read	BASE.INR							

Figure 28 Results of importing the translated file, BASE.INR

Using the comment capability, we can annotate our file for more clarity.

21 Right-click the cell, $Read to insert rows above or below it. See Figure 29.
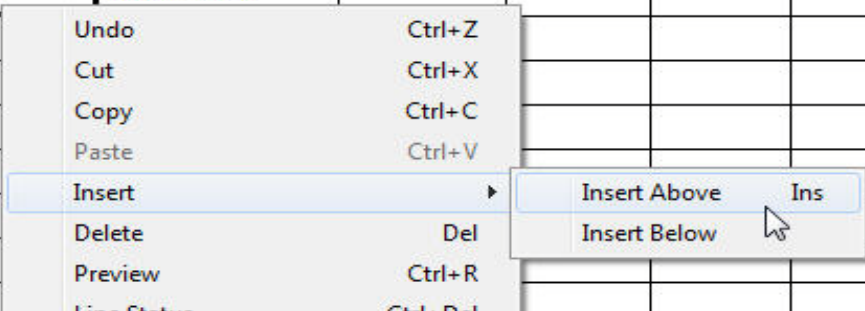

Figure 29 Inserting a row above $Read

22 In the Text column, use the Command Editor again to add the comment shown in Figure 30 (refer to Step 14 on page-22). You can view the whole text by hovering over the cell.

*	Type	Text		
CMD	Comment	!!SYSTEM FILE		
SYS	Units	Millimeters	WATTS	
SYS	Wavelength		550	
CMD	Comment	!!MAIN REFLECTOR GEOMETRY		
CMD	$Read	BASE.INR		

Figure 30 Hovering over the Text cell with the cursor to view the full comment

This is a good time to save our work, and it is a good habit to do this frequently.

23 Click ![save icon] the Builder toolbar, type the filename, STEP_1 in Save As, and click **Save**. The default file extension is *.enz. The file is saved to your Working Directory and is listed on ASAP Workspace, under Builder.

BUILDING ASAP NATIVE GEOMETRY

We are now ready to build some simple ASAP native geometry. For this example, we need to create an absorbing detector plane to collect rays for analysis. Since ASAP is an **OBJECT**-based code, we can enter geometry as:

- **SURFACES** or polynomial equation representation of geometry
- **EDGES** or **CURVES**, a parametric equation representation of geometry, or
- **LENSES**, a series of refractive and reflective conicoids describing traditional optical lenses and mirrors.

LENSES are geometrical entity types commonly found in lens design codes. The reflector geometry, created earlier in SolidWorks and translated into ASAP, is an example of a parametric representation of a geometrical entity, or an ASAP **EDGE** or **CURVE**. For this example, we will create a **SURFACE** type to demonstrate the ASAP **OBJECT**-oriented database structure.

As mentioned previously, ASAP **OBJECT**s are bounded geometrical entities with optical properties. The "bounded" part sets the size of the entity associated with the **OBJECT**. ASAP uses a single, global coordinate system to define system geometry, and we can define the entity size relative to its local coordinate system. Similarly, the optical properties associated with the **OBJECT** are assigned to the **OBJECT** through our previously defined optical properties databases, **MEDIA** and **COATINGS**.

For this example, we are setting up a simple **PLANE** as the detector for our reflector to demonstrate these concepts. As you have probably determined, ASAP uses a concise, logical, domain-specific language in its Builder and script files. ASAP command names actually have real meaning in the context of physics, mathematics, and optical engineering. There is no need to memorize abbreviations or cryptic mnemonics. A planar detector is created as a **PLANE**.

24 Double-click a cell in an empty row of the Builder. Under **System**, click Geometry> Surfaces> Plane. See Figure 31.
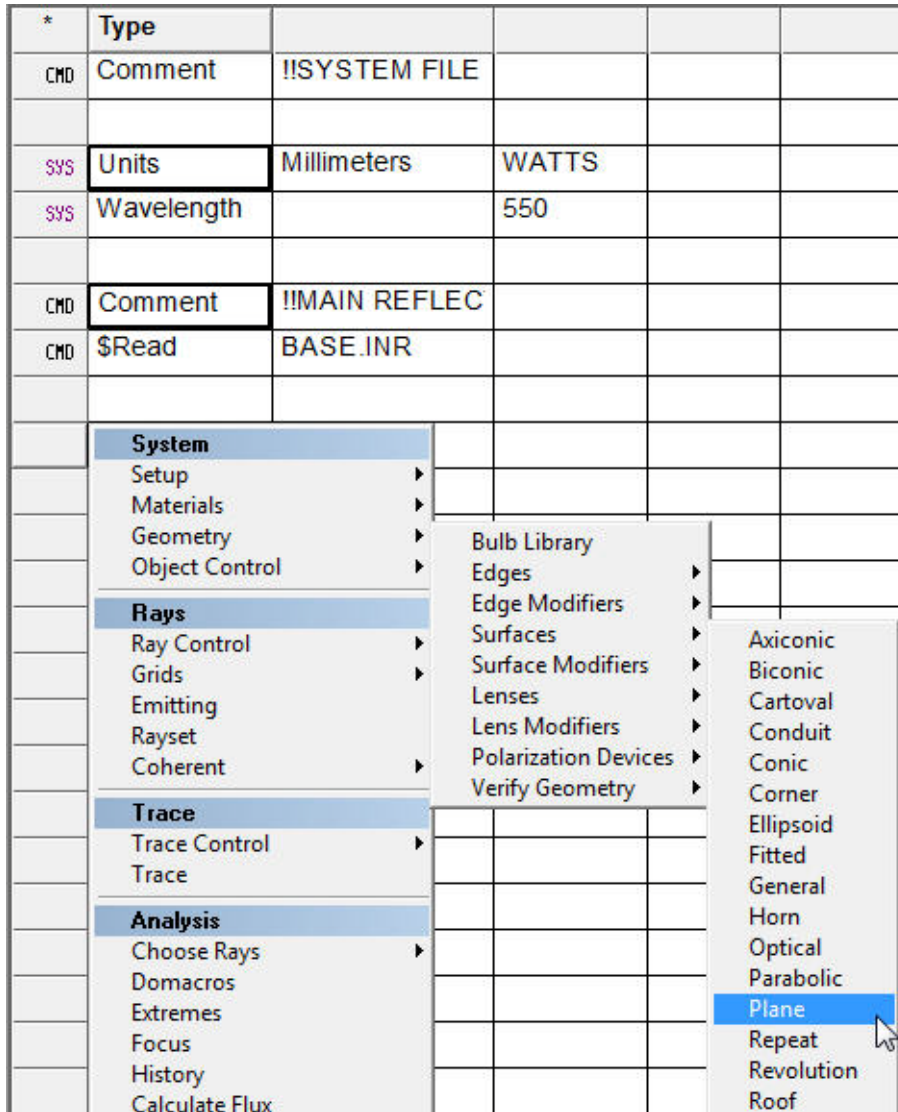

Figure 31 Inserting a Plane

25 Set the Z axis location to 250 by clicking the cell to the immediate right of the Z cell, in the Location column.

TIPS Always click the row you want to edit so that you can view the column headings that apply to that row. To view an entire row, you may need to use the horizontal scroll bar on the lower portion of the Builder.

26 In the Aperture column, double-click the cell and change the type to **Rectangle**. Remember to change the Semiwidth from 1.0 to 100 (millimeters) for this example. See Figure 32.

*	Type	Name	Option	Axis	Location	Aperture	Semiwidth
SYS	Wavelength		550				
CMD	Comment	!!MAIN REFLEC					
CMD	$Read	BASE.INR					
OBJ	Plane		Axis	Z	250	Ellipse ▾	1.0
						Ellipse	
						Rectangle	
						Hexagon	

Figure 32 Changing the Aperture to Rectangle and Semiwidth to 100

27 Right-click the Plane cell, and select **Entity Status** on the shortcut menu. This changes the status label in the column to the left of the cell from OBJECT (**OBJ**) to ENTITY (**ENT**). Changing the entity status from OBJ to ENT. See Figure 33.

*	Type	Na
SYS	Wavelength	
CMD	Comment	!!M
CMD	$Read	BA
ENT	Plane	

Figure 33 Changing the status from OBJ (object) to ENT (entity)

28 To switch this purely geometrical entity to an **OBJECT**, double-click the cell below the Plane entity, and under **System**, click Object Control> Object.
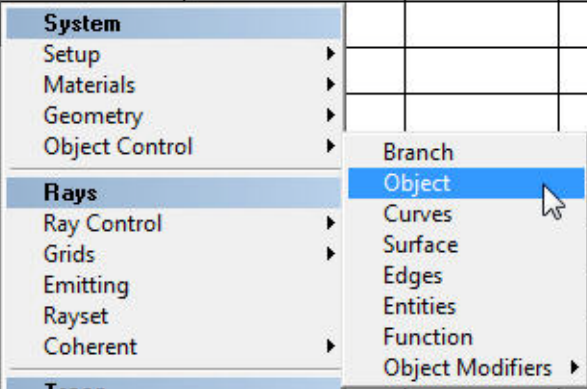

Figure 34 Selecting Object for the new row

29 Give the **OBJECT** the name, DETECTOR.

30 In the Entity column, type 0.1. This tells ASAP to grab the most recently defined entity, which in our case is the **PLANE**, and assign it to the **OBJECT**. Type DETECTOR in the Name column.

The Builder should now look like Figure 35.

*	Type	Name	Entity					
SYS	Wavelength		550					Nanometer
CMD	Comment	!!MAIN REFLEC						
CMD	$Read	BASE.INR						
ENT	Plane		Axis	Z	250	Rectangle	100	
CMD	Object	DETECTOR	0.1					

Figure 35 Builder after creating the OBJECT

VERIFYING THE OPTICAL SYSTEM

Now that we have created our basic optical system, we need to verify it graphically and numerically for correctness. The Builder has an easy method for doing this.

31 Click ▣ (Preview All) on the Builder toolbar.

ASAP uses the information to produce a simple graphical picture of the system geometry, which is displayed in the BRO 3D Viewer window. The geometry is temporarily sent to ASAP only to generate this picture. It becomes part of the database for analysis *after* the file is run.

In the 3D Viewer, we can rotate our system model by right-clicking and moving the mouse, zoom in or out, turn objects on or off, change shading features, and more. See Figure 36. For information about the 3D Viewer, please see the topic, "3D Viewer Overview" in ASAP Help.
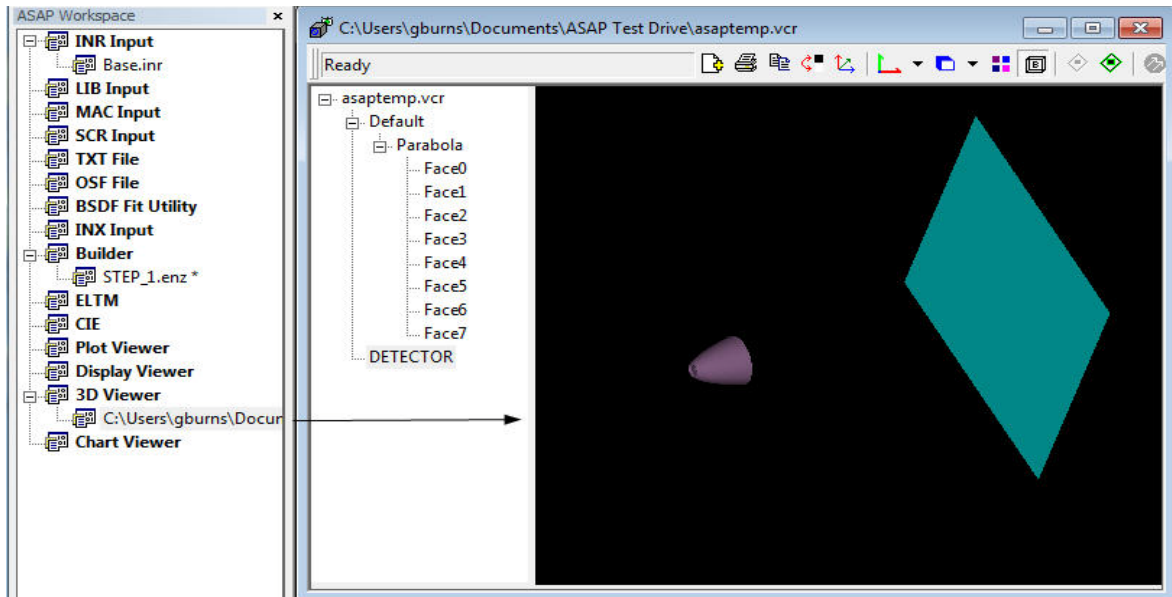

Figure 36 Viewing geometry in the 3D Viewer window (right), and the active file name listed in ASAP Workspace (left)

   32 Close the 3D Viewer file from ASAP Workspace by right-clicking the file node on the **3D Viewer**.
        Click **Close All**.

