

Copyright (c) 1991, Impact Software.

Windows is a trademark of Microsoft Corporation

**ICON MANAGER**  
**Version 2.1**

# CONTENTS

## INTRODUCTION

Chapter 1: Welcome to Icon Manager!

## INSTALLATION

Chapter 2: Setup Procedure  
2.1 Hardware and Software That You Will Need  
2.2 Installing Icon Manager

## OPERATION

Chapter 3: Introduction to Icon Manager  
3.1 The Main Display and Menu  
3.11 The File Menu  
3.12 The Window Menu  
3.2 The Button Bar  
3.3 Working with Icon Manager Files

Chapter 4: The Bitmap Tool  
4.1 The Viewport  
4.2 The Bitmap Tool Edit Menu

Chapter 5: Icon File Formats

Chapter 6: Putting it All Together  
6.1 Customizing Program Manager  
6.2 Using Icon Files with Program Manager  
6.3 Working In Paintbrush

## REGISTRATION

Chapter 7: Registration  
7.1 Licensing Options  
7.2 Registration Form

# INTRODUCTION

## CHAPTER 1 Welcome to Icon Manager!

Icon Manager provides you with a complete set of tools which allow you to create, modify, extract, and organize full-color icons for use with Microsoft Windows 3.0. Special attention has been given to attempt to make every feature as straight-forward and easy to use as possible. To fully enjoy the power and features of Icon Manager please take a few minutes to read the installation procedure and especially the information provided in Chapter 5 on the different file formats that Icon Manager supports and Chapter 6, "Putting it All Together".

Icon Manager features include:

- Support for four different file formats:
  - (1) **\*.ICO** Icon resource files (\*.ICO) usable by Program Manager, which contain only one type of icon.
  - (2) **\*.ICL** Icon library files, also usable by Windows Program Manager, which can contain multiple icon images.
  - (3) **\*.ICA** Icon archive files which allow you to store multiple icons like libraries but save over 20% of your disk space (save over 30% compared to icon resource files). Additionally, archive files remember the exact placement of each of your icons as well as the size of the document window.
  - (4) **\*.EXE** Icon Manager will extract all icons it can find in Windows executable programs from any version of Windows. These images can then be edited or copied directly into any of the three file types described above.
- A Multiple Document Interface (MDI) design which allows up to 30 files to be open simultaneously and provides auto-arrange functions for open and iconic windows.
- A straight-forward "drag 'n' drop" interface allows icons to be effortlessly moved or copied from file to file or in and out of the bitmap tool.
- The bitmap tool provides import and export support for engineering icons in conjunction with paint applications like Windows Paintbrush. The clipboard is used to transport bitmap images between applications. Special support is provided for creating the transparent and inverse screen attributes of icons.

- A screen capture feature allows you to easily duplicate any area of your video display into the bitmap tool.
- A convenient "button bar" provides quick access to frequently used editing switches.

# INSTALLATION

## CHAPTER 2 Setup Procedure

### 2.1 Hardware and Software That You Will Need

Icon Manager requires an IBM Personal Computer or compatible, equipped with the Microsoft Windows operating environment, level 3.0 or higher. A color monitor of at least EGA resolution is recommended. Although Windows will make adjustments internally to display Icon Manager images on CGA or monochrome monitors, an extra level of effort is required by users to conform to the restrictions imposed by these types of monitors. Icon Manager creates icons of the highest color range and resolution currently supported by Windows for the best possible end results on high-performance color monitors.

The Icon Manager program and documentation files require approximately 150K bytes of file space. If you do not currently have a version of the Windows common dialog library (COMMDDL.DLL) installed on your system, an extra 90K bytes (total of 240K bytes) will be required. Maintaining icon images will require more disk space depending on the quantity and format of the icons. More information on icon disk requirements can be found in Chapter 4, Icon File Format Considerations.

### 2.2 Installing Icon Manager

Icon Manager includes a Windows-based installation program named SETUP which should be used to copy the application files onto your hard disk. In addition to installing Icon Manager, SETUP executes routines supplied by Microsoft that will install or update the Windows common dialogs, if required.

