Tutorial: Gauge graphics
Part 1

Overview

This tutorial will illustrate two types of software used to create graphics for a gauge that will run inside MS Flight Simulator. XaraX will make the vector images. Paint Shop Pro will then finalize those into a format suitable for FS2004. XaraX shines at vector drawing, PSP shines at bitmap editing. XaraX is perhaps the less familiar of the two and will be introduced first.

To illustrate the techniques used to simulate the various parts of a gauge, a demonstration gauge will be designed. It is one example of how XaraX can handle intricate details and add realism through the use of its excellent object drawing and editing abilities. The gauge will demonstrate many features available in this software. For those interested in pursuing XaraX further, it has a thorough help section, plus numerous free tutorials at their web site.

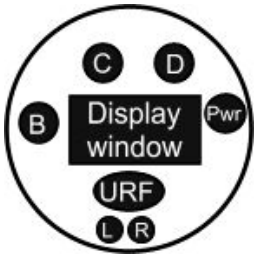
XaraX is a small (12MB), yet highly sophisticated application that produces vector drawings that can be exported in the BMP format that is required by FS. It contains many editing tools highly suited to gauge graphics. Because of its small size, it makes no attempt to have every feature that other, larger applications may contain, but are seldom applicable to gauge graphics. This limits what XaraX can do. Fortunately, what it does is simply superb. And its talents happen to be perfect for gauge graphics. Best of all, Xara's learning curve is relatively shallow. This makes it even more attractive for those who desire to begin producing graphics rather than attending software classes or searching through the local bookstore.

The demo gauge is called a Pattern Gauge. It is fully functional and designed for FS 2004. In reality it is a calculator gussied up to look like a gauge. The purpose of this gauge will be to calculate the two-digit headings for the five pattern legs around a typical airport. This includes patterns that have both left and right turns. The gauge will require the pilot to input the runway heading. From this, all other headings will be calculated and available to the pilot at the push of a button. The brains of the gauge will be XML; however, this article is not about programming. XML will be mentioned only when it may have an affect on the appearance of the gauge.

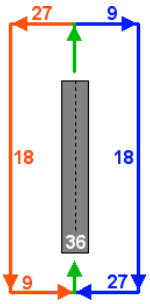
Adherence to reality is not my goal in gauges. I prefer instruments that serve some useful purpose without being restricted to what exists in the real world. However, I do want gauges to look as if they "belong" on a panel. When possible, I also want them to be dirtied-up a tad and not have the pristine appearance common to many computer-generated graphics.

The operational version of the gauge is shown at left. The eight elements that comprise the instrument have been assembled on the background of the gauge. For this illustration, the gauge displays a heading of 360-degrees. At right, each gauge control is shown in its illuminated state. These bring the gauge to life by simulating lighted buttons.

All components were not born looking this way. Trial-and-error drawings were made using various colors and shapes before the final choice was made. But XaraX greatly eases this task with the many tools available in it. All the discarded graphics will not be shown because they used the same techniques that will be discussed here. Although the demo gauge is rather basic in its graphic sophistication, it is a good place to begin. For the curious, more advanced examples produced by XaraX can be seen on the next page.



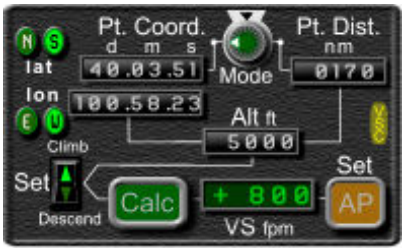
The gauge was originally envisioned something like the sketch shown at left. That was the starting point. Compared to the final version, it is obvious that some rethinking took place as the gauge evolved; however, the essentials were in place on this early concept. It contained the data window where the runway heading would be entered. The same window would display the headings of the various pattern legs as the appropriate button was pressed. The upwind leg, the runway heading, and the final leg were combined into a single button since they contained identical data.



A left-turn and right-turn button would determine how headings were calculated. As illustrated at right, the two downwind legs have the same heading regardless of whether the pattern is left-turn or right-turn. But the crosswind and base legs are calculated differently. Finally, a power switch was added to turn off the display when the instrument was not needed.

Other examples

Because the demo gauge is simple, the impression may be given that XaraX does nothing different than other software available on the market. That judgment will be left to the reader. However, to better showcase XaraX's ability to handle more complicated gauges and gauge effects, the pictures below are presented. They do not represent working FS gauges because of my limited programming skills. But they do illustrate some of the more complex graphic techniques available with XaraX.



To keep the playing field level, I must confess that some elements seen here were not solely XaraX. For example, the toggle switch on the terrain radar was made in 3D Studio. And the textured background of the radar is a bitmap technique produced with Paint Shop Pro.

Don't have 3D Studio? Not to worry, there are images by the ton of switches and other gadgetry. Keeping copyright restrictions in mind, these may be brought into XaraX for editing such as trimming and discarding of undesirable parts.

Don't have PSP? Or PhotoShop? You should have some type of quality software (not MS Paint) which can work directly at the pixel level of a bitmap image. The reason? Because some touchup work may be necessary before finalizing a gauge graphic. A pixel by pixel search for 100% black (0,0,0) within the interior of a gauge is also suggested. A vector program like XaraX cannot do this type of work.

If you have an interest in learning the techniques that produced the graphics shown so far, then this tutorial will be of interest to you. Many side trips will be taken to discuss topics not directly involved with the demo gauge. For example, the textured bitmap on the radar above is thoroughly discussed and illustrated in a future part of this tutorial.

Panels? XaraX will tackle those also…a modified R22 panel ready for new gear.

A quick note about text; XaraX treats all text as just another object to be manipulated to your heart's content. It will place text on a gauge in a realistic way that compliments the instrument. This may be text that is curved so that it parallels the edge of a round gauge, rotated text around a dial, or just text that has been given a transparency value that "tones down" the letters. This technique will allow an underlying color or texture to bleed through. Showing such a texture is the effect in this 3X view of the radar gauge.

Because of limited real estate, the design challenge of the Pattern Gauge becomes this: put all necessary components into the minimum area while eliminating confusion as to what information is being shown. No gauge should require study in order to glean essential data. A glance is all it should take because of the hectic environment that often defines a cockpit. Imparting information quickly is why so many gauges have color bands around the dial. You may not have time to read a numerical value and ponder its significance, but you can discern whether a needle is within an acceptable range such as a green band. Of course, this doesn't work for all gauges; an altimeter, for example.

The demo gauge was conceived to prevent a pilot from having to think about pattern turns should an approach be aborted and a go-around become necessary. These may happen shortly before touchdown. If so, many events are initiated, all requiring attention at the same time. If the demo gauge has been set with the runway data prior to the approach into an airport, pattern information will be available with the click of a switch, reducing the number of firing neurons.

Some final design considerations

Speaking of imparting information; some means had to be used to show what heading was being viewed inside the display window. This could be done using buttons that light up when pressed, such as those shown on page 1 of this tutorial.

Another solution, a third character could be added to the display window. This addition would show a letter identifying which leg the heading information represented. The example illustrated at right would confirm to the pilot that the heading on display was for the final leg.