TIPS You could also close the 3D Viewer from the File menu. However, when using the Preview button, we would not have a record of our analysis, because the Preview button completes the entire task without showing you the ASAP commands. This is where the ASAP Builder and scripting language provide invaluable information directly in the file. Many other programs require users to navigate multiple menu levels and tabs to perform an analysis, but they do not give a history of your work. In ASAP, this history is automatically captured in the Builder and script files, allowing you to easily repeat your design and analysis almost effortlessly at a later time. There is no need to remember which menu items to select or tabs to navigate.

GRAPHICAL VERIFICATION

Now it is time to add these same graphical verification procedures to the Builder file.

33 After defining the DETECTOR, double-click an empty Type cell, and under **Macros** click Other> Comment. Type the comment, VERIFY SYSTEM GEOMETRY.

34 Set up a plot window by double-clicking an empty cell, and under **System**, click Geometry> Verify Geometry> Graphics> Window. See the red **A** marker in Figure 37.
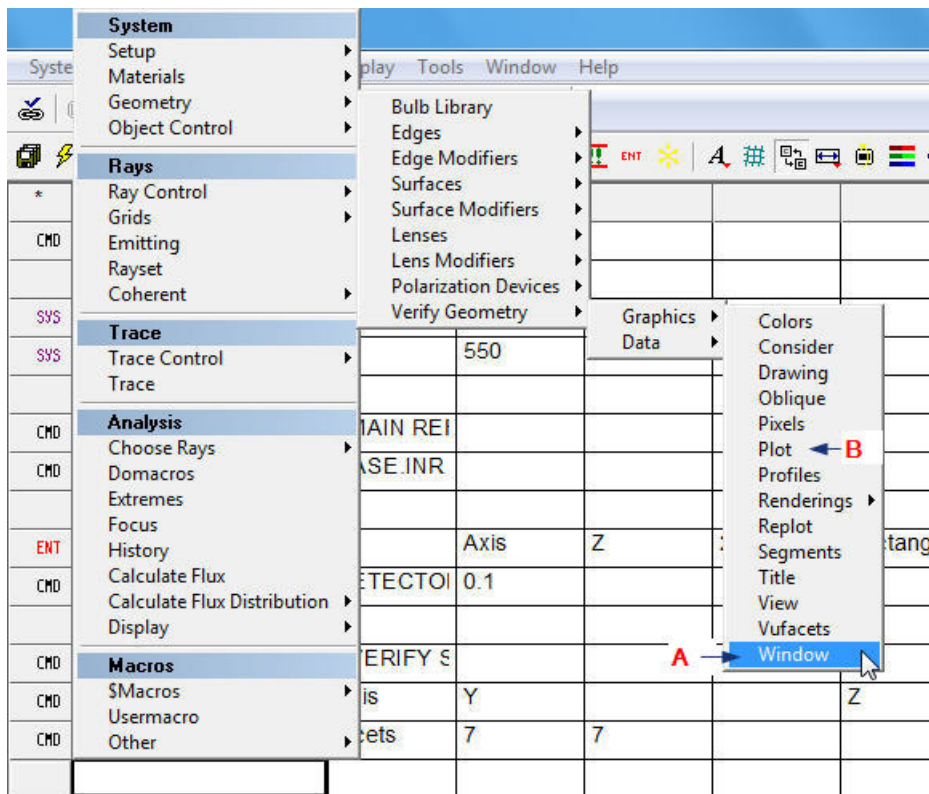

Figure 37 Adding plot commands to the Builder file

35 Replace the X axis with Z as the second axis (scroll right if necessary).

36 In the cell below the Window row, follow the same path, but click **Plot** this time, which is the option for **PLOT FACETS**, and is referenced by the red **B** in Figure 37.

37 Change the cell values from 5 to 7 for both Intra Facets and Inter Facets, and type 'SYSTEM GEOMETRY', including the single quotes, in the Title cell (scroll to the right to see this column).

38 Double-click the next row and under **Macros**, click $Macros> I/O Control> $View to generate a 3D view of the geometry. See Figure 38.

Figure 38 Builder command, $VIEW, to generate a 3D view of the geometry

39 Before running any files, save the Builder file by clicking ⌷ on the Builder toolbar.

40 Refresh ASAP by clicking the Restart Kernel button, ⌷ on the ASAP toolbar. The text in the Command Output window now states, "System database and settings have been re-initialized". See Figure 39.
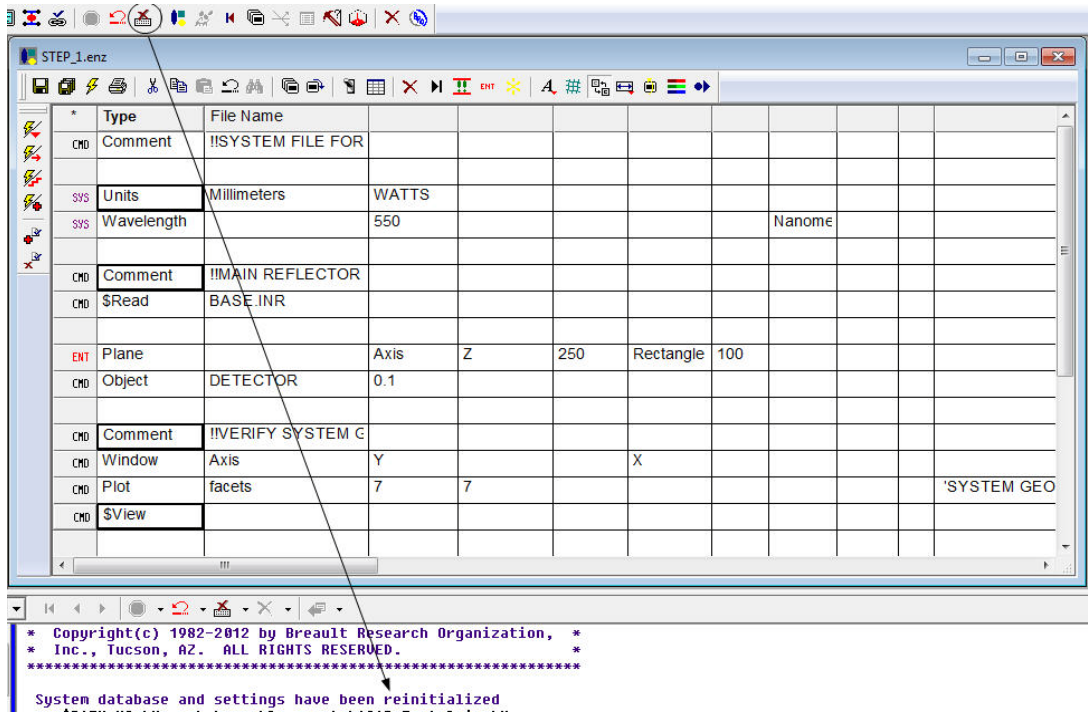

Figure 39 Refreshing ASAP with the Restart Kernel button

41 Run the Builder file by clicking ⌷ on the Builder toolbar.

ASAP runs the commands, loads the geometry and optical properties into its databases, plots a 2D picture of the geometry, and transfers that information to the 3D Viewer. See Figure 40 and Figure 41.
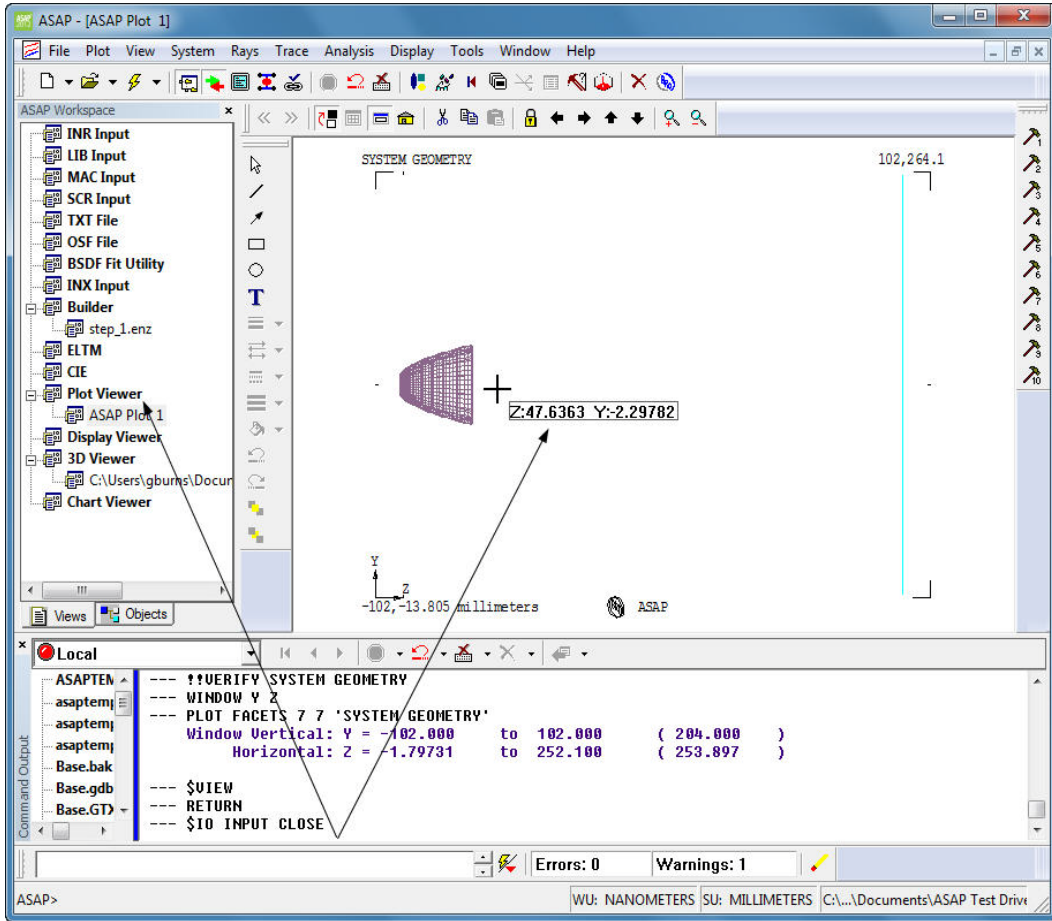

Figure 40 2D plot file results in the Plot Viewer document window

You can verify dimensions by pressing and holding the Shift key while moving the cursor over the 2D picture window. Rotate the geometry in the 3D picture window by clicking ⊞ on the 3D Viewer toolbar.