1. Insert the program disk into a floppy drive on your system. Select the 'File' menu from Program Manager or File Manager. Choose 'Run...'. Enter "A:SETUP" (or B:SETUP) when Windows prompts for the command line.
2. SETUP will prompt you for the directory on your hard drive that you would like to install Icon Manager to. The default directory suggested is "C:\ICONMAN", however you may specify any directory on a valid disk drive that you would like.
3. SETUP will extract the Icon Manager files to your hard drive and update the Windows Common Dialog Library (COMMDDL.DLL) if required. If Program Manager is present, SETUP will also install an Icon Manager program group and icon on your desktop.

# OPERATION

## CHAPTER 3 Introduction to Icon Manager

### 3.1 The Main Display and Menu

Icon Manager displays an empty main window with a menu at the top and a button bar down the left side when started. The main menu bar contains two primary selections, "File" and "Window". Both contain standard sub-menus for their class within Windows.

#### 3.1.1 The File Menu

The "File" menu contains selections allowing files to be created, opened, closed, saved, and renamed ("Save As..."). Additional selections are provided for exiting the program and displaying the "About..." dialog.

When opening or saving a file, a dialog is provided which allows the selection of the file format and the disk directory to be used for the open or save procedure. The combo-box which appears in the lower left corner of the dialog allows the choice of the file format to be opened or to be used when saving the file. The "Program" format is not available during a "Save" or "Save As..." operation. The listboxes provide easy access to select an existing file for action or to change the file drive or directory.

In addition to opening individual files, a check box labelled "Open All" will be activated when the icon resource file format is selected and multiple icon resource files exist in the current directory. Use "Open All" to open an untitled document which contains all of the icon resource files in the current directory. This command is provided to assist users establish icon archive and library files using from their collection of \*.ICO files.

Opening a \*.ICO file which contains multiple icon formats (i.e., CGA/EGA/VGA images of the same icon) will cause a new document window to be opened in Icon Manager in which the icons will be displayed. If an icon resource file is opened which contains only one icon image, the icon will be placed into the screen cursor, much like dragging an icon for a copy or move operation. Click the left mouse button over the area you would like to drop the icon. If you do not have an open file to place the icon, you can drop it into the Bitmap Tool temporarily.

When you open a multiple icon file for editing by Icon Manager, the file is read into a temporary workspace area where all editing changes are made without affecting the original file. The file must be saved to make any modifications made during the sessions permanent. You can make changes to your files and then save them only when you are satisfied with the results. Icon Manager will remind you to consider saving a file if you forget and try to close the file or

exit the program without saving changes that you have made.

When saving an icon file, the icon resource file radio button will only be active and available if you are saving a file containing a single icon image. This file format will not accept multiple icon images.

### **3.12 The Window Menu**

The "Window" menu contains standard selections for arranging and selecting the windows within a Multiple Document Interface program. An additional menu selection, "Arrange Window", is provided to arrange the icons within the currently active document window.

The "Tile" and "Cascade" options are both used to automatically arrange the open document Windows within Icon Manager. "Tile" will size all of the open windows to approximately equal size and place them flush against each other across the entire main window. "Cascade" will arrange the Windows in a "waterfall" pattern beginning from the upper left hand corner of the main window, each window on top of each other but slightly lower and offset to the right so that you can easily select any window by clicking on its title bar.

The "Arrange Icons" selection will arrange any minimized document windows appearing as icons in a neatly organized pattern across the bottom of the main window. This selection is a standard option provided by the Multiple Document Interface and should not be confused with the next selection, "Arrange Window." The latter selection is an extra feature which has been added to Icon manager's menu to allow the arrangement of the icons that you have placed within a document. Choosing "Arrange Window" will make sure that all icons in your document are positioned in an even pattern with consistent spacing between each image. Note the icons are not associated with a document position in their native file types, only icons stored in an icon archive file are guaranteed to appear in the same position between saving the file and reopening it. You can force the sequence of an icon library file by choosing the "Arrange Window" command before saving. Icon Manager performs an internal sort when arranging a window so the icons will appear in the same order when using the "View Next" command within Program Manager.