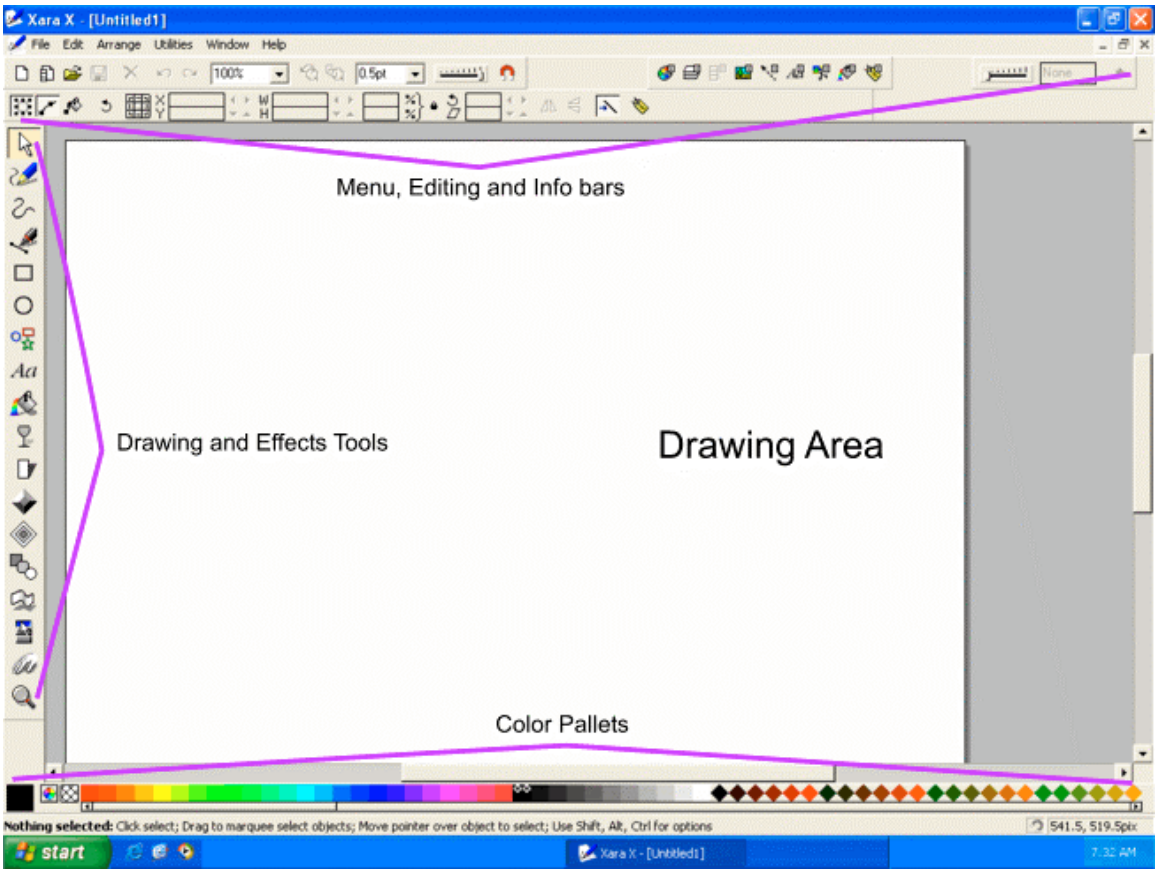
Even thought this solution had great appeal, a compromise to save space eventually gave preference to using lighted buttons in conjunction with a two-digit heading window.

Thinking how a gauge will work is an important step to take before the design phase begins. It will bring no comfort to have the graphics almost finished, then discover something is amiss, that a needed element was forgotten, or that a function is required but no provision had been made for it. Such oversight may mean a redo of the entire set of graphics. The only way to minimize a redo is to "fly" the gauge in your imagination before ever taking pencil to paper, or mouse to monitor. But thinking is only a dress rehearsal. Showtime begins when XaraX is switched on.

After graphics have begun to be generated, you then have the benefit of those visuals. They will improve decision-making. It's like tasting the soup while cooking. With each additional graphic, your feel for the instrument increases. *Gauge making*, the discipline dealing with the art and science of applying knowledge to solve a gauge design problem.

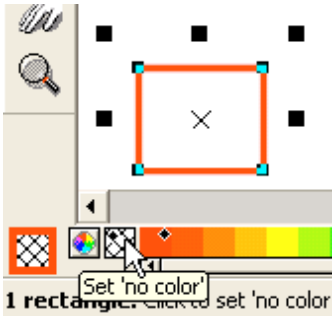
A B-52 crew was flying over the ocean. The pilot demanded to see the navigator immediately. As the navigator came up, the pilot pulled out a pistol and pointed it at him.
"If you get me lost again, I'm going to shoot you!" he yelled.
The navigator pulls his weapon and pointed it at the pilot.
"Why are you doing that?" asked the pilot.
"I'll know before you do!" said the navigator.

An introduction to the XaraX screen



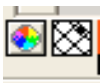
The XaraX screen shown above is a fairly standard Windows layout. It is easy enough to navigate for anyone familiar with graphic software. One difference between this application and most others is the default way it draws shapes, whether graphic or text. It places a line (border) around objects. In many respects this border is independent from the body of the graphic because it can be sized in thickness, or changed in color. Or it can be eliminated altogether.

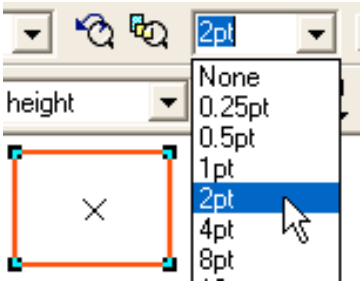
For example, the selected rectangle at right has a red border that is 2-pts in width. The body of the rectangle has no color. The white you see is the surface of the workspace. This rectangle is an object. Object colors are set using the left and right mouse buttons. The left button sets the interior color of the object, the right button sets the border color. By clicking on the color squares located at the bottom of the screen, a small black dot is placed. This dot is located in either the upper left corner (left mouse button) or upper right corner (right mouse button) of the clicked square. It indicates the color/s selected. If you look closely at this picture, you can see the "no color" square has a dot in the upper left corner. This means the interior of the rectangle has no color (transparent). The red square has a dot in the upper right corner indicating the border color of the rectangle. Using this pallet and the mouse, default pallet colors can be set as fast as you can click a button.



The small, black squares around the red rectangle are selection handles that indicate the object is "selected".

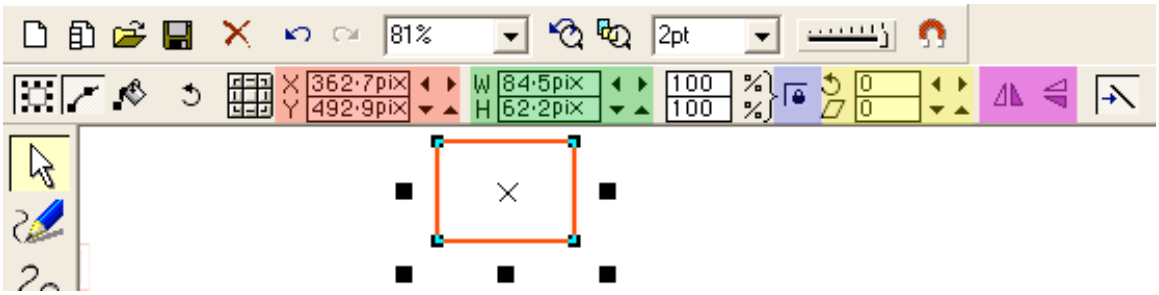
For custom colors, the little square with the colorful circle (next to the "no color" square) will activate the Color Edit box. This box allows a range of color options to be set using a variety of methods such as hexadecimal. It also has a color-picker tool similar to the one inside Paint Shop Pro.





The graphic at left shows object information available at the top of the XaraX screen. Here, we can see the border thickness drop-down box showing the 2-pt thickness of the red line around the rectangle. You are not limited to the default widths inside this box. You can type a value directly into the window and set any size you want. A value of .001 will draw a line almost invisible to the eye at 100% zoom.

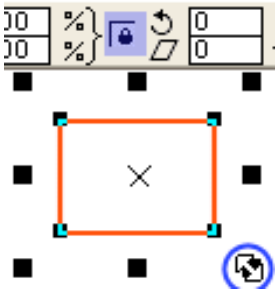
While we are looking at the top of the XaraX screen, let us discuss another extremely useful bar with lots of information and aids. This strip of digital cubicles will be an area frequently visited during gauge design. Some of the more important areas have been color-coded for illustration.



The red area provides the screen coordinates of the rectangle when measured at the left, bottom corner. This is relative to the origin (0,0) of Xara's workspace, which is also located at the LEFT, BOTTOM corner. These X and Y coordinates will change if the rectangle is moved about in the workspace. The small arrows to the right of the data windows will shift the selected object one pixel left/right, up/down.

