Chapter 11
Working with Icons and Cursors

Objective

This lesson introduces you to icons and cursors and their uses in CA-Visual Objects. By the end of this lesson, you will know how to:

- Create icons and cursors
- Attach icons to your application windows
- Use icons as controls
- Customize your Application Browser with an icon
- Use cursors

Overview

Icons

As a user of Windows, you have seen icons used in the following situations:

- Icons used to represent programs and groups within your desktop manager. Typically, these icons are defined in the program's .EXE file, a .DLL file, or an .ICO file. The icons are identified to the desktop manager via the Program Item Properties.
- Icons within a program's Help About dialog box. In this case, the icon is displayed on a window as a Fixed Icon control.
- Icons used to represent a minimized window or application on the desktop.

Cursors

Two commonly used mouse cursors (often referred to as mouse pointers) are the arrow and the hourglass. This lesson shows you how to create and use cursors in your applications.

Note: Mouse cursors are implemented by using the Pointer class. Do not confuse mouse cursors with text cursors, which are implemented via the Cursor class.

Exercise

The CA-Visual Objects Icon Editor allows you to create and manage your own icons and cursors. Within the Icon Editor workspace you can create, modify, save, and retrieve icons and cursors.

Icons and cursors are defined in an .ICO or .CUR file. Your application makes use of an icon or cursor via a related class, its Init() method, and a resource entity that contains the path and file name. Double-clicking on the Icon entity opens the Icon Editor with the icon or cursor displayed.

In the South Seas Adventures application, all of the icons are stored in the same module. To open this module, perform the following steps:

1. In the Application Browser, double-click on the South Seas Adventures application button.
2. Double-click on the App:Resources module button.

   The Entity Browser for this module is opened.

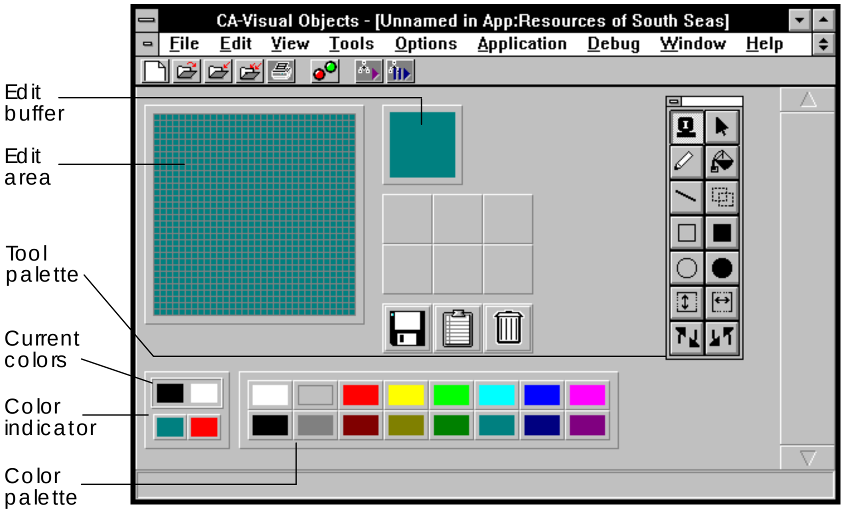
You will now explore the various ways icons and cursors are used in a CA-Visual Objects application.

Creating an Icon

Let’s start by creating a new icon to associate with the minimized Edit Customer window in the South Seas Adventures application:

1. Select the Icon Editor command from the Tools menu.

   The Icon Editor window is displayed. The various parts of the Icon Editor workspace, including the edit area, the color palette, the color indicators, and the tool palette, are shown below:

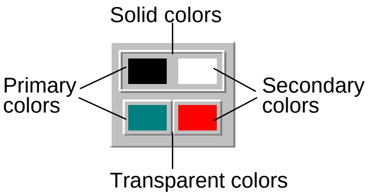


2. Maximize the Icon Editor.

The Icon Editor features many of the tools you may have seen in popular graphic editors and paint programs. These tools allow you to:

- Draw freehand
- Draw lines
- Draw rectangles (solid and outlined)
- Draw ellipses (solid and outlined)
- Fill areas
- Flip areas
- Rotate areas

When the Icon Editor is first displayed, solid colors are selected by default, as indicated by the highlighted box in the color indicator:



The two left, or primary, colors in the indicator represent the colors used when you draw pressing the left mouse button. The two right, or secondary, colors in the indicator represent the colors used when you draw pressing the right mouse button.