Select the "Close All" option to clear your main workspace and begin a fresh editing session. Icon Manager will prompt you to save any files that have been modified before closing them.

### **3.2 The Button Bar**

A button bar is displayed to the left of the main window containing three buttons and a "trash can." The top two buttons are used to toggle the action invoked when you drag an icon within Icon Manager. The top button selects a "copy" operation, while the button directly underneath selects a "move" operation. Each button displays the current selection state by visual depression as well as a green indicator for a selected button or a red indicator



if deselected. Please note that some operations are restricted by the type of file being used. You cannot "move" an icon from an executable (because it never really is removed from the program) nor can you drag an icon into an executable file.

Directly under the copy/move buttons is a third button which conveniently toggles the Bitmap Tool on when required or out of your way for maximum viewing/workspace area.

Under the Bitmap Tool button is a trash can area. Use the trash can with the "move" action selected to remove unwanted icons from your files (or select the copy operation if you just want to have fun throwing things into the trash can!). Simply drag the icon over the trash can and then drop it and Icon Manager will dispose of the image.

### 3.3 Working with Icon Manager Files

Organizing icons using Icon Manager is a breeze! First, select either a "copy" or "move" operation by clicking the appropriate button at the top of the button bar. Now, simply point at the icon you wish to copy or move, click the left mouse button down, drag the icon to where you would like it to be, and release the left mouse button to "drop" the icon into its new location. This drag and drop procedure can be used to move or copy an icon within its original document, between documents, or in and out of the Bitmap Tool (described in the next chapter). The cursor will change to a monochrome representation of the icon during the drag operation to provide visual feedback that it is being moved. This cursor shape will revert to a "DO NOT" image of a circle with a diagonal line through it whenever the cursor is positioned over an area of the screen which cannot accept the icon. When dragging an icon from an executable file, the icon will temporarily display in monochrome to confirm the operation, and then switch to the "do not" cursor style because the icon cannot be copied into the executable file. Move the cursor to the Bitmap Tool or another document window type to continue your copy operation.

As you fill an icon window on your display you will probably want to size and shape the window as it becomes full, making room for more icons. Experiment using the window sizing frames and the "Arrange Window" command to become comfortable with "growing" your own icon libraries. You will find that you can drop icons in any area of the icon document, even on top of each other, and later use the "Arrange Window" command to expand the internal size of the icon document and neatly separate each image.

## CHAPTER 4

### The Bitmap Tool

The Bitmap Tool is used to engineer bitmap operations and to allow icons to be created and modified. It can be used to import and export bitmaps to and from paint applications such as Windows Paintbrush where you can create or modify the design of an icon. The Bitmap Tool also provides features for capturing a bitmap directly from your video display, and a tool which allows the transparent/inverse portion of the icon to be created or removed when importing or exporting icon bitmap images.

#### 4.1 The Viewport

The focal point of the Bitmap Tool is a square viewing window called the Viewport. The Viewport is used to display a bitmap image which is precisely 32 by 32 pixels square (the size of a high resolution icon). Any bitmap used by the BT must be at least 32 by 32 pixels in size. Larger images can be worked with, activating scroll bars outside of the Viewport which are used to position the desired 32x32 portion of the bitmap within the Viewport window. Oversized images can be placed into the Viewport using the "Paste" command from the BT "Edit" menu or by using the "Capture" feature of the BT.

#### 4.2 The Bitmap Tool Edit Menu

The Bitmap Tool contains a few standard selections and a couple of special features unique to Icon Manager. The first three choices, "Paste", "Copy All", and "Copy Viewport", refer to the cut and paste operations which are associated with the Windows clipboard. The "Paste" command will paste any currently available bitmap contained in the Windows clipboard into the icon viewport (the image must be at least 32 by 32 pixels in size). The "Copy All" is the reverse of this procedure, placing a copy of the bitmap currently in the Bitmap Tool into the Windows clipboard. "Copy Viewport" is a special instance of the "Copy" command which only transfers the 32 by 32 pixel portion of the bitmap which is visible in the Viewport into the clipboard. Use these three commands to import and export bitmap images between Icon Manager and your favorite paint application.