Figure 41 Results in the 3D Viewer document window

We have now completed Step 1, defining and verifying the system geometry. Although we have done this graphically, ASAP also has a variety of options for looking at the numerical values of our system geometry.

Step 2: Define and verify the source

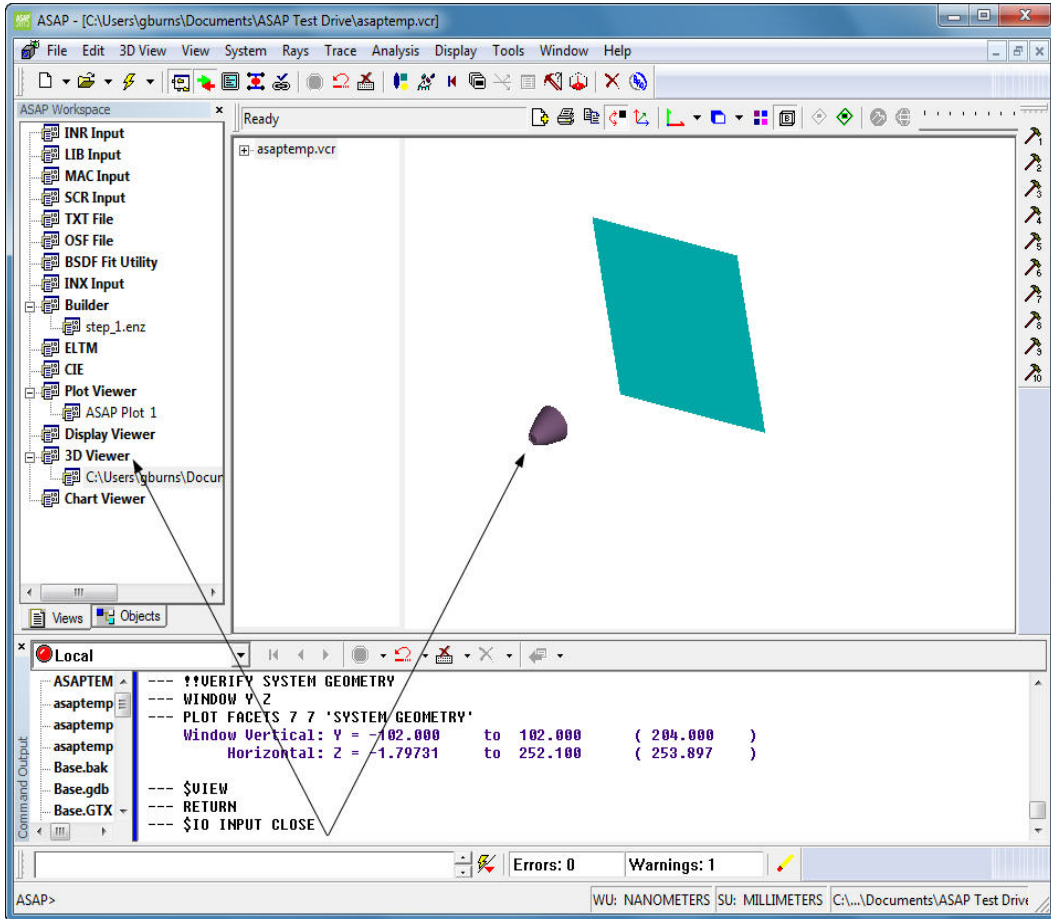ASAP can model a wide variety of sources, including point (coherent or incoherent) or extended with simple-to-complex geometries and behaviors. Imaging systems typically use point sources. The radiation or light from a point source emits from a single point.

Extended sources are made up of collections of point sources and are commonly found in illumination systems. In either case, *sources in ASAP are a collection of rays emitting from a surface or volume in space with specific properties that describe the near-field and far-field emission characteristics.* The near-field describes the source behavior very close or within 10 times the maximum source dimension, and the far-field emission characteristics describe the source behavior far from the actual source or beyond 10 times the maximum source dimension. The far-field emission properties are heavily influenced by the source geometry. We can use the extensive library of sources in ASAP, or use its powerful source modeling capabilities to create sources that are not in the library.

In our example, we will use a source from the ASAP Light Source Library, and create the source in a second Builder file. While the source can be put in the original Builder file, we want to illustrate the ASAP four-step process through the use of different files.

Since the files will be run sequentially, we need to perform a few tasks before we set up the source.

1 Open a new Builder file and save it as STEP_2. Do not overwrite the existing file, STEP_1.

2 Rewind the 3D Viewer file, which concatenates all the plot information created in an ASAP analysis.

In this next step, we want to look only at the source geometry; however, the 3D Viewer still contains the reflector and detector geometry.

3 To set up the Builder to rewind the 3D Viewer file, double-click the first cell in the new Builder file, and under **Macros** click $Macros> I/O Control> $Io. See Figure 42.
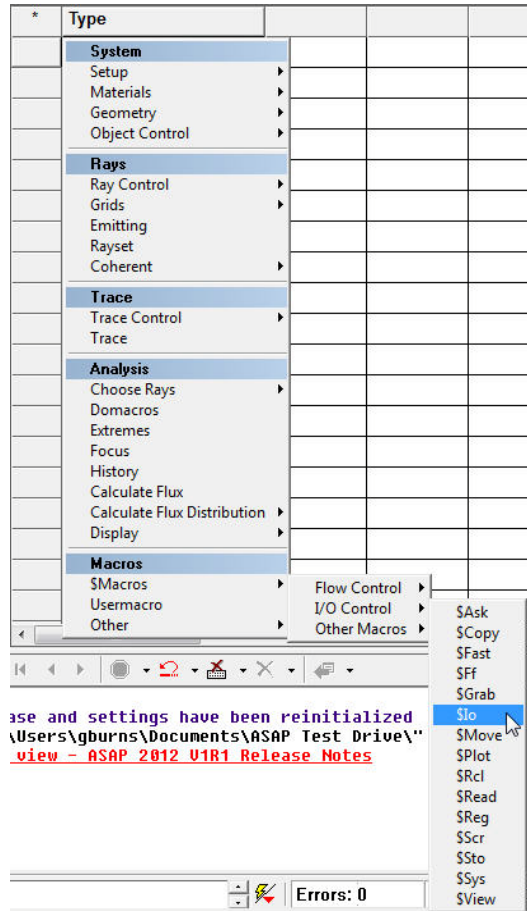

Figure 42 Setting commands to rewind the 3D Viewer file

4 Change the $Io row as follows: double-click the cell in the File I/O column and click **Vector** on the list; double-click the Option column and click **Operation**; double-click the Operation column and click **Rewind**. See Figure 43.

*	Type	File I/O	Option	Operation
CMD	$Io	Vector	Operation	Rewind

Figure 43 Setting Builder commands for rewinding the 3D Viewer file

CREATING AND VERIFYING THE SOURCE

We are now ready to create and verify our source, using a Lumileds LW3C LED. The Quick Start toolbar, located on the right side of the ASAP window, includes a list of all the filament, LED, CCFL, and arc sources from a variety of manufacturers that are included in ASAP. If the toolbar is not visible, click View, Quick Start Bar.

5 Click Sources> LED> Manufacturer> Lumileds and hover over the **LW3C** LED to see its basic specifications. This LED is a white light LED. See Figure 44.

*	Type	Bulb	X Pos	Y Pos	Z Pos	Primary Ax	Secondary	Copy	Kelvin
CMD	$lo	Vector	Operation	Rewind					
CMD	BulbLibrary	LT_W5SM	▲		0	+X	+Y	1	10000
		LUW_CP7P							
		LUW_E6SG							
		LUW_W5SM							
		LW3C							
		LW_E67							

Figure 44 Selecting an LED on the Sources tab of the Quick Start toolbar

6 Double-click **LW3C** on the Quick Start toolbar to display the Light Source Library Wizard. The wizard facilitates setting up sources for use in the Builder and script files. See Figure 45.
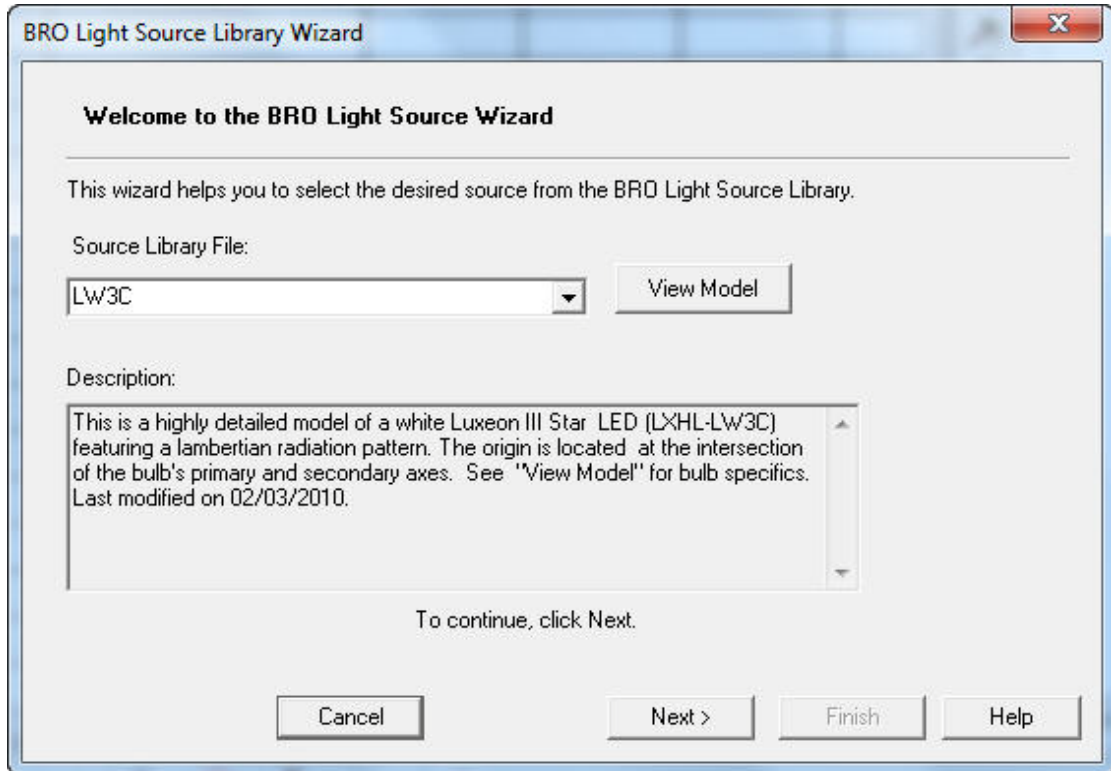

Figure 45 Light Source Library Wizard in ASAP, with description of LW3C LED source file