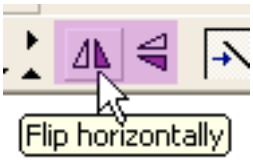
As you can see, the coordinates read to the nearest 1/10th pixel. The Y coordinate reads 492.9 pixels from the zero corner of the screen. By typing values directly into the data windows, you can locate object with great precision. For example, you could shift the above rectangle a hair's width down by typing an Y-value of 492.8. This amount of accuracy is really not needed for flight simulator but it doesn't hurt to use it when designing a gauge. In graphics, like many things, the higher you aim, the higher you hit.

The green section in the above illustration shows the pixel dimensions of the selected object. This includes the thickness of a border, if any. These sizes are also given to the nearest 1/10th pixel. This gives total control over the size of objects. Again, the arrows to the right of the data windows will adjust the dimensions in the X/Y directions.



The blue area (above and at right) contains only one item, a tiny lock icon. This spot controls how resized graphs will draw. When the lock is on, graphics retain their original proportions when typing a value into the size windows (green area, above). The same will happen when dragging a corner selection handle (inside blue circle at right) of a selected object.

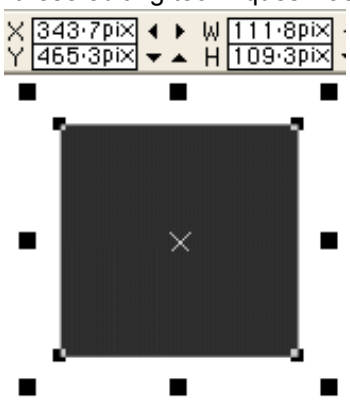
The yellow area on the information bar has one useful window that will often be needed in gauges, although not the demo gauge. The top window beside the circular arrow will rotate objects. The angle can be typed directly or the black arrows can be pressed. This window is used to produce hash marks (rotated lines) around a gauge like those shown at left. Even though the Pattern Gauge uses no hash marks, their placement will be demonstrated later in this tutorial because they are so common on gauges.



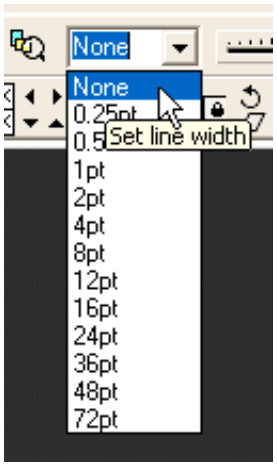
Lastly, the purple portion of the bar contains two flip buttons; horizontal flip is on the left, vertical flip is on the right. In some software a horizontal flip is called mirror. A common task in gauges is to duplicate an element such as an arrow, then flip it to get a replica that faces in the opposite direction. Duplication in XaraX is called Cloning.

A bit more about objects

As mentioned, drawn shapes in XaraX are known as objects. These can range from shapes formed using one of drawing tools, text, or imported bitmaps. Each of these can then be modified in endless ways by combining effects. Some of the more common modifications used in gauge graphics are adding or subtracting multiple shapes to form a new object, applying beveled edges of all sorts, transparency in many configurations, or different types of shadows. Each of these editing techniques was employed one or more times in the Pattern Gauge.



At left, a square has been drawn. While it is still selected (the black handles around the object indicates it is selected), the info bar at the top of the XaraX screen displays the dimensions of the object. In this case the square measures about 111 x 109 pixels.



Does the black object have a border? Since both the border and the body of the object have the same default color (black), how do you know? That data can be found at the top of XaraX.

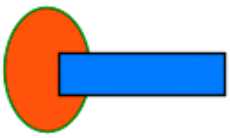
At right, the line width drop-down box is shown. It reads "None" If a border were present, this data box would show the width. One is not limited to these default values. Any value, including decimals, can be typed into the data window.

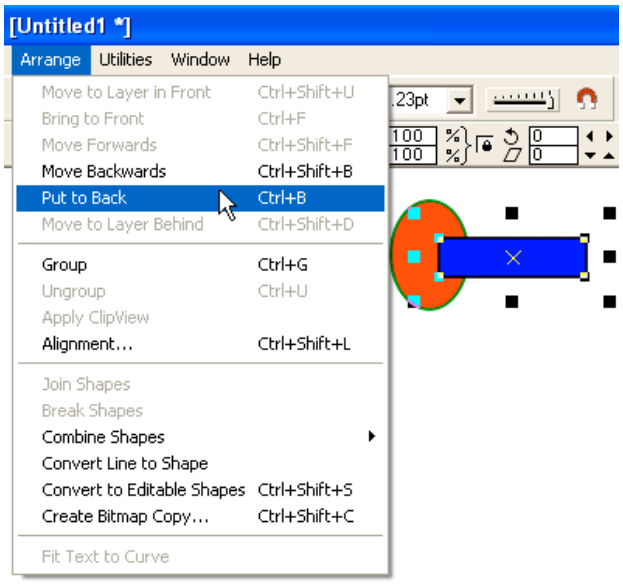
Borders straddle the edge of an object. An 8-pt border lies partially outside the object and inside the object. A really wide border around a small object would completely cover it.

Borders around objects can be edited at anytime. If, at design step 15, you find a border is needed on an object, you have the ability to create one. Even if the object has previously been rotated, given transparency, beveled, change colors, given a shadow, or all five. This is another of Xara's best qualities. You don't have to "undo" everything to get back to step one before making a change.

There are exceptions to the above statement. These include multiple objects that have been joined into a single object using one of the many methods available in XaraX. Once multiple objects have become one, the individual parts can no longer be selected. Therefore, they cannot be edited individually. However, the new object is fully editable.

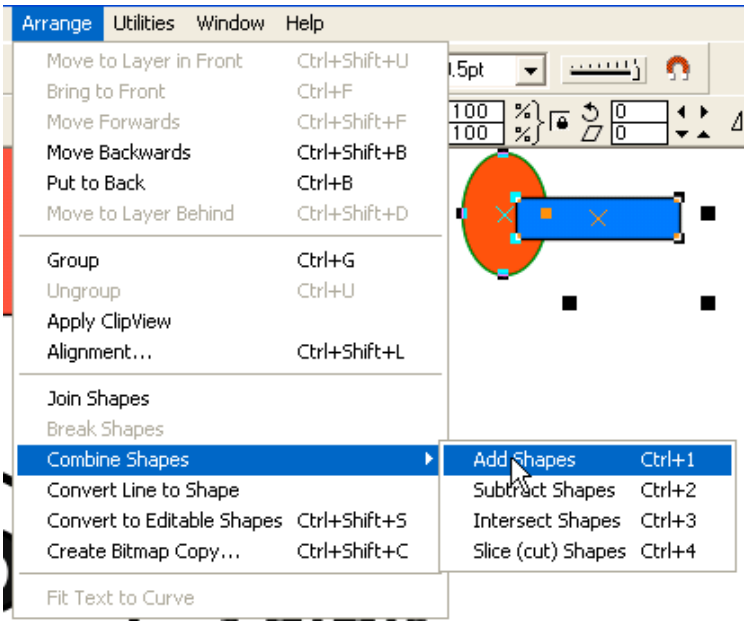
Single objects formed from multiple objects can be edited like any other object drawn in XaraX. For example, the red/green oval at right will be combined (added) with the blue/black rectangle. Note that the rectangle is atop the oval.





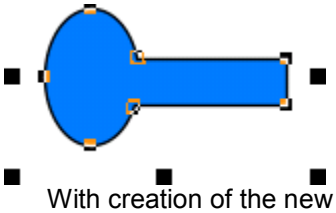
The oval was drawn first, the rectangle second. That established the drawing order, or which object gets top billing. This is similar to pages in a book. Each object is drawn on its own page. The object on page 3 will cover the object on page 2. Both will cover the object on page 1.

The drawing order is easily changed so objects can be moved forward and backward at will. The illustration at left shows how the selected blue rectangle could be moved behind the circle using the "Put to Back" tool found in the Arrange menu. In this tutorial, objects atop each other will sometimes be referred to as stacks.



To combine the two objects, they both must be selected. This notifies XaraX you intend to do something with the pair. But what? The Combine Shapes/Add Shapes tool that is found inside the Arrange menu will answer that question.

The two shapes will be joined to create a third shape. It will contain features inherited from both parent objects. The new object is shown below.