Following the copy/paste selections is the "Capture" selection. This feature is used to capture bitmap images from your video display into the Bitmap Tool. When you select "Capture", Icon Manager will make its own windows transparent, exposing the entire Windows desktop which remains. It will also turn the cursor into a crosshair "+" shape. Icon Manager will retain control of the cursor to allow you to execute the screen capture. Point the crosshair cursor to one corner of the area you would like to capture. Press the left mouse button down just like you would do when dragging an icon, and drag the cursor to the opposite corner of your capture area. As you drag the cursor you will see the screen flash an inverse rectangle the size of your clip. Release the left mouse button to complete the capture operation. Icon

Manager will reappear with the captured image displayed inside the Bitmap Tool's Viewport. Use the scroll bar controls to position the 32x32 area you wish to work with inside the viewport, or choose the "Copy All" command to copy the entire bitmap that you captured into the clipboard for pasting into another application.

**NOTE:** The inverse display action provided during screen capture is not meant to be used to precisely line up your clip. It is only provided to give you positive feedback that the capture operation is being performed. It would be not be "kosher" to change the color of another window's client area as this may cause extremely undesirable results if the application was updating the area when the change was made. Icon Manager can only briefly invert these areas during the instant that it has control of the system, and must revert the screen to its original colors before relinquishing control to Windows. Don't get caught up trying to clip a precise area. Clip an oversized area that insures you have the portion of the screen you need and then address the excess later in Paintbrush or with the Viewport's scroll bar adjustment.

The last two selections appearing on the Bitmap Tool's "Edit" menu are "Add Screen Inverse..." and "Remove Screen Inverse... ." These options are used to create the transparent and inverse screen areas of an icon, or to substitute a color from the 16 color palette for these areas when editing them in a paint application. Using both a transparent and an inverse screen portion in your icon will limit your icon design to fourteen colors. This should not be too restrictive for most icon designing. The only icon we are aware of today that uses all sixteen colors is for the program "Mypal", a color palette application, and it does not use any screen or inverse screen area.

The dialogs for color/screen substitution are straight-forward, allowing you to easily point and click on the color(s) you want to exchange. A "Reset" button can be used to start over, and a check box labelled "Available Colors Only" can be activated to display only the colors which have not been used within your icon image. This is a handy feature to use when you are substituting colors for the transparent/inverse areas before exporting an icon to Paintbrush. By selecting "Available Colors Only", you can quickly see what colors can be used to uniquely identify these areas.

The background color that is used to preview your image against is initially set to gray. You can change this color to examine how the icon will appear against different backgrounds by clicking any of the color buttons with the right mouse button.

## CHAPTER 5

### Icon File Formats

Icon Manager can read four different file formats and will create or write three of these. The following section describes each of the file formats in detail and explains a few strategies behind using each type.

#### **Icon Resource File**

An icon resource file can contain no more than one icon, but may contain several different representations of that icon suitable for display on different types of video monitors. Icon resource files commonly have the file extension of "ICO". These files were originally creatable only by the SDKPAINT program which accompanies the Microsoft Software Development Kit. Several utilities are now available from both shareware and commercial vendors that allow manipulation of icons and their data. Many of the icons which are currently available through shareware channels were produced by PBIcon (Icon Manager's predecessor) or other shareware programs. These icons contain only one representation of the icon (the high resolution 32 by 32 pixel, 16 color format) used by most color monitors of EGA or higher resolution. Other commercial packages provide capabilities to software designers which allow the creation of up to three additional representations of icons to be stored into these resource files. These are 32x32 monochrome (2 color), 32x32 three plane color (8 color), 32x32 four plane color (16 color) and 32x16 CGA (2 color). Windows examines the different representations contained in the file and chooses the best one for the current display at program execution time. This allows the designer the ability take advantage of color capability and to avoid using colors which will not appear as intended without worrying which video display it will be used with.