The upper color indicators show *solid* colors. Solid colors appear on the icon exactly as they are displayed in your Icon Editor.

Note: When being dragged, solid colors are only displayed in black and white.

The lower color indicators show *transparent* colors. These primary and secondary colors work differently from the solid colors:

– A primary transparent color will not appear on your icon when displayed, allowing what is behind the icon to show through. When selected, it fills the icon background completely—allowing you to view your icon in different color backgrounds.

– Areas that you draw using a secondary transparent color will be displayed as the complimentary color to what is behind it.

You can select Solid or Transparent colors from the Options menu.

Now, it's time to discover your hidden talents by drawing a simple icon, using solid colors. Since the icon will be used to represent customers for South Seas Adventures, let's draw a happy face:

1. Select the color yellow from the color palette by clicking the left mouse button.

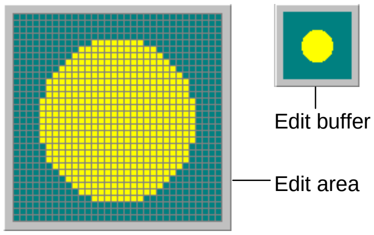
   The left color in the indicator shows yellow as the primary color.

●

2. Select the Filled Ellipse tool from the Tools Palette, then move the mouse to the editing area.

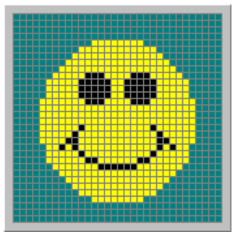
3. Click the left mouse button and drag it to create a circle.

   Yellow circles appear in both the Edit Area and the Edit Buffer. The Edit Buffer displays the icon in the same size in which it will appear on the screen.



4. Select black as the secondary color by clicking with the **right** mouse button on the black color in the color palette.

5. Use the right mouse button to draw eyes and a mouth for the happy face. Use the Filled Ellipse tool for the eyes and draw the mouth using the Pencil tool.



Note: The green background that you see behind the happy face, which is a primary transparent color, will not show up when the icon is displayed.

Saving the Icon

Let’s save your creation. Saving the icon will require a file name and an entity name. When referring to this icon in the application, you must use its entity name.

1. Select the Save ICO/CUR command from the File menu.

   The Save Icon Entity and File dialog box is displayed.

2. Type **MYICON.ICO** as the file name.

   For the directory path, select the SAMPLES\SSATUTOR\FILES subdirectory which is located below the CA-Visual Objects installed directory (for example, C:\CAVO\SAMPLES\SSATUTOR\FILES).

3. Type **MY_FIRST_ICON** in the Entity Name edit control.

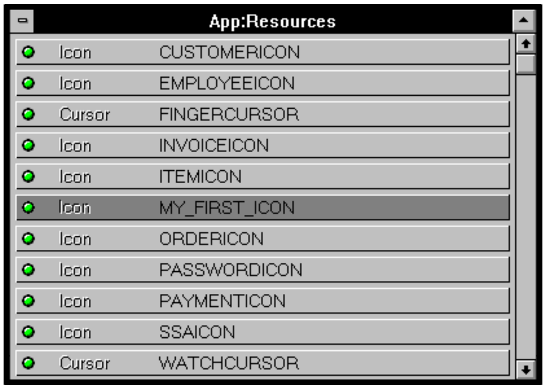
   An entity name may be long and descriptive. All characters are capitalized and spaces are replaced by underscores.

4. Choose OK to save the icon.

   The Save operation creates an icon entity, an icon class and Init() method, a resource entity, and an .ICO file.

5. Close the Icon Editor by double-clicking on its system menu.

6. To view the four new entities, scroll down through the App:Resources Entity Browser. There is an Icon, an RCIcon, a Class, and an Init() method for the class—all named MY_FIRST_ICON:



App:Resources

●	Icon	CUSTOMERICON
●	Icon	EMPLOYEEICON
●	Cursor	FINGERCURSOR
●	Icon	INVOICEICON
●	Icon	ITEMICON
●	Icon	MY_FIRST_ICON
●	Icon	ORDERICON
●	Icon	PASSWORDICON
●	Icon	PAYMENTICON
●	Icon	SSAICON
●	Cursor	WATCHCURSOR

7. Close the App:Resources Entity Browser by double-clicking on its system menu.

8. Your new Icon entity must be made visible to the rest of the application, so rebuild the application using the Build toolbar button.