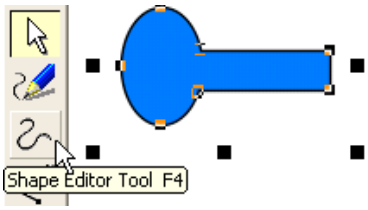
With creation of the new object, it becomes impossible to edit just the oval since it no longer exists. However, XaraX has an "Undo" and a "Redo". To edit the oval now would require undoing the previous Combine Shapes/Add Shapes step to regain the two original objects.

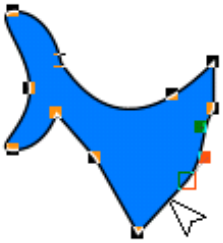
Note how the border on the new object traces the entire outline. The new shape inherited all colors of the top object, a blue interior with black border. But that is not always the case as will be demonstrated next.

At left, three additional objects made with tools found inside the Combine Shapes menu are shown. As you can see, each tool has a unique way of combining objects.

These shapes were created with the same two objects, an oval and a rectangle. Again, the rectangle occupies the top spot. But these shapes inherited the red/green colors from the oval, the lower object. The Slice is the only shape composed of two separate parts, separated here for illustration. Compare those parts to the Subtract and Intersect objects.

One other note before starting the Pattern Gauge design. The smaller squares along the border of the new object at right are editing handles. These can be pulled and pushed to form a totally new shape. That editing is done with the Shape Editor Tool shown in the illustration at right.

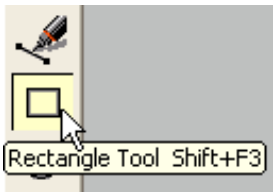




The drawing at left shows how the object was changed more dramatically using these points. They were moved about until the object no longer resembled its original form. The triangular-shaped cursor seen in this illustration will also allow the perimeter of an object to be clicked and moved. Also, if you look closely, you will see handles (red) that allow for curvature control between points. As the shape of the object is changed, the fill and border colors will follow automatically. Combining multiple objects to form a new object is a technique useful in gauge graphics. This method will be covered in more detail as the Pattern Gauge is developed.

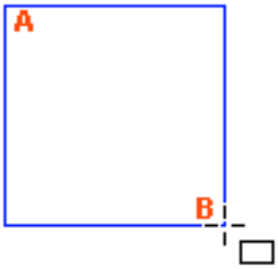
There are other tools in XaraX that may be unfamiliar to the reader. If all those were covered now, the gauge may never get designed. I will save discussing other tools, tips, and techniques until they are needed during gauge construction. To initiate that task, we will begin with the element that occupies the bottom of the gauge stack, the body of the gauge.

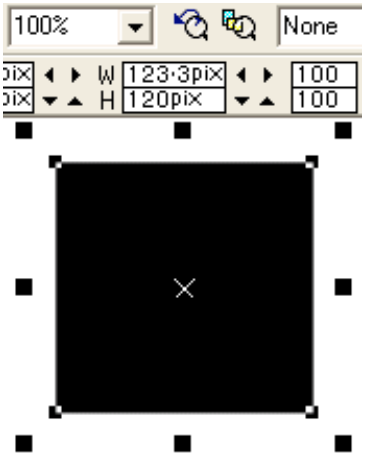
The gauge body



The body for this gauge is almost square. To construct it, the Rectangle Tool (left illustration) is selected. As mentioned previously, the rectangle will consist of two parts, the body (interior) and border. By default, both will be drawn in black unless other colors are set after choosing the drawing tool. If no border is desired, that too can be set to "None" at that time. No border was used around this rectangle.

The drawing operation is a standard Windows technique; clicking the mouse in the workspace and pulling out the rectangle until it looks to be about right. At right, this operation is underway. The blue line shows the current size of the rectangle should the drawing operation be terminated. The cursor in the lower right corner indicates that the rectangle tool is being used. The rectangle will fill with color once the mouse has been clicked to end the operation. The rectangle began at corner A and will end at corner (B).





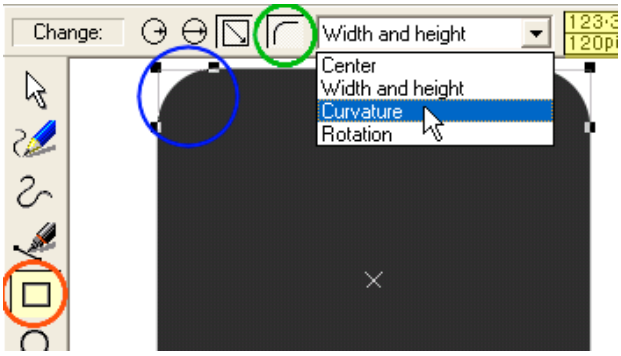
At left, the rectangle is complete. While it is still selected, the info bar at the top of the XaraX screen displays the dimensions of the object. In this case the gauge body measures approx. 123 x 120 pixels. How big should it be? It does not matter at this early stage of design. That is the beauty of vector drawing. We can change the dimensions at any time without experiencing graphic deterioration. This is true even if multiple effects have been applied. Such is not the case with bitmap graphics.

The first modification made to the body of the gauge is a color edit. We want to stay away from pure black (100-percent black, 0,0,0). For this gauge 80% black (48,48,48) appears to be a suitable color. As mentioned, this rectangle was drawn with no border, so no color change is needed for that element. No border is the same as a transparent border. The small dot in the illustration at right is located in the upper right corner of the "no color" square. That corner indicates the border color. The square to the far left in this illustration confirms the interior color (black) with a border of transparency. The diagonal grid around the black square indicates transparency. As you can see, there are multiple ways to gather and determine the same information.

To change the color of the rectangle, it is first selected to notify XaraX what object you intend to edit. Then, a left-click on the 80% black square is all that is required. At left, the dot indicator has been placed in the upper left corner of the black square. In this case, XaraX surrounds the dot by white so it can be seen. Note that a portion of the black rectangle can be seen above the color pallet. And it is now 80% black, a dark gray.

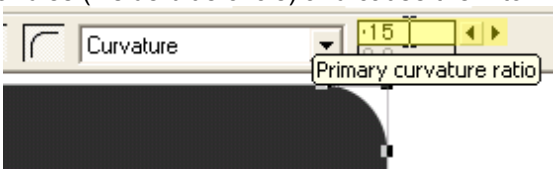
The percentage of black and the equivalent Red, Green, Blue values are given in the following table. Color patches containing the various percents of black were made in XaraX, then exported into PSP. An area 6-pixels square was then read and averaged by PSP to arrive at these numbers. The Red, Green, and Blue values are equal for each percent of black.

	Black										White
Percent	100	90	80	70	60	50	40	30	20	10	0
R G B =	0	24	48	76	101	128	152	179	204	231	255



The next editing step will fashion rounded corners for the gauge. This operation again requires selecting the object to be edited. The Selection Arrow is used for this purpose. To edit an object or change a previous edit, the tool used to create the object or edit should be active. In this case, the rectangle tool (inside red circle) must be turned on. This will make available all the editing features inside the Rectangular Tool. These appear in the information/editing bar at the top of the XaraX screen.

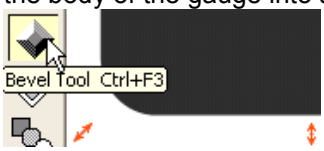
The Curved Corners button (inside green circle, above) is chosen. In the drop-down box to the right of that button, a choice of operations can be made. For this editing step, the Curvature tool is highlighted. That will activate the selection handles (inside blue circle) and cause them to appear at each corner of the gauge rectangle.