The size of an icon resource file depends directly on what type and how many representations of the icon it contains. For the readily available icons described above which contain a 32x32 16 color image only, that size is 766 bytes. One portion of the data which makes up 128 of these bytes is the "AND" mask. This data is a 32x32 monochrome bitmap used to provide screen transparency and inverse screen color to the color portion of the icon. The capability to create or alter this portion of the icon has been over-looked by the early shareware programs. Most users do not notice the lack of an icon's transparent area because they use the default application workspace color of white that Windows installs with. You can easily spot the difference between icons that do or do not have a transparency mask by changing your application workspace color to light gray or cyan from the Control Panel's colors module. Icon Manager allows you to create a transparency mask by using any one or two colors of the 16 color palette available to you to specify transparent or inverse screen icon areas.

#### **Icon Library**

An icon library is a collection of icon resource files which have been specially formatted with header and directory information to appear to Windows as a

runtime link library of resources, or DLL. The header information and two types of directories (one for the icon resource files and another for the individual icons representations) require an additional amount of disk space above and beyond the size of the resource files alone. The advantage of an icon library is the ability to collect or group multiple icons (by category perhaps) to allow the user to select an icon from the file using the 'View Next' button within Program Manager's dialog box. Of course, storing too many icons in an icon library or including icons for a wide-variety of applications can make this process tedious and time consuming for the user. This is one of the tasks that Icon Manager is meant to ease. By organizing your icons into icon resource files and icon libraries in advance using Icon Manager, it should be much easier to find and select the icon or icons that best meet your needs, no matter how extensive your icon libraries become.

Icon libraries are stored using the file extension of "ICL" by Icon Manager. Programs offered by other vendors may also create icon library files, such as Icon Designer by hDC, which uses the extension of "IL". The use of different extensions is not meant to add confusion to their usage, but to maintain a distinction between processors. Icon Manager will read icon library files which are created by the current 1.0 release of Icon Designer. We do NOT recommend a casual interchange of data between the two programs, nor do we commit to supporting the use of icon libraries maintained by Icon Manager within any other application.

### **Icon Archive**

An icon archive file is a special file format maintained and used by Icon Manager to help you get the most out of your disk space and program operation. These files are stored using the file extension of "ICA." Because Icon Manager deals exclusively with high-resolution color icons and does not need to assemble the resources into sophisticated link directories, icon archive files can save you over 20% of the disk space that an icon library would require. This savings is even greater when compared to storing individual icon resource files (something that might surprise you) due to the overhead required by individual file space allocation on your hard drive. A typical hard drive allocates file space in 512 byte sectors. Two sectors are required for a 766 byte file, of which 258 bytes, or over 33% of the total disk space, is wasted.

**The most efficient way to manage icons if you are primarily concerned with disk space is to store only those icons that you are currently using in icon resource files and keep all of your remaining icons in icon archive files.** Icon libraries can be used discriminantly where convenient, and can require less space than multiple icon resource files if you are changing your icons frequently.

You should use icon archive files to store icons that you do not require on a regular basis, but want to keep available for possible future use. In addition to saving disk space, archive files use a small portion of their data area to maintain information about where each icon has been placed within your library "window", as well as the size of the window when you last saved the

file. This adds one more convenience to the operation of these files.

**Executable**

Icon Manager will read and extract icons from both pre-3.0 Windows executables as well as current programs. Icon Manager process each icon representation that it finds within the executable, and converts it if necessary, to either a monochrome or color 32x32 pixel icon resource in the current 3.0 format. Yes, although Icon Manager likes to work with 16 color icon formats, it will preserve the attributes of a monochrome icon which it reads from either an existing icon library file or executable file, until such time that you edit that icon using the bitmap tool to export it to your paint application. No need to waste any more space until it's needed!

Icon Manager will not write icons back into executable files.

## CHAPTER 6

### Putting it All Together

The following sections provide information to help you get the most out of using Icon Manager within your Windows environment. We begin by explaining some strategies of organizing your Program Manager desktop. Next is an explanation of how to use other Windows programs and files created by Icon Manager to customize your Program Manager desktop. Don't look for this information in your Windows User's guide, it isn't there. The last section of the chapter is devoted to special tips to help you use Windows Paintbrush to create and modify icon designs.