Attaching Icons to Data Windows

In Windows applications, windows can be minimized. When a window is minimized, an icon is displayed in its place. In CA-Visual Objects, you can specify an icon for each window.

You are now going to attach your happy face icon to the Edit Customer window.

1. From the Module Browser, open the Customer:Forms module by double-clicking on its button.
2. Open the Window Editor by double-clicking on the EditCustomerWindow window entity.
3. In the Data Window Properties window, select the Icon Name property.
4. Type **MY_FIRST_ICON** or click the down arrow button and select MY_FIRST_ICON from the drop-down list.

5. Save your changes by clicking the Save toolbar button and responding Yes when prompted.
6. Close the Window Editor by double-clicking on its system menu.
7. Rebuild the application using the Build toolbar button.

To view the icon you specified, you need to run the program as follows:

1. Click the Execute toolbar button.
2. Log in to the application (Name: **User**, Password: **Trainee**).
3. Open the Customer Browser, by choosing the Open command from the File menu.

   The Open File dialog box appears.
4. Click on the Customer radio button, then choose OK.
5. Select any customer from the browser by clicking on a cell.
6. Click on the Edit button to open the Edit Customer window.
7. Minimize the Edit Customer window to view its icon—the happy face.
8. Close the application by double-clicking on its system menu and then answering Yes when prompted.

Labeling Your Application with an Icon

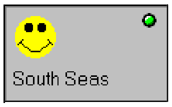
You can use an icon to uniquely identify the South Seas Adventures application in the Application Browser by following these steps:

1. Select the Application Properties toolbar button.

   The Properties of South Seas dialog box is displayed.
2. Click on the Icon button.

   All of the icons defined in the application, as well as any in its include path, will appear in the Icons of South Seas dialog box.
3. Click on the happy face icon, then choose OK.
4. Choose OK to close the Properties of South Seas dialog box.

5. Return to the Application Browser. You will notice that the icon you just selected is now displayed on the South Seas Adventures application button:



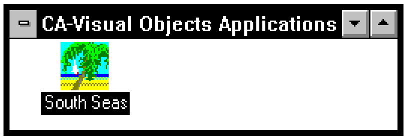
6. Before proceeding, however, change the application icon back to the palm tree. Simply follow steps 1 through 4 above, selecting the palm tree icon in step 3.

Icons in the Program Group

The icon that appears in the Application Browser is also used as the program's icon in the program group window where the resulting .EXE file appears.

1. To view the icon in the program group, create the executable file by choosing the Make EXE toolbar button.

   CA-Visual Objects adds the generated executable to the program group specified in your application's properties where you see its icon.



2. Return to CA-Visual Objects by pressing Alt+Tab.

Attaching Icons to Shell Windows

Just as you can minimize a window within an application, you can minimize the entire application. This capability is implemented by attaching a specific icon to the application's main shell window.

Now, let's attach an icon to the main shell window of the application:

1. From the Module Browser for the South Seas Adventures application, open the Entity Browser for the SSA Shell:Forms module by double-clicking its button.
2. Open the Window Editor by double-clicking the _SSAWindow window entity.
3. In the Shell Window Properties window, select the Icon Name property.
4. Click on the down arrow button and select MY_FIRST_ICON from the drop-down list.
5. Save your changes by choosing the Save toolbar button and responding Yes when prompted.
6. Close the Window Editor by double-clicking on its system menu.

To view the icon you have specified, you need to run the program as follows:

1. Rebuild the application using the Build toolbar button.
2. Click the Execute toolbar button to run the application.
3. After you log in to the South Seas Adventures application (Name: **User**, Password: **Trainee**), minimize the application.

   The happy face icon is now used for the minimized shell window. Clicking on the icon will restore the application.
4. Close the South Seas Adventures application by double-clicking on its system menu and answering Yes when prompted.