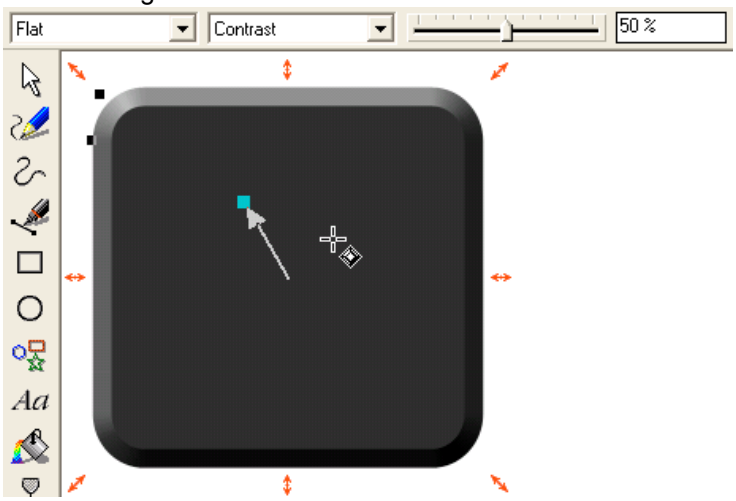
In the enlarged view at right, the yellow window shows the ratio of the curved corner. Instead of accepting the default value of 0.2, the value .15 was typed into the window. This makes the radius of the curve a tad sharper. To me, a corner with a smaller radius looked better in proportion to the overall size of the gauge. Decimal values are allowed when entering values for the curvature ratios.

Curvature ratio means how far the curve extends in an X-direction compared to how far it extends in the Y-direction. Rounded corners are not limited to forming a true arc (part of a circle).

A second way to adjust these curves is by dragging the selection handle located at one corner. As the handle is moved, the curve will change radius, even to the point of transforming the body of the gauge into an oval. As the selection handle at one corner is moved, all corners will change by equal amounts.



A third edit on the gauge body will be needed to add a bit of depth; otherwise, the rectangle looks flat and uninteresting. A rim will be added in this step. The Bevel Tool will also be introduced, one of my favorites because it has so many useful options. It is located on the Drawing and Effects toolbar shown in the illustration at left.

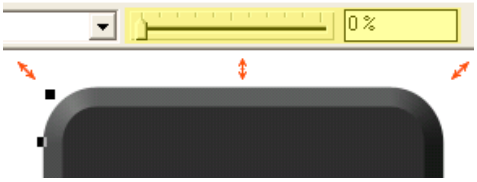


Notice the red selection handles. These appear around the selected object as soon as the Bevel Tool is chosen. The arrows indicate that both an outside and inside bevel is available. You can drag the handles in either direction. Dragging outward increases the overall size of the object. Dragging inward does not alter the overall size. It causes the bevel to slant toward the center of the object while allowing the object to keep the same footprint.

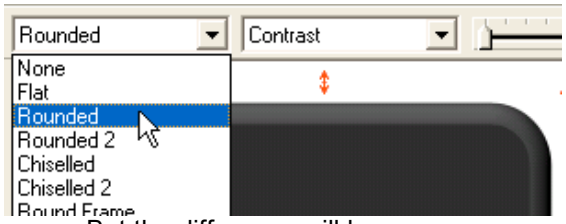
The initial bevel for use on the gauge body is shown at left. A Flat bevel appears by default. This will be changed shortly. The bevel width was set by eye.

At the top of the drawing, the default bevel parameters can be seen. It is a Flat bevel having a Contrast of 50-percent. The arrow in the center of the object points to where the light source is located. This is the default angle. It will be used for all entities needed by this gauge. I envision the major light source entering a cockpit through the left window and flowing down onto the panel.

Before transforming the flat bevel into a simulated rim, a few other edits are needed. First, the bevel will have Contrast reduced to zero. This lowers the metallic glare around the edge. To do this, the contrast slider (yellow area) is moved all the way to the left. The contrast window will then read zero percent.



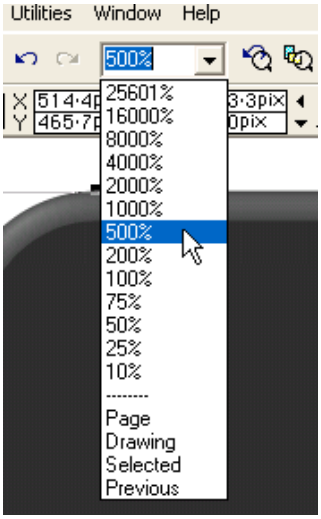
A flat bevel does not resemble the rim around a gauge. It must be changed to a bevel type that looks more appropriate. While the gauge body is still selected, all the different bevel types can be quickly tried. A few of those are shown inside the drop-down box at left. Click on each one to see the effect.



For the demo gauge, the Rounded bevel type was selected. In the gauge body at left, the rounded effect is shown. If you compare this bevel with the first, not a lot of difference can be seen. But the difference will become more pronounced as other effects are applied. This will be illustrated shortly.

I also decided to reduce the width of the beveled edge. For that, the Contrast drop-down box is opened. The Contrast tool is exchanged for the Size tool. That also changes the function of the slider. Now, it affects the bevel width, not the contrast. By moving the slider or typing a value in the window, the bevel width will change accordingly. The Pattern Gauge will get a bevel width of 3.4 pixels. Why? Because that value looks good and fits the overall size of the gauge.

But the rim is still not satisfactory. The edge around any type of bevel can be changed in color independently of the object color. This works in a similar way as the border. For the gauge, I want to brighten just the rim of the gauge but still retain the darker zero contrast. To do this, the cursor is placed directly over the beveled edge. Some zoom may be needed to position it properly. The zoom window is shown at right. A default value can be chosen from the drop-down window or a value typed.



In the illustration at left, the Selection Arrow is in position over the beveled edge. A left-click notifies XaraX that the beveled edge will receive further work. A second left-click on the color pallet at the bottom of the XaraX screen will change the color of ONLY the beveled edge. The interior of the object will remain unchanged. A value of 70-percent black was applied. This brightened the rim of the gauge increasing the sunken appearance of the interior.

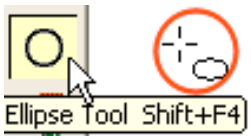
At left, the before and after color of the edge can be compared more easily. This was not a huge change but an effective one nevertheless. As work progresses, the body of the gauge will be covered in various controls. Some distinction must be made between elements including a humble rim. Color tones are one way to do this.

As promised above, the reason for changing from a Flat bevel to a Round type is illustrated at right. In this side by side comparison the Round bevel on the left simulates a rolled metal rim more effectively than the Flat bevel. Flat bevels have sharp divisions between surfaces and make the interior higher in appearance. A Round bevel lowers the interior of the gauge.

The body of the gauge is now complete. No effect was added to simulate a rim shadow falling to the interior of the gauge. It was considered but dismissed. This gauge is rather small. If too many shadows occupy the gauge, it might as well be colored with a darker gray in the first place. I would prefer some lighter grays to show between buttons. This was a compromise only for the sake of appearance.

Power Button

If the Rectangle Tool was used to draw the rectangular body of the gauge, what tool would be used to draw the circular Power Button shown at right? If you answered the "Circle" tool, then you are a logical thinker with an incorrect answer. It is constructed using the Ellipse Tool, shown at left. The ellipse cursor is illustrated inside the red circle. Ellipses are just circles that are low on air. To pump them up, the CTRL key is pressed while drawing.



The first step in making the Power Button is to draw a circle at a reasonable size. Remember that vector objects can be resized at will with no loss in graphic data. However, if you draw elements unreasonably far from their final size, you may find effects, such as a shadow, losing their effectiveness if the image is shrunk. To save wasted effort, it is best to produce all elements at a size that is in the ballpark to their final size. Then you will have a better idea how the soup will taste.

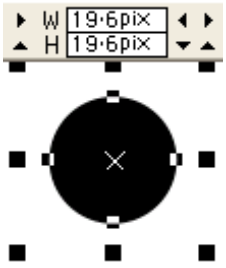
A workman was standing near the hanger stirring cow manure inside a tub.

A mechanic walked by and asked the workman what he was doing with all the manure. The worker replied that he was making an airplane mechanic. Angry, the mechanic stormed away.

A truck driver stopped by and was also curious what the workman was doing with a pile of manure. The workman informed him he was making a truck driver. Thinking that funny, the driver pulled away chuckling.