#### 6.1 Customizing Program Manager

The whole concept behind Icon Manager is to provide the tools required to allow you to customize your desktop within Program Manager. Adding custom icons to your desktop is not the only important topic here. Customizing your desktop using the "Properties" dialog of Program Manager can significantly improve your user interface through organization of your program groups and specialization of the icons they contain.

Let's begin by taking a look at the program groups that comprise a standard Windows installation. These are the Main, Accessories, Games, Windows Applications, and Non-Windows Application groups. They provide an organizational scheme that meets the requirements of your initial Windows installation. If you have not made any significant changes to this configuration, you are missing out on much of the power Program Manager is meant to provide.

Few users actually run all of the programs that are installed as icons into program groups. The best way to start creating the most potent desktop for your computer is to eliminate the program icons that you do not currently have a use for, and to collect the infrequently used icons into a group of their own. Program items and their icons require Windows resources that could be used by your applications when they execute. This isn't meant to suggest that you should remove all of your program groups and items to give your applications more memory. It is only mentioned to make you more aware of considerations which you might find interesting and prudent. Deleting the program icons from Program Manager will not delete the programs from your system, it only removes the 'quick access' startup features of Program Manager. You can restore any program items that you delete whenever you desire.

You will most likely want to organize your desktop by eliminating, renaming, and adding program groups. But before we do that, let's consider new program items that you will want to add. Any DOS applications that you run from within Windows can be entered into a program group and depicted with a custom icon. You may want to follow the standard installation format of



placing these items into a "DOS" program group, however you may find that placing these programs into groups related to your own tasks provides a friendlier and more efficient interface.

In addition to adding DOS program items, consider adding specialized program items that provide initialization information to your applications. Most Windows applications accept command line parameters which are processed by the programs when they are started. The most straight-forward command line parameter is a filename to begin processing. For example, Windows Notepad and Windows Write will accept the name of a text document to load at startup rather than entering "File", "Open...", and so forth after starting the program. If you run programs like these and frequently start with a beginning "template" document of some kind, adding a program item which specifies the command line information can be a very powerful tool. This becomes even more evident with more sophisticated applications such as Microsoft Excel or Word for Windows. The macro facilities that are supported by these programs can allow an entire customized process to be initiated by specifying command line document and/or workspace parameters. You may choose to create several instances of the same program in different program items to specify different types of tasks which the program initializes for. Refer to the manuals for your Windows applications to see what command line parameters are accepted and consider delving further into any macro capabilities supported by programs that you use frequently.

The most suitable desktop layout will be different for everyone. There is no rule of thumb as to how many groups you should have or how many items in each group. You will find the best layout for yourself by refining your desktop over time. It is best to avoid a large number of groups containing few items in each; this not only wastes system resources in a manner of speaking, in most instances it is also more difficult to work in a more cluttered environment.

There are several short-cut methods to placing program items within groups. One of these is simply dragging the icon of the program from File Manager and dropping it into a program group of Program Manager. The most straight-forward way to create a program group or item is by selecting the "New..." dialog selection from Program Manager's "File" menu and typing in the information directly. (For information on managing program groups and items refer to Chapter 3 of your Windows User's Guide).

## **6.2 Using Icon Files with Program Manager**

The focal point of customizing program items is the "Properties..." dialog selection under Program Manager's "File" menu. This is one area that the Windows User's Guide does not cover in much detail. Before you add custom changes to program items, first create your program groups and enter "raw" program items into the groups using any of the procedures described in the Windows User's Guide. The following information describes how you can add command line parameters and customize the icons for your program items after this foundation has been laid.