7   Click **View Model**, next to **Source Library File** on the Welcome page of the Light Source Wizard, to display specifications for the source file. See Figure 46.
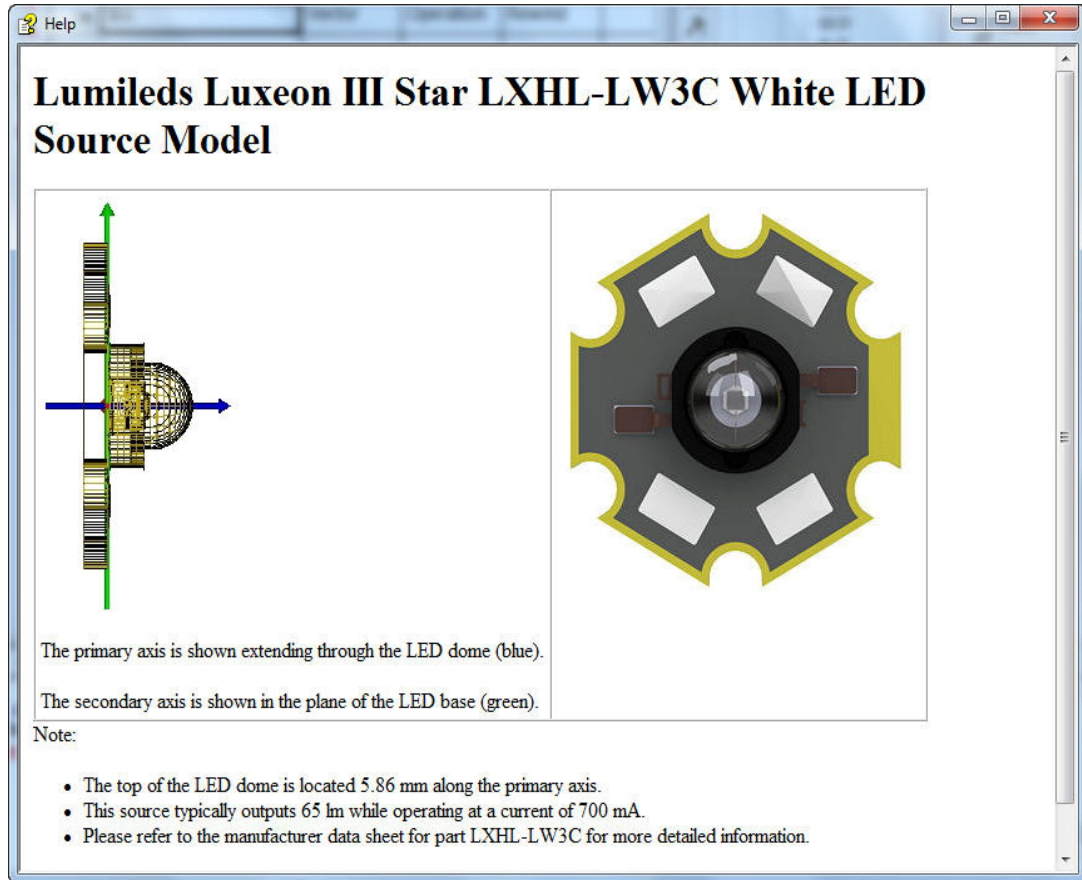

Figure 46 Viewing the model's specifications and images

The model you see is an actual rendering of the ASAP source. It displays the die geometry, including bond pads and wires as well as the external heat sink geometry. Sources associated with the Light Source Library in ASAP are highly accurate models that include the source geometry, optical properties, and near- and far-field emission properties (unless otherwise specified), which give these sources a high degree of fidelity and accuracy. In fact, library sources in ASAP have set the standard in the industry.

TIP When using sources from a library, it is critical to ensure that they are the maximum quality and fidelity. Many times, shortcuts are taken in the construction of sources that lead to inferior quality, usually to achieve higher quantity. The source is the most important component in, for example, an illumination system. A poor-quality source inevitably leads to a poor design.

Source creation via the Light Source Library Wizard is a two-step process:

• Creating rays for use with the source geometry.

• Creating ASAP commands that are used later in an ASAP template to generate the source geometry and assign the rays created in the first step to the geometry. (The Builder files do not need this template, only the rays created in the first step.)

8 Close the model's specification window by clicking the **X** at the upper right corner of the window.

9 Click **Next** on the Light Source Wizard to display the Rayset Creation Setup dialog box. See Figure 47.

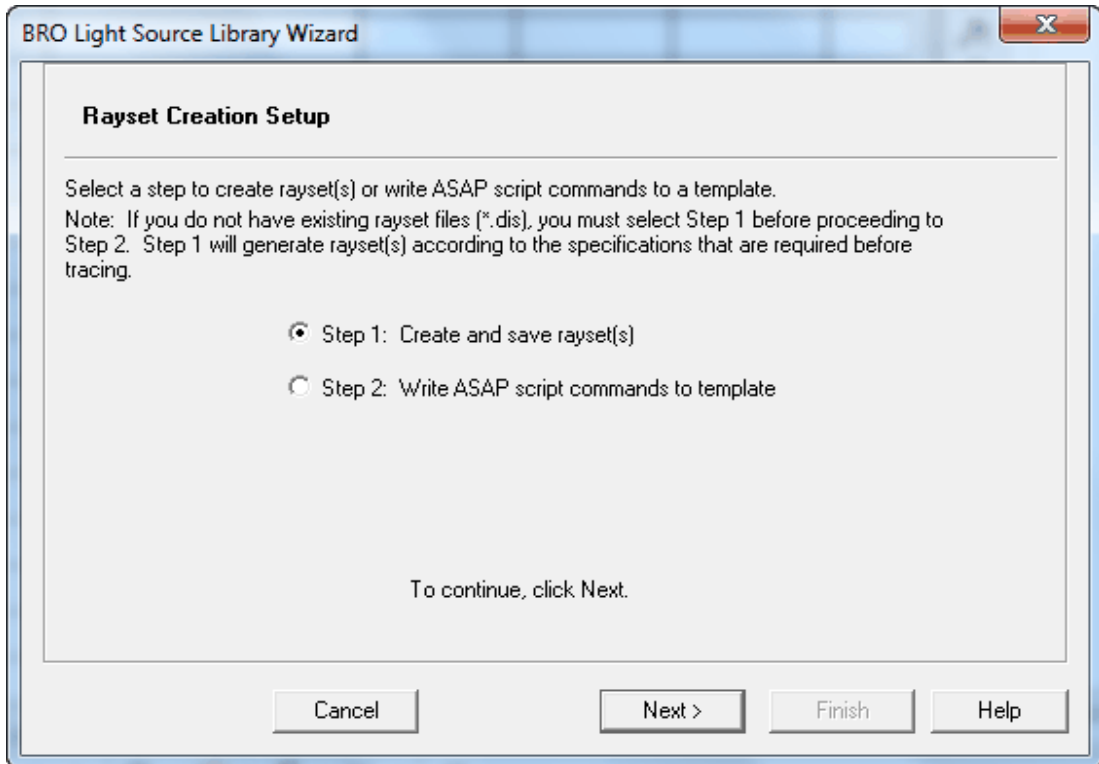**10** Select **Step 1: Create and save rayset(s)**. See Figure 47.

Figure 47 Setting up rayset creation in the Light Source Wizard

11 Click **Next** to continue.

Source Geometry Setup - Create Raysets is displayed.

ASAP models single **WAVELENGTH** or monochromatic sources with a single **WAVELENGTH**. A multiple **WAVE-LENGTH** source in ASAP, also referred to as a polychromatic source, consists of individual monochromatic sources with powers that are spectrally weighted according to the source power spectral density or PSD. We have the choice of creating a single **WAVELENGTH** source or a source with multiple **WAVELENGTH**s. In our simple example, we are assuming a single **WAVELENGTH** analysis.

12 Click **Single wavelength**, set the source **Wavelength** to 550 nanometers, and set the ray count in the table to 100000. See Figure 48.

Figure 48 Wavelength choices: monochromatic (single wavelength) or polychromatic (multiple wavelength)

13 Click **Finish**.

With these inputs, ASAP performs a number of calculations that you can see running in the Command Output window, resulting in a rayset file, lwc_550NM_100k.dis, to simulate the source. This file is included in the associated project file, bro4318_ASAPTestDrive_Project_Files.zip.

TIP When ASAP is calculating, the word, **Executing** *is displayed with a revolving line in the lower left corner of the ASAP window.*

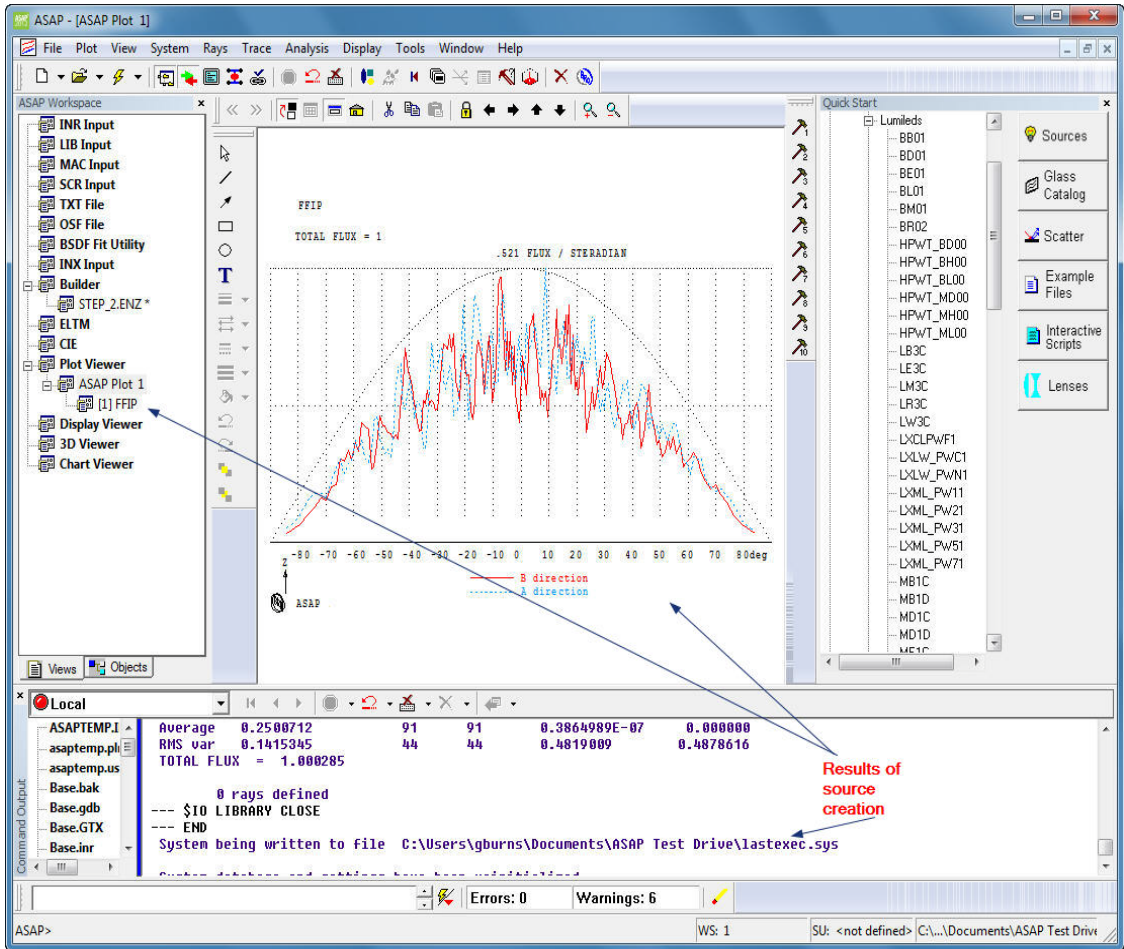The output also includes a simple intensity plot of the source on a polar plot diagram. See Figure 49.

Figure 49 Intensity plot output after rayset creation

After reviewing the plot displayed in Figure 49, we can go back to the Builder file that we recently started.

14 Click STEP_2.enz on the ASAP Workspace window on the left.

We are now ready to assign our source to the source geometry and verify it is correct.