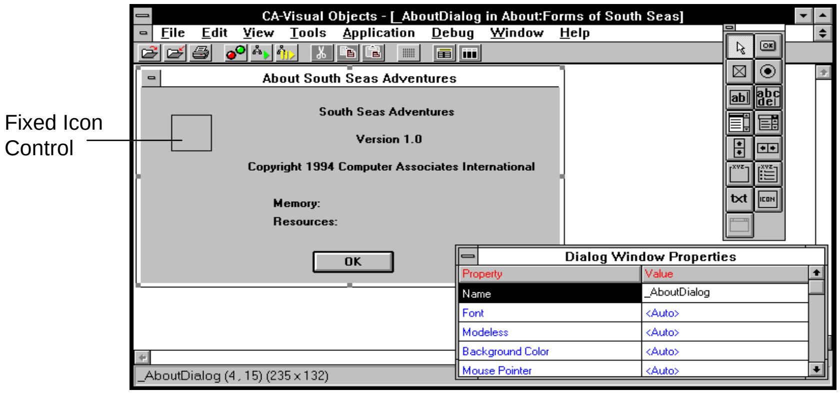
Displaying an Icon on a Window

You can display any icon in a window as a control. An Icon entity name must be provided as the Caption property for a Fixed Icon control. For additional information on controls, see the “Creating and Using Windows” chapter in this guide.

Let’s replace the icon on the About Help dialog window:

1. From the Application Browser, open the South Seas Module Browser by double-clicking its button.
2. Open the About:Forms module by double-clicking its button.
3. Select the _AboutDialog window from the Entity Browser by double-clicking its button.

   The Window Editor appears:



CA-Visual Objects - [_AboutDialog in About:Forms of South Seas]

About South Seas Adventures

South Seas Adventures

Version 1.0

Copyright 1994 Computer Associates International

Memory:

Resources:

Dialog Window Properties

Property	Value
Name	_AboutDialog
Font	<Auto>
Modeless	<Auto>
Background Color	<Auto>
Mouse Pointer	<Auto>

_AboutDialog (4 , 15) (235 x 132)

4. Click on the fixed icon control in the dialog window.
5. Select the Caption property from the Properties window.

   It is currently set to SSAICON, which is the name for the palm tree icon you have seen used throughout the South Seas Adventures application.
6. Enter **MY_FIRST_ICON** to change the icon to the happy face you created earlier in this exercise.
7. Close the Window Editor by double-clicking on its system menu and answering Yes when prompted to save the changes.

To view the icon you specified, run the program as follows:

1. Rebuild the application by clicking on the Build toolbar button.
2. Run the application by clicking on the Execute toolbar button.
3. Log in to the application (Name: **User**, Password: **Trainee**).
4. Select the About command from the Help menu.

   You will notice the happy face icon is displayed in the dialog window.
5. Close the application by double-clicking on its system menu and answering Yes when prompted.

Using Predefined Cursors

Typically, when an application is processing, the hourglass pointer or *cursor* appears. CA-Visual Objects provides this and other predefined pointers. To use them, simply assign your window's Pointer property with the appropriate Pointer object, as in:

```
SELF:Pointer := Pointer{POINTERHOURGLASS}
```

where SELF refers to the current window object.

For a full list of the predefined pointers, see the Pointer Class topic in the CA-Visual Objects online Help.

Creating and Modifying Cursors

You can designate a mouse cursor, or pointer, for a shell window or dialog by using its Mouse Pointer property in the Window Editor. You may also modify a mouse cursor by directly modifying the source code, as is demonstrated here.

To define a mouse cursor of your own for a window, let’s revisit the Icon Editor:

1. Select the App:Resources module of the South Seas Adventures application by clicking on its button in the Module Browser.

2. To create a new cursor, select the Icon Editor command from the Tools menu.

   When you first enter the Icon Editor, it is in Icon Mode so you must switch to Cursor Mode.

3. To switch to Cursor Mode, click on the Cursor Mode button from the tool palette.

   Cursors are limited to only two solid colors. Notice that you can only choose between black and white.

   Also, notice that the top-left pixel in the edit area is marked with a diagonal line. This indicates the *hot spot* of the cursor. When using a cursor, the hot spot is the actual location of the cursor on the screen. For example, the hot spot for the standard arrow pointer is the tip of the arrow.

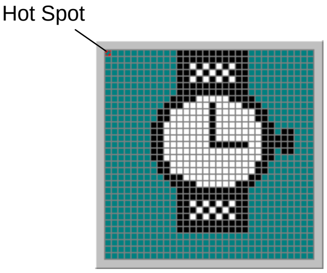
Let’s import a cursor that was installed with the tutorial:

1. Select the Open ICO/CUR command from the File menu.

   The Open Cursor File dialog box appears.

2. Select the WATCH.CUR file in the SAMPLES\SSATUTOR\FILES subdirectory below the CA-Visual Objects installed directory (for example, C:\CAVO\SAMPLES\SSATUTOR\FILES\WATCH.CUR) and choose OK.