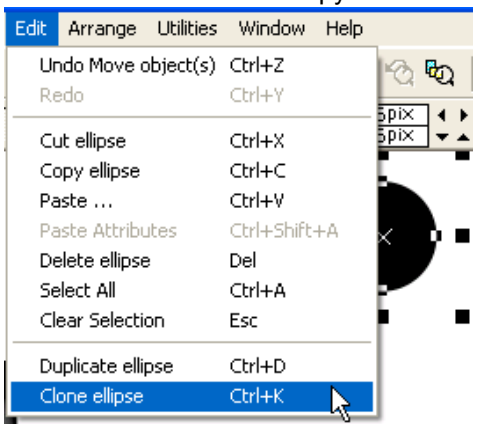
An airline pilot standing nearby had overheard the worker's previous remarks. He strolled over and stated, "I guess you are making an airline pilot."

"Oh, no, sir," said the workman, " I don't have nearly enough manure for that."



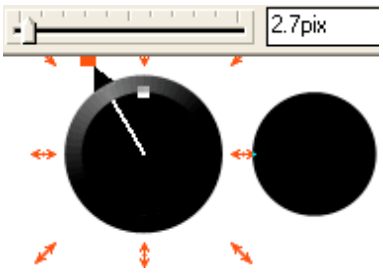
The circle at left has been drawn with no border. The dimensions are shown in the size windows. This circle uses 100-percent black. Why? Won't that cause trouble in Flight Simulator? That answer will become clear as this button is developed.

I know generally how this button will evolve because, after a while, they begin to follow design patterns. I know that an exact copy of this circle will be required for later use, therefore, a clone is made while the circle is still selected. The tool needed for this is found in the Edit menu at right. For now, the cloned circle will be kept nearby on an unused area of the workspace awaiting its assignment.

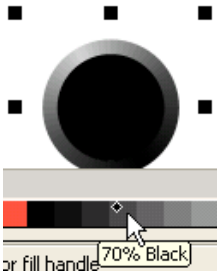


Tip: To move selected objects one pixel at a time, use the arrow keys on your keyboard. To move the objects faster, hold down the CTRL key in conjunction with the arrows.

With the original black circle selected, a beveled edge is applied. This time it will be an outside bevel. Consequently, the overall size of the button will increase from 19.6 to 25 pixels. The default Flat bevel type will suit our purposes for this button.



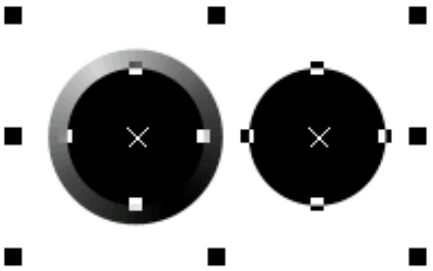
This illustration shows both the beveled circle and its cloned cousin. The Size slider was moved until the bevel width looked appropriate. That happened when the 2.7-pixel value was reached. Note that the light source is again from the upper left, the default Light Angle. The Contrast is left at the 50-percent default value. The Light Elevation also remained unchanged.



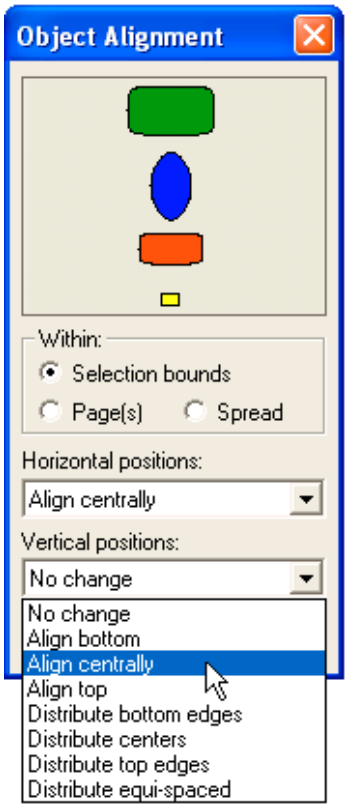
A decision was made to lighten the beveled edge a bit. The edge was selected and a color of 70% black was applied. That effect is shown at right. Why not keep the bevel color the same and increase the Contrast instead? The graphic at left shows a comparison using this idea.

The left bevel used only a Contrast change. The right bevel used only a Color change. The left bevel has 100% Contrast using the original 100% black color on the bevel. The right bevel retains the default 50% Contrast, but the bevel color was changed to 70% black. As you can see, High Contrast values cause highlights. If that is the desired effect, great. But for the Power Button, the glare was distracting.

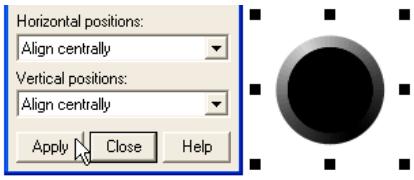
Time has come to put the clone in action. By dragging the Selection Arrow around both objects, they each become selected. This is necessary because the clone will be centered directly over the beveled object. If you remember, the clone was drawn AFTER the beveled object. That gives it the top spot in the drawing order.



To align the selected pair, the Object Alignment tool is used. It is found in the Arrange/Alignment menu. The box is a persistent type and can be left open in any unused corner of the workspace. It is a very helpful and a feature used often.



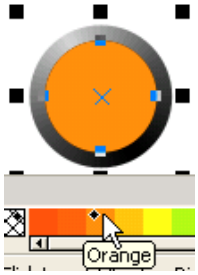
To position the clone centrally over the beveled circle, set the Horizontal positions drop-down box and Vertical positions drop-down box to "Align Centrally". Then, press "Apply". The two selected objects will jump together as shown at right. The clone is sitting atop the beveled circle. This is important to know because we now want to select only the clone. The pair must first be de-selected by clicking any unused spot on the workspace. If this were not done, both objects would be the recipients of the next edit.



To select only the clone, simply click on it. Because it occupies the top spot in the drawing order, it will be the object XaraX selects.

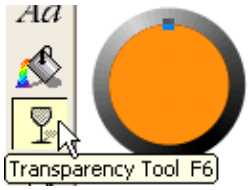
Tip: To select the beveled circle below the clone, press the ALT key. The Selection cursor will turn downward. This indicates the object to be selected will be below the top object.

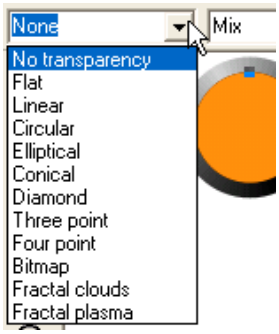
If you recall, the body of the beveled circle uses 100% black. The clone inherited that color. The reason for assigning pure black will now become apparent.



At left, the clone is shown selected. An orange color has been applied. The orange will simulate both the on and off condition of this button. When off, the button will be dimmed. When on, the button will be illuminated. First, the OFF state will be constructed.

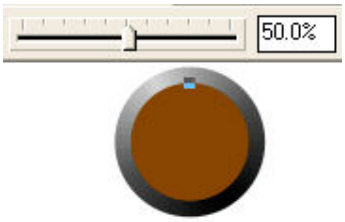
The bright orange must be toned down in color value. To do that a new tool will be called into play, the Transparency Tool. It is located on the toolbar at the left side of the XaraX screen. Clicking the wineglass icon at right, the property boxes for the Transparency Tool will be activated. These are located at the top of the XaraX screen.





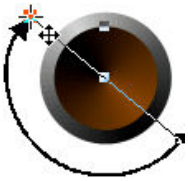
The types of transparency available are shown in the illustration at left. By default, a Flat transparency will be applied if no other choice has been made. For this demonstration, let's apply the Flat transparency initially so that other features regarding this outstanding tool can be explained. By choosing Flat from the list, transparency is applied using default values.

If you look to the far right of the information bar at the top of the XaraX screen, a slider with the window shown at right can be found. The 50% value represents the default transparency that is automatically applied. You can see in this illustration, this amount of transparency transformed the bright orange into a shade of brown.

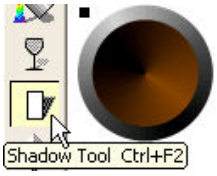


This is the reason for having pure black lying underneath the orange clone. The brown color is a mixture of 50% orange, 50% black. This effect illustrates the phrase "toning down the orange". Even though this shade of color will work for a button that is "off", the flat surface of the control requires more visual interest.