15 Double-click in another cell, and under **System** click Geometry> Bulb Library.
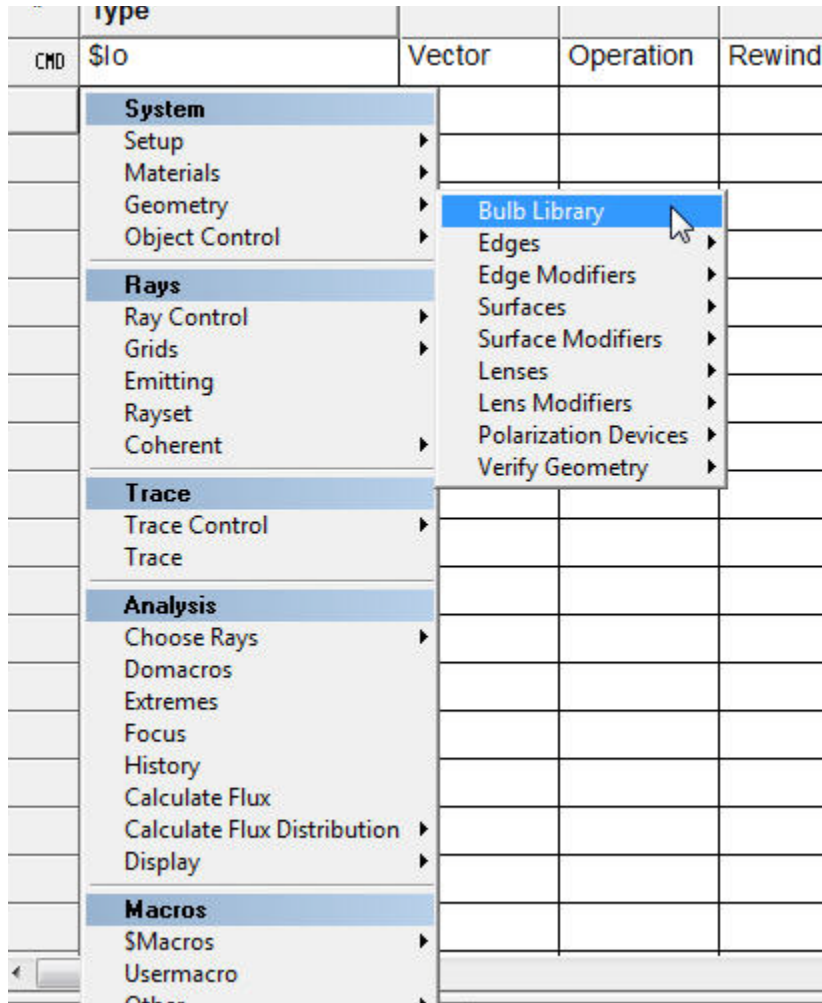
See Figure 50.

Figure 50 Assigning a source from the Light Source Library

It may seem counter-intuitive to create our source from the **System** area of the menu rather than the **Rays** area. This is because the source is first made of geometrical entities. The Bulb Library command is displayed in the Type column.

16 In the Bulb column: double-click the cell, locate **LED** on the tree, and click **LW3C**. See Figure 51.

*	Type	Bulb	X Pos	Y Pos	Z Pos	Primary Ax	Secondary	Copy	Kelvin
CMD	$lo	Vector	Operation	Rewind					
CMD	BulbLibrary	LT_W5SM LUW_CP7P LUW_E6SG LUW_W5SM LW3C LW_E67			0	+X	+Y	1	10000

Figure 51 Setting up the source - completed Builder row for the Bulb Library

17 Set the cell in the Z Pos column to -2.5, Primary Axis to +Z, and leave the Secondary Axis as +Y.

18 Set the Kelvin value, which is the source color temperature, to 0 since we are using a single wavelength source. In the Rays column, double-click and then click **PlaceRays**, and type the ray file name with its extension (lw3c_550nm_100k.dis) into the cell. *Make sure there are no additional spaces before the file name.* Set the Wavelength to 550.

*	Type	Bulb	X Pos	Y Pos	Z Pos	Primary Ax	Secon	Copy	Kelvin	Rays	Rays File	Wavelength
CMD	$lo	Vector	Operation	Rewind								
CMD	BulbLibrary	LW3C	0	0	-2.5	+Z	+Y	1	0	PlaceRays	lw3c_55nm_100k.dis	550

Figure 52 Settings for the LW3C LED

Now, we can enter commands to see the LED geometry and plot some rays from the LED, setting up a window just like in Step One and a `PLOT FACETS` command. However, this time we will `OVERLAY` on our plot facets a sampling of rays and their ray directions with respect to the LED geometry.

19 Double-click a cell below the source definition, and under **System** click Geometry> Verify Geometry> Graphics> Window to set your plot window. Set both plot axes to Y and Z.

20 In the next row in the Type column, double-click the cell and under **System**, click Geometry> Verify Geometry> Graphics> Plot. Select the **Overlay** option in the cell in the Overlay column. This instructs ASAP to overlay the next plot on top of this one. In the Title column, type the title with single quotes, 'SOURCE GEOMETRY WITH PLOT RAYS'.

21 In the next row, double-click the cell in the Type column, and under **Rays** click Ray Control> Verify Rays> Graphics> Plot. Double-click the Cmd column and click **rays**. Set the ray Scale Factor to 3 and the Ray Number to 1000.

This scales the ray vectors to a reasonable size and only plots every 1000th ray. Remember, we created a ray set with 100,000 rays and we do not need to see every one of them plotted!

22 In the next row, in the Type column, double-click the cell, and under **Macros** click $Macro> I/O Control> $View.

The Builder file should look like Figure 53. If it does not, delete lines, re-enter the commands, and run the file again.

*	Type	File Name										
CMD	$Io	Vector	Operation	Rewind								
CMD	BulbLibrary	LW3C	0	0	-2.5	+Z	+Y	1	0	PlaceRays	lw3c_555nm_100k.dis	550
CMD	Window	Axis	Y			Z						
CMD	Plot	facets	5	5				Overlay		'SOURCE GEOMETRY V		
CMD	Plot	rays	3	1000								
CMD	$View											

Figure 53 STEP2_ENZ Builder file showing source and output commands

23 Remember to do these best practices: save changes to the Builder file; click the Kernel Restart button, ![icon] to refresh ASAP for the next phase; and click ![icon] to run the file.

ASAP creates the source geometry and rays, plots a 2D picture, and displays the results in the 3D Viewer. See Figure 54 and Figure 55.
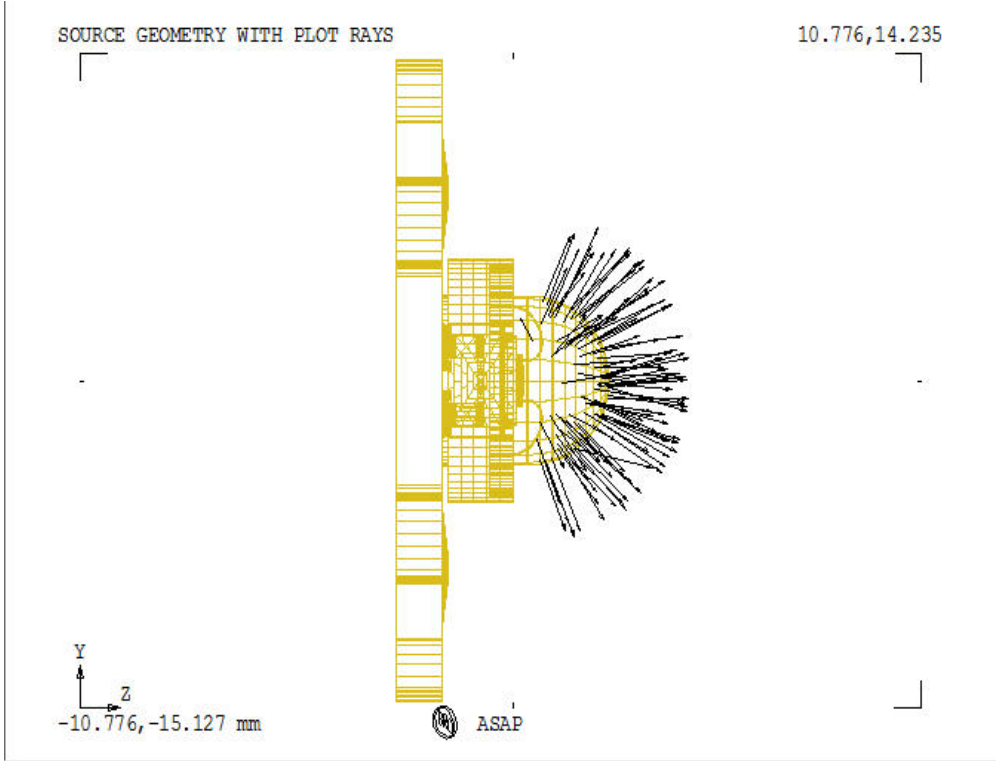

Figure 54 2D picture of the LED in ASAP

24 Click the dome in the first 3D view of Figure 55, and click View Mode, [icon]. Change the view to "wireframe" and zoom in to enlarge.
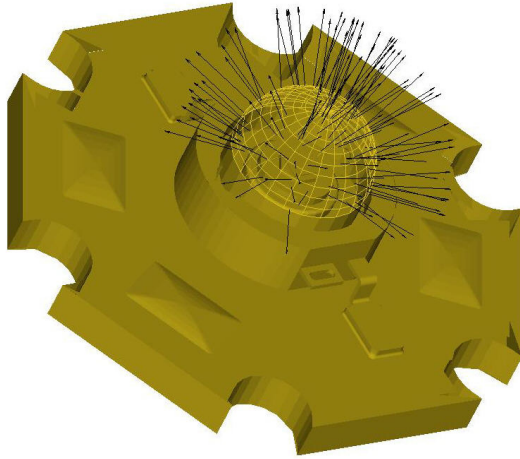

Figure 55 3D pictures of the LED in ASAP

Step 3: Trace rays

Step 3 involves tracing rays through our system while viewing the ray trace.

1 Start a new Builder file and name it STEP_3.

2 To begin the file, "rewind" the 3D Viewer file by double-clicking the first cell in the new file. Under **Macros**, click $Macros> I/O Control> $Io.

3 Change the column entries: In the File I/O column, double-click and click **Vector**; in the Option column, click Operation. In the Operation column, double-click and click **Rewind**.

4 Set up your window: in the next row, double-click the cell in the Type column. Under **System**, click Geometry> Verify Geometry> Graphics> Window. Change the plot axes to Y and Z.

5 In the cell in the row below, double-click, and under **System** click Geometry> Verify Geometry> Graphics> Plot. Click the **Overlay** option from the Overlay column to instruct ASAP to overlay the next plot on top of this one. Add the title including single quotes, 'SYSTEM, SOURCE AND RAY TRACE'.

6 In the next row, click **Trace**, click **Plot** in the Plot column, and type 1000 in the Plot Rays column.

Optionally, we will add an additional 3D Viewer command to see the results in the 3D Viewer:

7 Double-click the next lower cell, and under **Macros**, click $Macro> I/O Control, $View. The Builder file should look like Figure 56.

*	Type	File Name									
CMD	$Io	Vector	Operation	Rewind							
CMD	Window	Axis	Y			Z					
CMD	Plot	facets	5	5				Overlay			'SYSTEM, SOURCE
CMD	Trace							Plot	1000		
CMD	$View										

Figure 56 Preparing to trace rays in Step 3

We are almost ready to look at the cumulative results of our work. First, we must open and run the previous two files, since they contain our reflector geometry and source.

8 Click the Kernel Restart button, ⌨. Open the files in this order: STEP_1.enz, STEP_2.enz, and STEP_3.enz, and then run each one consecutively by clicking ⚡.