   You are going to use this new watch cursor in place of the one that is currently being used in your code. The hot spot for this cursor was mistakenly left in the top-left corner:

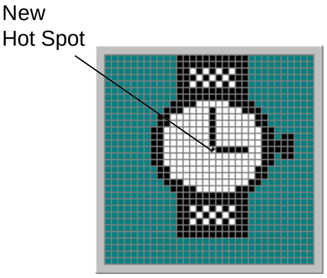


To correct the hot spot:

1. Move the mouse pointer to the intersection point of the hour and minute hands.

2. While pressing the Ctrl key, click the left mouse button.

   This places the hot spot at the pointer location, as seen below:



   The hot spot is now in the center of the watch.

3. To save the new cursor definition, select the Save ICO/CUR command from the File menu.

The Save Cursor Entity and File dialog box appears. As with saving icons, you must specify a file name and an entity name.

4. Type **NEWWATCH.CUR** in the File Name edit control and **WATCH_CURSOR** in the Entity Name edit control.

   The entity name is used when referring to the cursor within your code.

5. Choose OK.

6. Close the Icon Editor by double-clicking on its system menu.

The SSAWindow:OptionsReindex() method currently uses the standard hourglass cursor to indicate that processing is taking place. You are going to change this source code to use your watch cursor instead:

1. Close the App:Resource Entity Browser by double-clicking on its system menu.

2. Open the SSA Shell:Forms module.

3. Open the Source Code Editor by double-clicking on the SSAWindow:OptionsReindex() method.

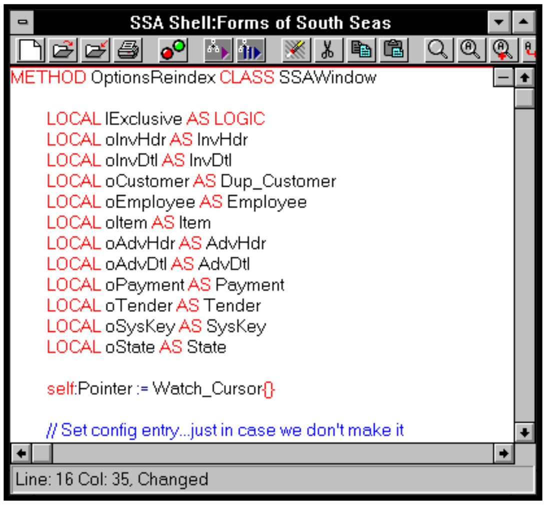
4. At or about line 16 of the method (just after the declarations), replace the following line:

   ```
   SELF:Pointer := Pointer{POINTERHOURGLASS}
   ```

   with

   ```
   SELF:Pointer := Watch_Cursor{}
   ```

Your method now looks like this:



```
METHOD OptionsReindex CLASS SSAWindow

    LOCAL lExclusive AS LOGIC
    LOCAL oInvHdr AS InvHdr
    LOCAL oInvDtl AS InvDtl
    LOCAL oCustomer AS Dup_Customer
    LOCAL oEmployee AS Employee
    LOCAL oItem AS Item
    LOCAL oAdvHdr AS AdvHdr
    LOCAL oAdvDtl AS AdvDtl
    LOCAL oPayment AS Payment
    LOCAL oTender AS Tender
    LOCAL oSysKey AS SysKey
    LOCAL oState AS State

    self:Pointer := Watch_Cursor{}

    // Set config entry...just in case we don't make it
```

Line: 16 Col: 35, Changed

5. Save your changes by choosing the Save command from the File menu.

6. Close the Source Code Editor by double-clicking on its system menu.

7. Rebuild the application using the Build toolbar button.

8. To run the program, click on the Execute toolbar button.

9. Log in to the application (Name: **User**, Password: **Trainee**).

10. Select the Re-index Database command from the Options menu.

    Your watch cursor is now visible for the duration of the process.

11. Exit the application by double-clicking on its system menu and answering Yes when prompted.

Summary

In this lesson, you learned how to create icons and cursors. You used icons to depict minimized windows, placed an icon in a fixed control, customized the Application Browser button using a specialized icon, defined the icon to use for the generated .EXE file in the Windows program group, and you modified a cursor by redefining its hot spot and used this cursor as an indicator for a particular process in the application.

In the following lesson, using the DrawObject classes, you will learn how to create, resize, and modify bitmap and text objects.