To provide some depth to the button, the Flat transparency is exchanged for Conical. At left, you can see how the shading changed when the new transparency was applied. You will also see the circular arrow across the top half of the button. This represents the direction of the light source. As applied, the light source causes the button to appear as a cone. The point of the cone is located at the center of the button. That could make a pilot's finger sore. Let's rotate the arrow and change the direction of the light source. This action will indent the button by removing the cone effect. The result of that action is shown at right.



Also note that the arrow can be increased in size or the center can be repositioned, even off the object. These all have an effect on how XaraX will display the final conical transparency. The Power Button looks acceptable at this point. Since it is made from two objects, these can be grouped. This step is often a good idea once an element has been completed. The Group/Ungroup Tool is found in the Arrange menu.



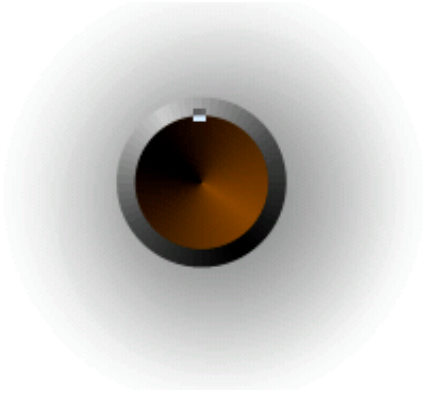
Let's group the clone and beveled circle together, then apply a shadow. That will be the last edit for this control. I feel a shadow is needed because the button will be positioned at the right side of the gauge. That will give it room to display the shadow near the rim of the gauge. It may also cast the shadow across a portion of the heading window. If so, that will add to the overall realism.

The Shadow Tool is located on the toolbar at the left side of the XaraX screen. When the button is pressed, the default shadow type is No shadow. Shadow information and parameters are displayed at the top of the XaraX screen.

The four types of shadows available in XaraX can be seen in the red area of the next graphic. The most useful type for gauges is the Wall shadow. That is the second button on the bar. The Wall shadow effect will be applied to the Power Button.

When a Wall type is applied, the default shadow will be cast down and to the right of the object (or grouped objects). That effect can be seen above. This direction follows with our lighting scheme. The default properties of a shadow can be seen at the top of the XaraX screen after applying this effect.

Above, the yellow area indicates shadow blur. This is the number of pixels XaraX uses for the transition from total shadow to no shadow. The default is 6-pixels. A zero value produces a solid shadow with no blur as seen at left. The maximum value for blur is 40-pixels. That haze effect is shown at right.

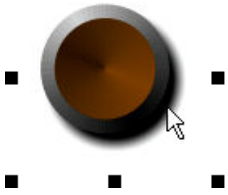


The green area on the above illustration indicates the degree of shadow transparency. No transparency is the default value. With this slider, the shadow can be adjusted to increase or decrease the visibility of those elements that lie under it.

Because the power button is located on a gauge body that is dark gray, any transparency in a shadow would render it less effective. Therefore, the transparency value will be left at zero.

The blue data windows contain the number of pixels that the shadow is cast in the X-direction and the Y-direction. Positive or negative numbers can be typed into these boxes to control the location of shadows. Or, the black arrows beside the data windows can be clicked. The location of a shadow is a judgment call. If you wish to make an element appear taller, move the shadow further from the element. For the Power Button, a value of 2-pixels to the right and minus 2-pixels down looked correct.

One last feature concerning a shadow that is not readily apparent. They are made using the default color of black. But this can be changed independently from their object. For example, our Power Button could be given a magenta shadow. First, click on the shadow (not the object) to notify XaraX that work will proceed on the shadow. A zoom may be needed to get the Selection Arrow in the correct spot. The selection process is shown at right.



Once selected, the shadow is given a color by clicking the mouse on any color square in the color pallet or by creating a custom color. While magenta may be unreasonable, other colors could prove useful in certain situations. For example, if the gauge body were a tan color, then a dark brown shadow would look more realistic than black. Shadows for colored objects should contain color themselves.

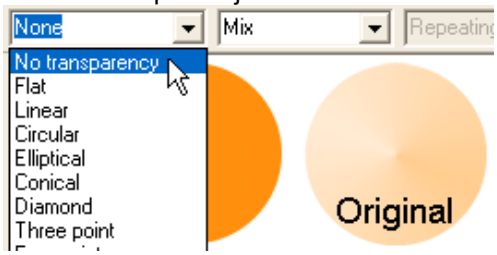
Associated with the Power Button is the text identifier, PWR. Because the Text Tool will be covered in a later section of this tutorial, the label will not be discussed at this time.

This completes the OFF state of the Power Button. But, it also needs an ON state, or an illuminated version. Two variations are needed for every control that simulates two conditions of operation. For the demo gauge, the only element with only one version will be the Heading window.

Illuminating the PWR button

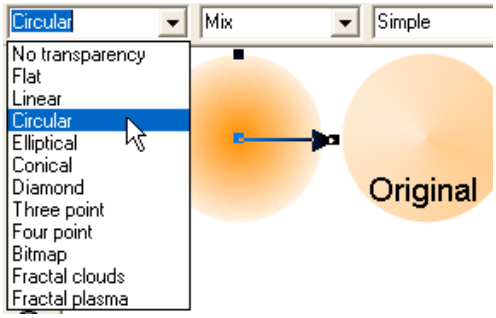
The OFF state of this control used only two elements; a beveled object and a transparent clone. To simulate the ON state shown at right, three elements will be needed. Those are shown at left.

The beveled object that forms the base of the button is identical to the beveled object that was previously constructed. A clone was made and will be used here. The center image is a clone of the transparent circle. The star-shaped object is new and will have to be created.



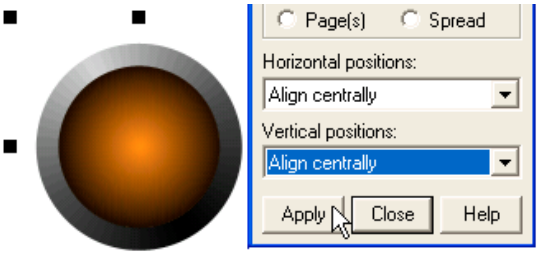
At left, the original circle and the newly cloned circle are shown. The new circle is selected, the Transparency Tool is activated, and the transparency is set to No transparency. As you can see, this removes the Conical transparency and resets the circle back to its original solid color. This step overcomes a quirk of XaraX. When exchanging one type of transparency for a new type, remove the initial transparency first. Then, the new transparency can be applied without encountering unexpected results. Such results may include the direction of the light source or an unforeseen effect of the transparency itself.

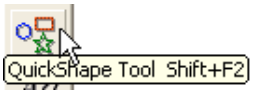
After the transparency is removed, a new type can be applied. This will be a Circular transparency. The result of that application is shown here. The arrow shows the default placement of the circular transparency. The arrow can be lengthened or shortened which will cause the effect to shrink or expand. The arrow can also be rotated to change the direction of the light source.

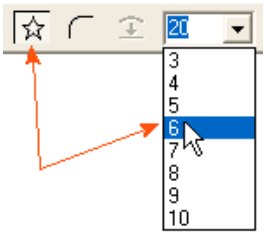


The reason for the new transparency was to simulate the circular glow from a bulb. The strongest light would be in the center and become attenuated as it nears the edge of the button.

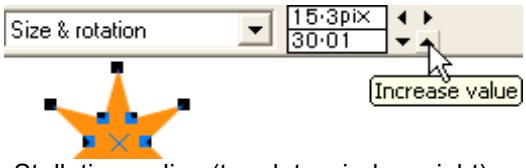
If the base of the button and the transparent circle are aligned centrally, the image at right will result. Although this looks acceptable, one more object will put icing on the cake. The new object will also allow another tool to be introduced, the QuickShape Tool, seen at left. It can be found on the toolbar at the left of the XaraX screen. When this tool is activated, the QuickShape parameter bar also activates at the top of XaraX.