After running the STEP_2 file, the pictures look slightly different than when the file was run by itself; this is because the pictures now contain the reflector geometry. Only the 2D ray trace file and 3D Viewer of the ray

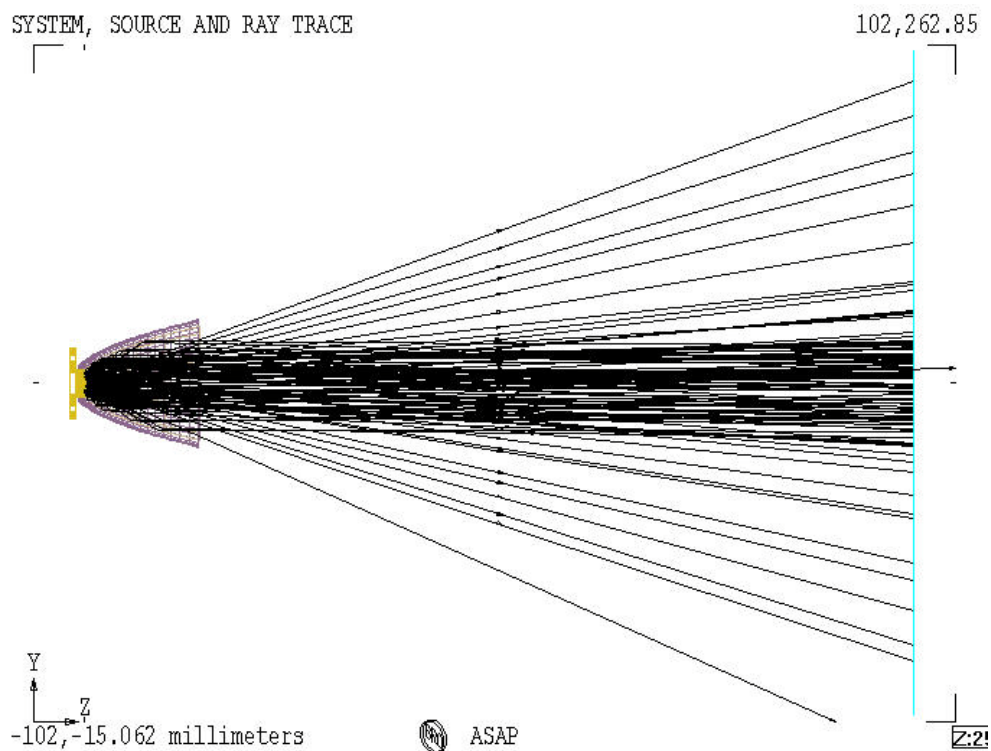trace are shown here. See Figure 57 and Figure 58.

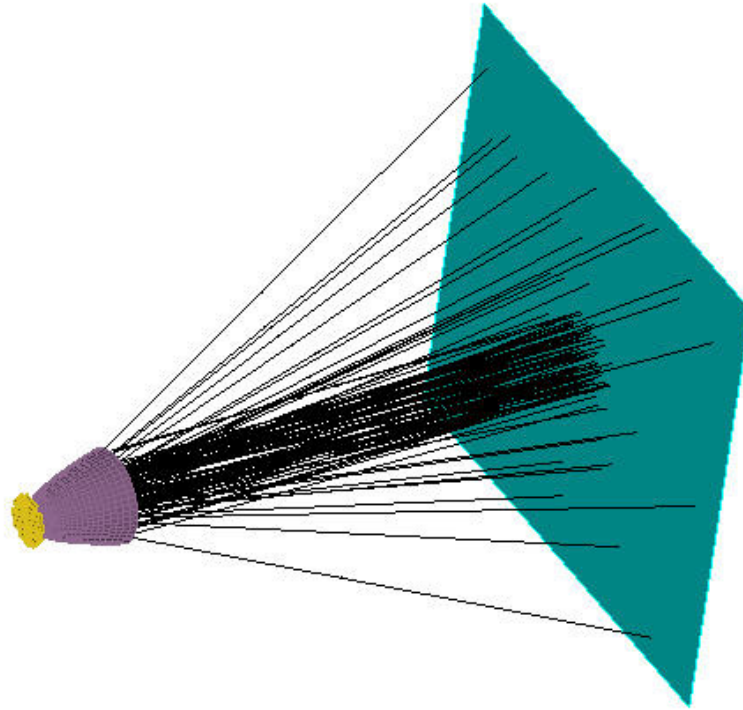Figure 57 2D plot of system, source, and ray trace in ASAP

Figure 58 3D plot of system, source, and ray trace in ASAP

Step 4: Analyzing the results

We are now ready to analyze the results of your ray trace.

1. Start with a new Builder file and name it STEP_4.

2. Rewind the vector file by double-clicking the first cell in the new file. Under **Macros**, click $Macros> I/O Control> $Io.

3. Change the column entries to **Vector** in the File I/O column, **Operation** in the Option column, and **Rewind** in the Operation column.

When rays are traced in ASAP, those rays propagate until they reach an **OBJECT** and cannot propagate to another **OBJECT** or they stop on an absorbing **OBJECT**. Rays normally stop on absorbing **OBJECT**s. The fundamental concept is that *rays are always associated with an* **OBJECT**. We are interested in calculating the irradiance pattern at the detector plane, so we isolate rays on that **OBJECT** by considering only that **OBJECT** for analysis. The **CON-SIDER** command is used to turn **OBJECT**s on or off.

4 Double-click the cell in the next row, and under **Analysis**, click Choose Rays> Consider. In the Option column click **Only**. In the **Object 1** column type DETECTOR to select only the **OBJECT** named DETECTOR for analysis.

We want to compute the irradiance pattern at the detector. *Irradiance is defined as the flux per unit projected area* or operationally:

$$E = \frac{d\phi}{da} = \frac{\text{differential power}}{\text{differential area}}$$

Equation 1

Physically, the computation involves counting the number of rays and their associated fluxes or power in an area and dividing by that area. We must have a way of setting up an area for the calculation, which is why the **WINDOW** command is used.

5 Double-click the cell in the next row, and under **Analysis**, click Calculate Flux Distribution> Window. The window axes of Y and X are okay.

We will use the **PIXEL** command to set the differential area by defining the number of power-counting bins within the **WINDOW**.

6 Double-click the cell in the row below the **WINDOW** cell, and under **Analysis**, click Calculate Flux Distribution> Pixels. Change the default number of vertical pixels (39) to 51.

Now, we are ready to calculate the irradiance.

7 Double-click the cell below the **PIXEL** definition, and under **Analysis** click Calculate Flux Distribution> Spots.

This command calculates a spots diagram resulting in an irradiance calculation.

8 Click **Position** in the Option column, **Attribute** in the Attribute Flag column. In the Attribute column, type 0.

This option creates an irradiance distribution file that we can view, which was created by the **SPOTS POSITION** command. We do not need to look at nearly 100,000 dots on the screen as displayed in the standard spots diagram.

9 After you make the basic calculation, initiate the file viewing by double-clicking in the cell below the **SPOTS** command, and under **Analysis**, click Display> Display.

Now, we will **AVERAGE** the results:

10 In a new row, under **Analysis**, click Display> Processing> Average.

11 View a picture of the results by double-clicking in the next cell down, and under **Analysis**, click Display> Graphics> Picture. Click ⚡.

The Builder file for STEP_4.enz should look like the following. If it does not, just delete lines and re-enter the commands. See Figure 59.

*	Type					
CMD	$Io	Vector	Operation	Rewind		
CMD	Consider	Only	DETECTOR			
CMD	Window	Axis	Y		X	
CMD	Pixels	51	1	Off		
CMD	Spots	Position	POSITION		Attribute	0
CMD	Display					
CMD	Average	1	1			
CMD	Picture					

Figure 59 Step 4, Analysis commands, in the Builder

If you have not re-initialized ASAP from running the previous three files, run STEP_4.enz. If you have re-initialized ASAP, run the previous three files sequentially, followed by this file. The result of the irradiance distribution is shown in Figure 60.
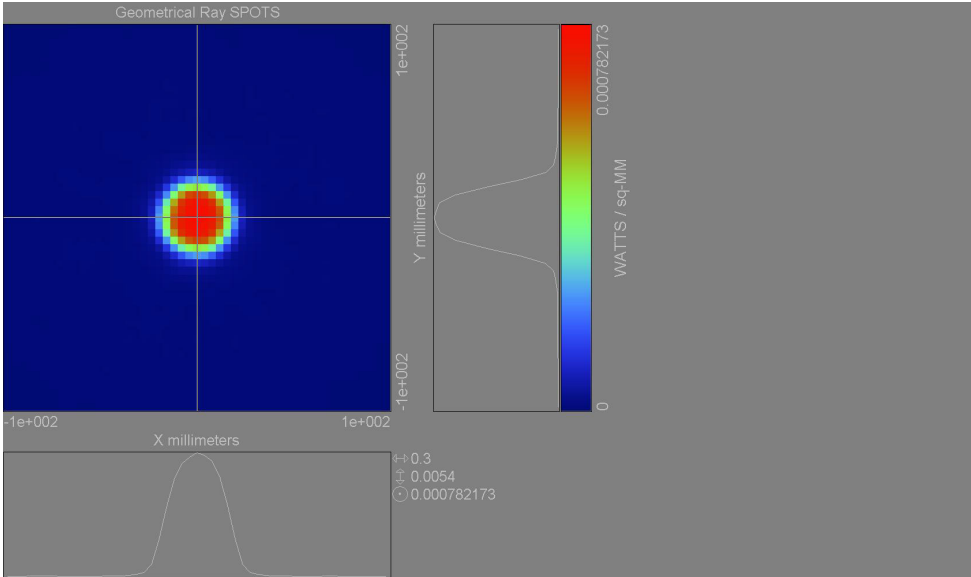
Figure 60 Irradiance distribution of Display Viewer in ASAP

12 Interrogate the 2D distribution output by moving the cursor around the pixel window to see the various flux readings, as well as change the appearance of the plots by right-clicking in various parts of the window.

We can change color palettes, gamma, cursor size, and many other useful features to get the desired plot. Once we are satisfied with its appearance, there are many options on the toolbar just above the 2D plot window.

13 Try to print, copy, make it 3D, or any of the other useful, easy to use options. See Figure 61.
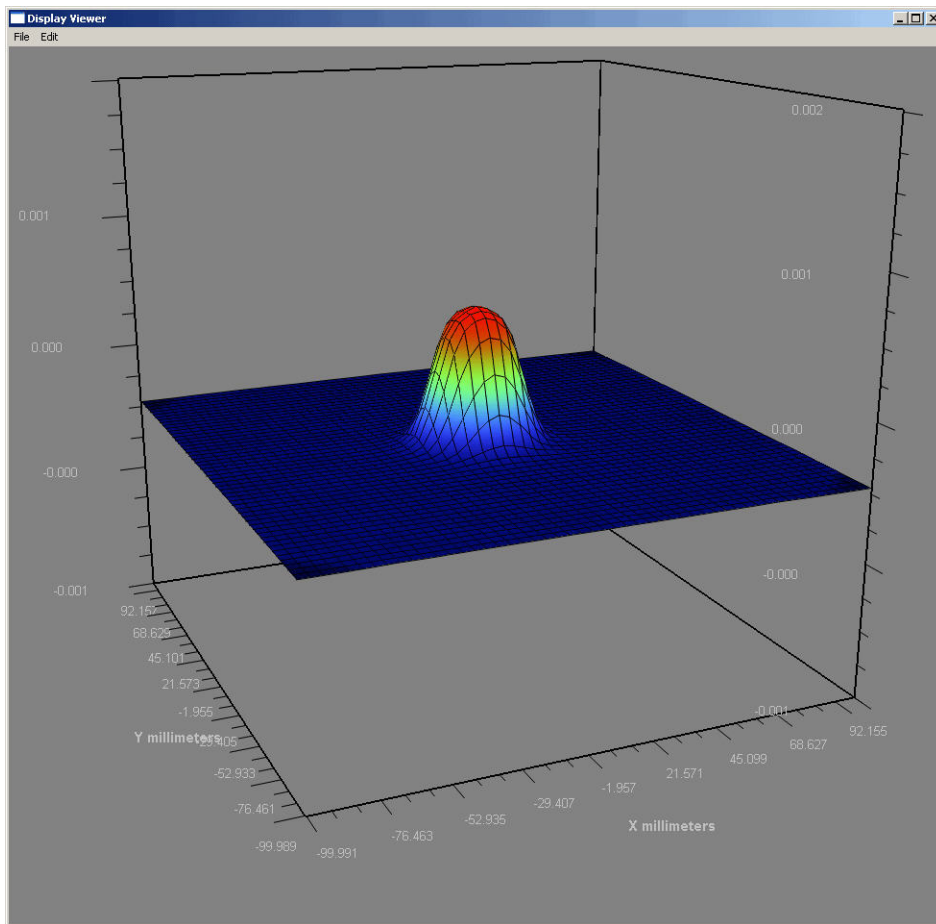
Figure 61 Display Viewer showing 3D view

Please note that ASAP has many other ways to visualize the data such as Isometric, Contour, and Graph options, which are in the Builder under **Analysis** by clicking Display> Graphics.

Solutions to our test drive are in these files:

- STEP_1_SOLUTION.ENZ
- STEP_2_SOLUTION.ENZ
- STEP_3_SOLUTION.ENZ
- STEP_4_SOLUTION.ENZ

ASAP SCRIPTING FILE

We have just test driven ASAP using the Builder. If we had done a similar analysis in script, what would this look like? First thing to note is that ASAP uses real words in its script language like **SURFACE**, **OBJECT**, **WINDOW**, **PIXELS**, and **SPOTS POSITION** to describe key words and commands. The ASAP script language is domain-specific, which relates to actual optical terms, and does not use three-letter acronyms or abbreviations. That is why it is called an optical programming language.

The first step file with an INR extension, STEP_1_SOLUTION.INR, would look like Figure 62 (note that a few "extras" have been added to the file to give you a preview about some other capabilities in ASAP).

14 Open and run the *.inr file sequentially for each step solution, using the Run icon ⚡.

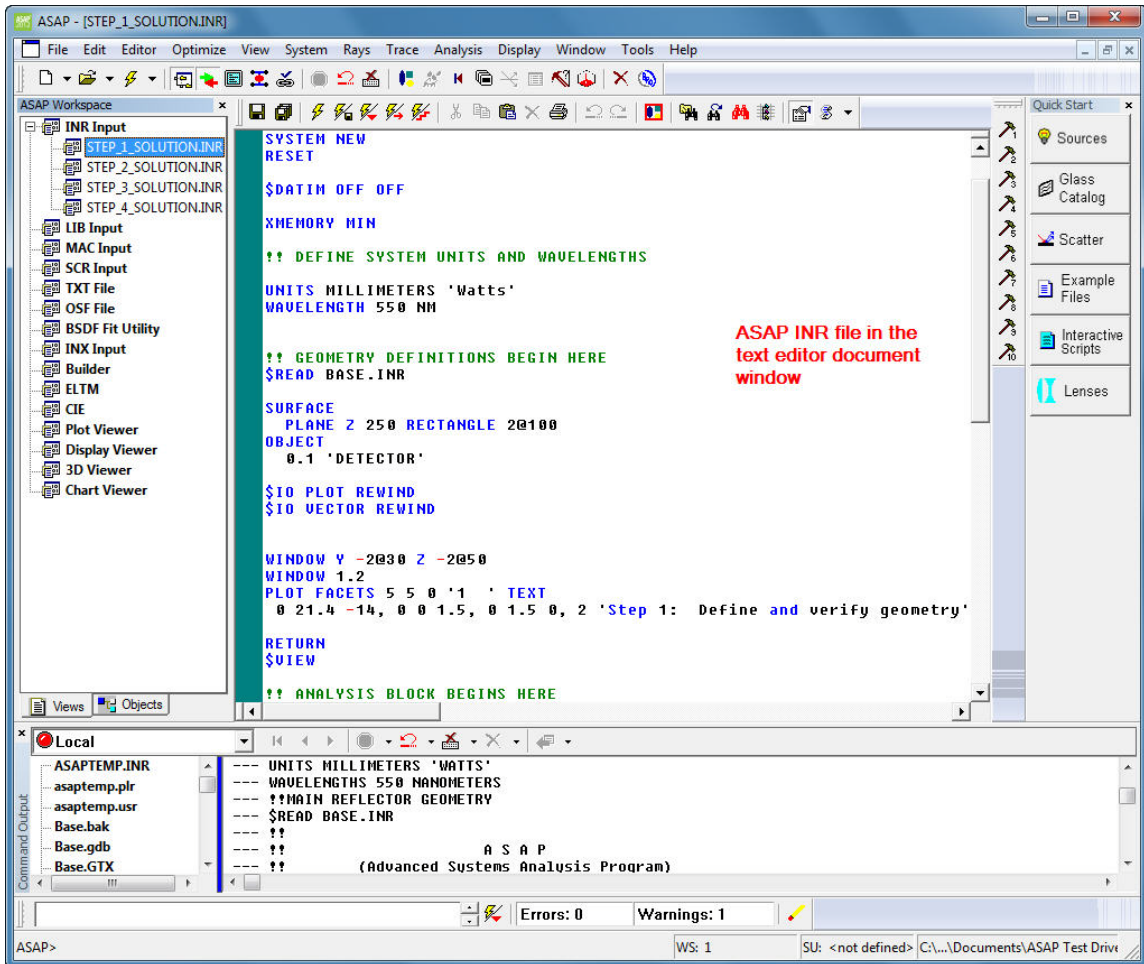As you can see in Figure 62, the scripting language is clear, concise and easy to use.

Figure 62 ASAP Workspace lists open INR files, and the document window displays STEP_1_SOLUTION.INR

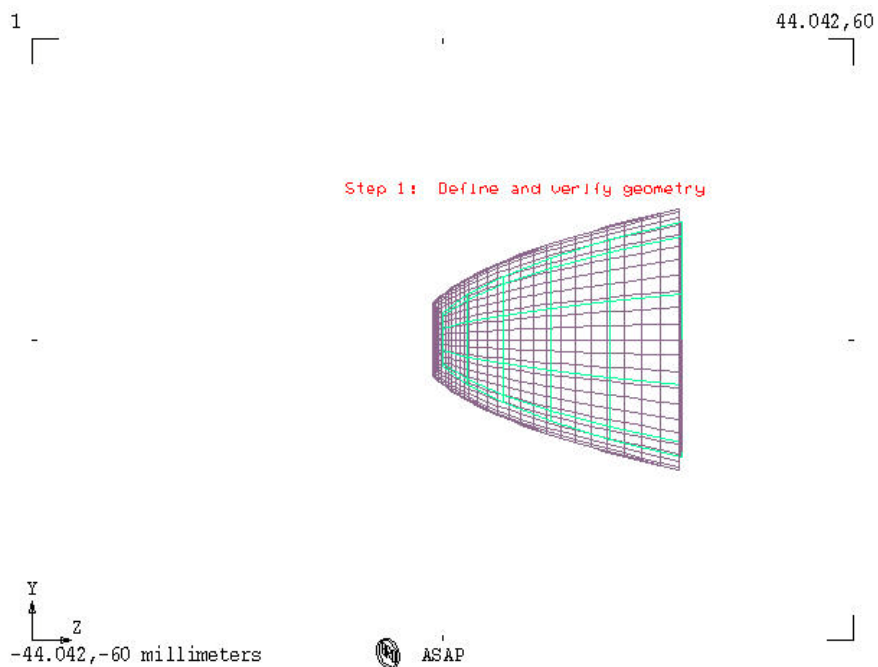Figure 63 STEP_1_SOLUTION.INR file in text editor

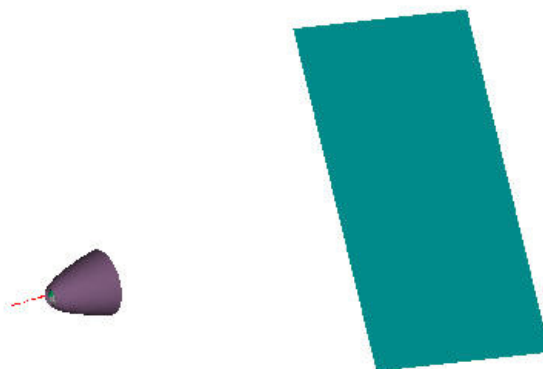Figure 64 Plot results for STEP_1_SOLUTION.INR

Figure 65 3D Viewer results for STEP_1_SOLUTION.INR

```
$IO PLOT REWIND
$IO VECTOR REWIND
CONSIDER NONE
$IO LIBRARY "C:\Program Files\ASAP2009V2R1\Projects\LightSource\LW3C.lib"
LW3C_GEOM 0 0 -2.5 +Z +Y 1 0
PLACERAYS 0 0 -2.5 +Z +Y LW3C_550NM_100K.DIS 550
$IO LIBRARY CLOSE

MISSED ARROWS 1.0
WINDOW Y Z
WINDOW 1.2
PLOT FACETS 5 5 0 '2  ' OVERLAY

PLOT RAYS 3 1000 TEXT
 0 11.8 -17.5, 0 0 .75, 0 .75 0, 2 'Step 2:  Define and verify sources'


$VIEW
CONSIDER ALL
```

Figure 66 STEP_2_SOLUTION.INR file in text editor
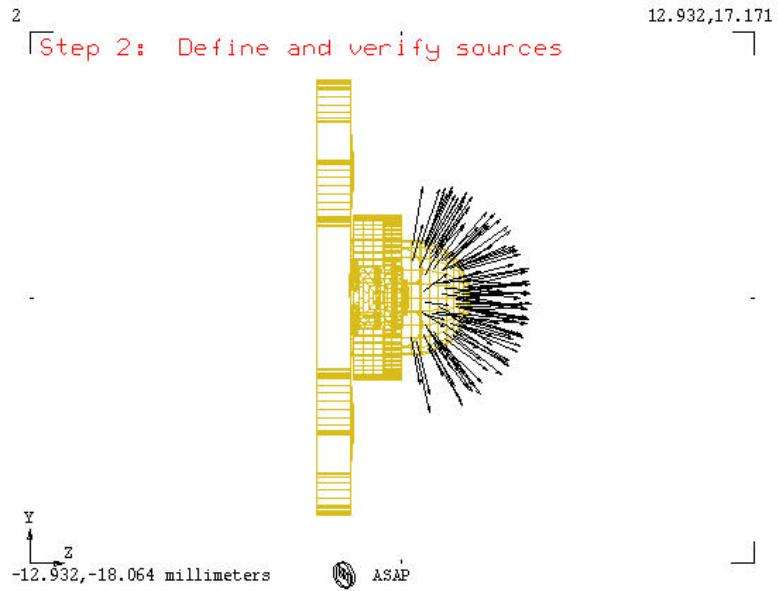

Figure 67 Plot results for STEP_2_SOLUTION.INR
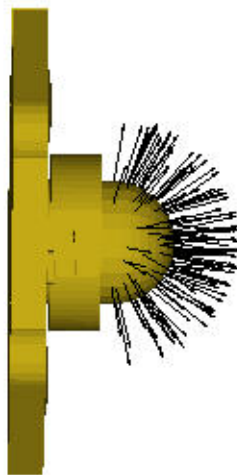
Step 2: Define and verify sources