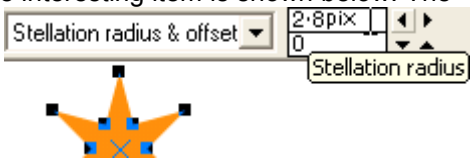




For this effect, a star-shape containing 6-points will be used. Those two property choices are seen at left. The mouse will create the star as it is pulled across the workspace. That operation is underway in the illustration at right. The drawing cursor confirms the star shape has been selected for this task. The star is an object and, by default, a border is placed around it. XaraX colors both the object and the border black. This can be prevented by informing XaraX that no border is desired and by choosing a color other than black. This must be done immediately after selecting the QuickShape Tool and before drawing has begun. However, those parameters can also be changed after creating the object.

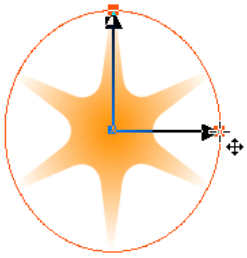


The illustration at left shows additional editing taking place on the star. The size and rotation were changed to the values shown. Inside the Size and rotation drop-down box are other goodies. One interesting item is shown below. The Stellation radius (top data window, right) and the Stellation offset (bottom window) affect the star even further. I will not attempt to describe these changes but will give visual examples instead. Below, the left image shows a change in radius, the center image shows a shift in offset. The image on the right was the product of the Curvature tool. This tool modified all sharp corners. The possibilities for editing are endless. Some are outstanding, some helpful, some are more suited to web pages rather than gauges.



However, the star is still not suitable for the Power Switch. It must be given transparency to eliminate the hard spikes around its perimeter.

The transparent version of the star is shown at right. It has been given an Elliptical transparency. The two arrows define the major and minor axis of the ellipse. As with most effects in XaraX, the elliptical values can be shifted at will until the desired effect appears. The eye is the best tool one has for gauge design. The Elliptical transparency will cause the points of the star to blend in with the black background of the Power Button while retaining the orange intensity near the center. The idea is to present a feeling more than presenting a star-shape per se.



The three elements of the button are aligned centrally, then grouped together forming one element. A Wall shadow is then applied. This shadow is identical to the shadow of the OFF Power Button. The graphic at left shows the Power Button in its final, illuminated state.

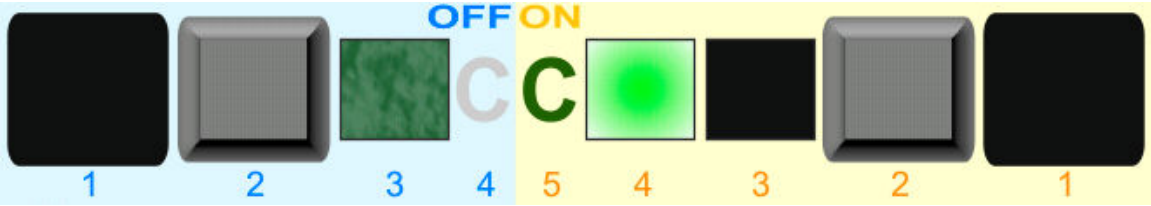
The blond airline captain decided to increase security on his flights. This required all personal to present IDs as they boarded. As the new, blond stewardess came aboard, he asked to see her ID.
She rambled in her purse, then asked, "What does it look like?"
"It is square-shaped with your picture on the front," he answered.
She rambled again and came across a small mirror. "I think I found it," she said.
She handed the mirror to the captain.
"Oh", he said, "I didn't know you were an airline captain."

Heading Buttons

The next element to be designed and developed is a Heading Button. All four of these buttons use identical techniques so only the Crosswind button will be demonstrated. Because they are illuminated, they will be developed for two states of operation, ON and OFF.

Both states of operation are shown at left. These buttons are identical in every way except for the illumination effect. The OFF button will be constructed first. The only new XaraX tool used on the button is the Text Tool. But new ways of editing previously covered techniques will be described.

If you study this illustration, you may begin to recognize certain effects such as rounded corners. These corners do not have the circular curves that were made for the body of the gauge, therefore, a new technique will be introduced. You may also recognize beveled edges and transparency applications.



The seven elements that comprise the two states of this one button are shown above. The four on the left make up the OFF state, the other five make up the ON state. The numbers below each image give the drawing order for that element. Number 1 occupies the lowest position in the stack. How these elements are produced with XaraX tools and why they are needed will be the focus of this section.

As you might guess this button begins life in the same manner as the body of the gauge, as a black rectangle without a border. But, some of the elements on the button will use a border. For example, the green squares contain a thin black border. The reason? Borders can often serve to better separate the various elements that form the identity.

Understanding borders is important. A brief explanation concerning their properties and how they interact with the object to which they are attached is now appropriate. This side trip will explain the various features of this object. Then, we will return to designing the Heading Button.

Borders

To prevent any potential confusion between borders and lines, both entities will be discussed here. A border resembles a line, except that it encloses an object and belongs to that object. Wherever the object goes, so goes the border.

To refresh your memory, by default, a border of 0.5-pts is placed around objects by the various drawing tools. Two exceptions to this are shown at left, The Freehand and Brush Tool and the Pen Tool. These tools draw only lines.

Lines are not regular objects. You can think of them as segments of a border. They cannot contain an object color and a border color, as do regular objects. Only a border color can be applied to them by right-clicking the mouse.

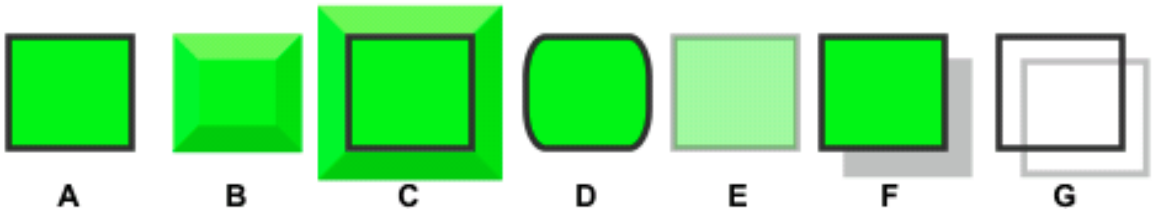
For example, the red line at right was drawn using the Pen Tool. The default 0.5-pt line width was changed to the much larger 8-pts for this illustration. The color squares below the line show that the object color is yellow, the border color is red. But, no yellow appears anywhere on the line indicating that lines have no interior, as do normal objects.

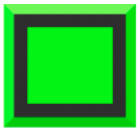
At left, the same line was given a "no color" border and an object color of red. The line, pointed to by the blue arrow, has disappeared. If the line had an interior, it would show in this illustration.

Lines and border share common traits. They both can have color, which is applied by a right-click of the mouse. Both can also have varying widths. These are properties that both must have in order for them to be useful. If they have no color or width, then they cannot be seen. If they cannot be seen, they have nothing to offer.

But lines and borders also share other editable parameters. Those will be demonstrated next. Let's look first at the parameters of a border, then compare it to those of a line.

In the following diagram, objects **A** through **G** have been edited using various tools available in XaraX. They illustrate how the border reacts as each edit is applied. The original object is **A**. It is a green rectangle having a 1-pt black border.

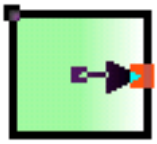




Drawing **B** shows the object after receiving an interior beveled edge. In that edit, the border disappears as it is covered by the bevel. However, XaraX is much more sophisticated than this simple example. Using a much wider border (4-pts) the interior bevel will produce an entirely different effect shown at left. By altering the bevel or border sizes, many such effects can be produced.

Drawing **C** shows the object with an exterior beveled edge. The border is visible because the bevel moved outward.

Drawing **D** illustrates the rectangle with curved corners. The border automatically stays parallel to the edge of the object during this edit.



Drawing **E** shows a Flat transparency applied to the rectangle. As you can see, the transparency affected both the object and the border equally. This does not hold true with all types of transparencies, for example, the Linear transparency seen at right. In that illustration only the object accepted the transparency.

Drawing **F** shows an example of the Wall shadow. This shadow is composed of both the object and the border. If the border were removed, the shadow would decrease in size accordingly. Shadows will be discussed in more detail later in this tutorial.

Drawing **G** is also a Wall shadow. This time the object was colored transparent so only the border casts a shadow.

Part 2 of this tutorial will continue with a demonstration comparing the properties of lines.