Figure 68 3D Viewer results for STEP_2_SOLUTION.INR

```
$IO PLOT REWIND
$IO VECTOR REWIND

!! TRACE RAYS WITH OPTIONAL PLOT

WINDOW Y Z
PLOT FACETS 5 5 0   OVERLAY '3'
TRACE PLOT 10000   TEXT
 0 93.5 -10, 0 0 7, 0 7 0, 2 'Step 3:   Trace rays'
$VIEW

RETURN
```
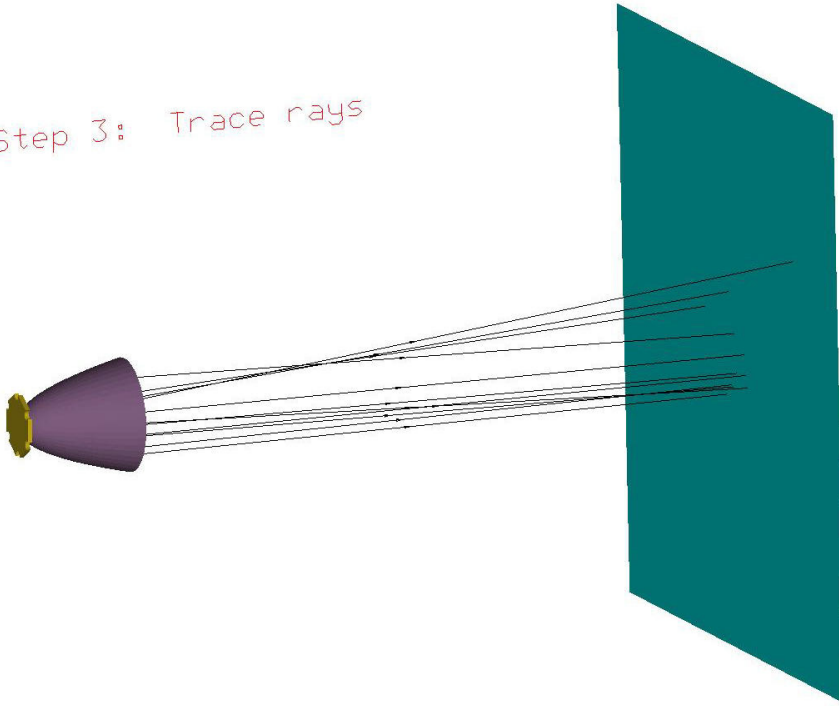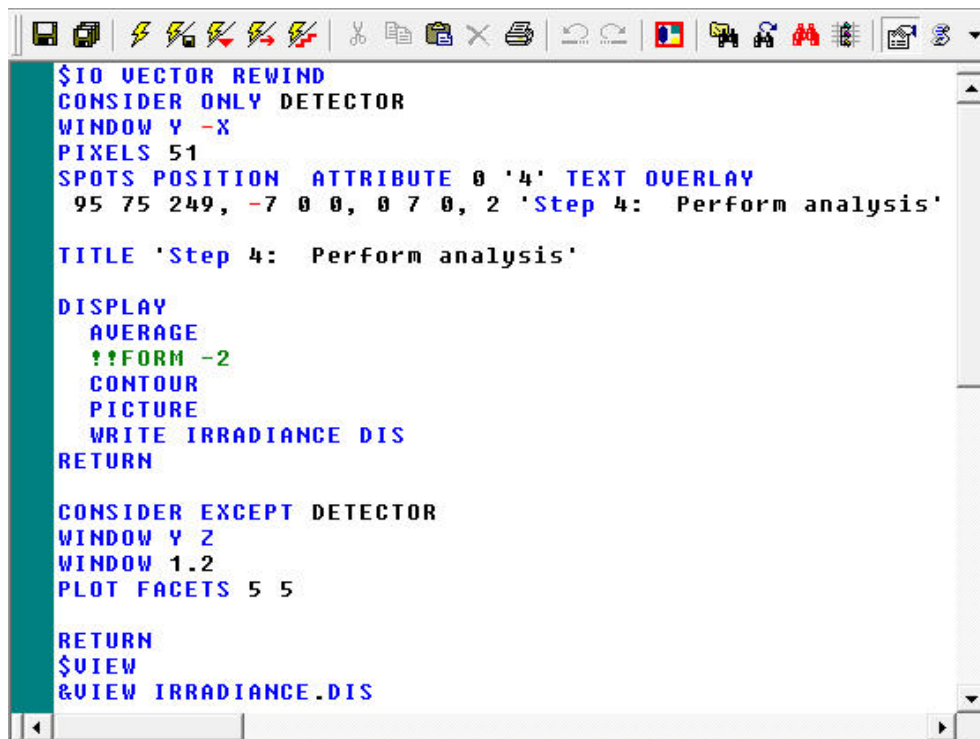
Figure 69 STEP_3_SOLUTION.INR file in text editor

Figure 70 Plot results for STEP_3_SOLUTION.INR

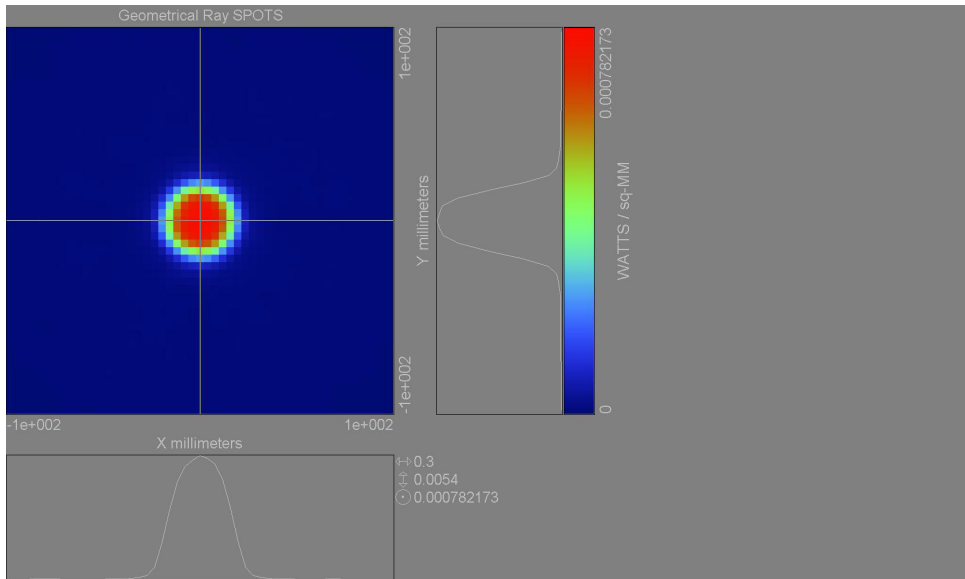Figure 71 STEP_4_SOLUTION.INR file in text editor

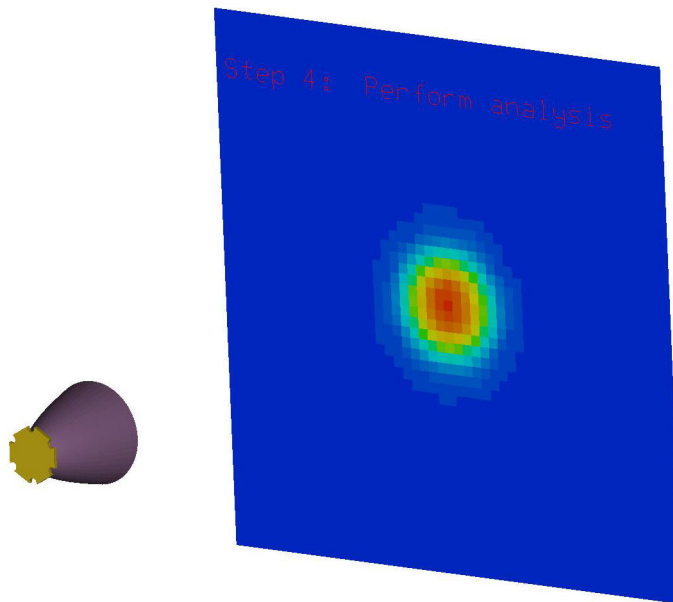Figure 72 Display Viewer output for STEP_4_SOLUTION.INR

Figure 73 3D Viewer output for STEP_4_SOLUTION.INR

CONCLUSION

Now that you have used ASAP, was it difficult? Probably not. ASAP is logically thought out and programmed to maximize workflow efficiency, without sacrificing power or flexibility. ASAP is predicated on a logical set of rules for creating and verifying your system geometry (Step 1), creating and verifying your source (Step 2), ray tracing (Step 3), and analysis (Step 4). These steps can be easily executed through the spreadsheet Builder or through its scripting and macro language.

Unfortunately, we cannot even begin to explore the full capabilities of ASAP in this test drive. However, ASAP is recognized as the most sophisticated tool available for virtual prototyping of optical systems and devices. The purpose of the test drive is to give you a "flavor" of how to use ASAP through its graphical user interface and scripting language. A short list of capabilities in ASAP includes:

*   Simulating virtually any geometry and sources found in both imaging and illumination systems including associated mechanical structures. ASAP can simulate not only the phenomena of refraction and reflection, including angularly dependent Fresnel (polarization) effects at interfaces as well as dispersion, absorption, diffraction and scattering.
*   Simulating the coherence, polarization, amplitude, and phase of virtually any source.
*   Optimization
*   Conformal Radiometry
*   CIE colorimetry

Now, as you are contemplating purchasing an optical engineering program like ASAP, we invite you to consider some other aspects of what we at BRO like to call "solving your problem" We are very confident that ASAP is the best optical engineering design and analysis tool available to address your technical software needs. However, at BRO, we are not just selling you software. We build relationships and partnerships with our customers, and actively solicit their feedback to guide our product development. In doing so, we enable our customers to solve their problems within the framework of the premier products and services we provide. "Premier" does not just mean higher price! Premier firms such as BRO are on the cutting edge of technology development and provide extensive customer support, user documentation, and education to our customers. In other words, BRO is a firm that innovates by:

Innovative product development:

*   A staff of 7 software engineers (including 2 Ph.D.s)
*   Feature-rich, reliable products
*   Quality customer-centric requirements

Excellent Technical Customer Service:

*   A staff of 5 optical engineers (including 4 Ph.D.s)
*   98% same-day inquiry resolution

Outstanding training:

*   Introductory and advanced training worldwide
*   Custom and on-site training worldwide

Extensive user documentation:

- Instructional videos
- Extensive Knowledge Base, including technical papers and user documentation
- ASAP Help
- Examples

We encourage you to compare us to our competitors, especially those who are not premier companies. Those companies excel in imitating products introduced by premier companies with the goal of satisfying their own convenience while maximizing their operational efficiency at your expense. These are companies that imitate. Sure, they sometimes offer "cheaper" software products but the software product is only part of your requirements. A software product that is easy to use but lacks the features you need to solve your problem is by definition impossible to use. BRO's goal is to make you successful in solving your problem, not just selling you software, and through that we will be successful ourselves.

We sincerely hope that you have found the ASAP test drive to be useful and ASAP itself easy to use. Please feel free to contact us with any questions you might have regarding ASAP. Also please see our web site www.Breault.com to see and hear more about our commitment to you, our customer through our:

VISION
- Taking Light Further

MISSION
Breault Research Organization, Inc. (BRO) is "Taking Light Further" by meeting our customers' most demanding requirements in the traditional optics community and markets in which optics is an enabling, enhancing, and emerging science. We do so as a premier provider of concurrent optical engineering software, training products, and turn-key hardware solutions, developed and supported by our world-class staff.

VALUES
- Our Customer, Integrity, Innovation, Responsiveness, Each Other