First, highlight the program item that you wish to customize. Now select "Properties..." from the "File" menu. A dialog box will appear which contains two text edit boxes. The first box can be used to enter the title that you would like to be displayed underneath the icon in the program group. You can use a custom title instead of the default program name by choosing a short, descriptive title of the action or task that the icon represents. Choosing too long of a title will cause the text of adjacent icons to overlap each other. The second text edit box is used to enter the command line for the program. This always includes the name of the program (the fully qualified pathname is required if the program is not in your Windows directory or in your DOS program search path). It can optionally contain added command line parameters for the application to use at startup (e.g., "WRITE.EXE BUSLETR.WRI"). If you are satisfied with the information that you enter into the edit controls and you do not wish to change the icon at this time, select "OK" to update the program item with the new information.

To select a different icon for the program item, select the "Change Icon..." button from the properties dialog. A dialog will be displayed which includes an edit box that initially contains the name of your Windows application, or "PROGMAN.EXE" if you have added a DOS application. This filename is the location where Program Manager obtains an icon for its display. Windows programs can contain more than one icon. Selecting the "View Next" button will step through the available icons in the file, one at a time. When you see the icon that you would like to choose, select "OK" to confirm your choice. You will have to select "OK" once again to close the properties dialog and complete the update.

The icon information is transferred into one of Program Manager's data files when the program item is created or updated. It is read from the file specified in the "Change Icon" dialog, not from the command line of the "Properties" dialog. This means that you can replace this name with the name of any other file containing icons that is in a format which Program Manager can process. This includes other Windows programs, icon resource files (\*.ICO), and specially constructed icon library files.

To change the icon of a program item to one found in another file, follow the instructions above for selecting the "Properties..." dialog and the "Change Icon" dialog. Enter the name of another Windows program, or an icon resource or library file (\*.ICO or \*.ICL) created with Icon Manager. (\*.ICA icon archive files cannot be read by Program Manager). Select the "View Next" button and you should see a new icon appear in the preview area. Select "View Icon" again for Windows programs or icon libraries to select other icons in the file. If a message box appears saying "No icons found in file.", double-check the name you entered into the file name edit box to be sure that it is correct. You may have to specify the complete drive and pathname of the file if it is not in your Windows directory or your DOS program path.

When you see the icon that you would like to use in the preview area of the "Change Icon" dialog, select "OK" to confirm your selection and then "OK" once again on the "Properties" dialog to complete the update.

### **6.3 Working in Paintbrush**

Windows Paintbrush (or any other Windows paint application which can use the bitmap data format) can be used to create or modify icons. Here at Impact Software we felt that more could be done to support icon users by working on icon management tools rather than by designing a complete paint application just to add a few extra features (i.e., defining the transparent/inverse screen portion of the icon). Windows Paintbrush provides an extremely rich set of drawing tools in addition to the pixel by pixel editing which is found in most icon paint programs. In the future, OLE (Object Linking & Embedding) additions to Microsoft Windows will allow programs like Icon Manager and Windows Paintbrush to seamlessly operate together as one program.

This section provides a few tips and tricks to make creating and modifying icons using Windows Paintbrush a little easier. You should consult your Windows User Guide (Part II, Windows Accessories, Chapter 8, Paintbrush) for a complete description of how to use Paintbrush and its features.

### **Pasting Into Paintbrush**

When you initially "Paste" an image into Paintbrush it will be placed flush in the upper left corner of the drawing area. This is NOT the ideal position to conduct editing. When you select the "Zoom In" option Paintbrush wants to use this upper left area to provide a true-life sized display of the zoomed area for your examination. Attempting to zoom in to this location in the drawing area produces undesirable operation of the editing process. The very first action you should perform when pasting an icon into Paintbrush is to position the cursor over the icon image, depress the left mouse button, and drag the image to the center of your screen (or maybe the bottom right corner if you intend to use the "Shrink + Grow" feature to produce an oversized image).

### **Sizing Considerations**

Icon images used by Icon Manager must be 32 by 32 pixels in size. There is no magic algorithm that can be used to shrink a large picture into a small icon while retaining all of the information contained in the larger picture. Keep your images 32 by 32 pixels whenever possible to avoid messy resizing operations. This is not meant to imply that you shouldn't use oversized images to work with. Using the "Zoom In" option is a must for editing these small images however this feature limits the drawing tools to only the paintbrush and the fill tool. You should find your own comfort level between the convenience of using oversized images and the difficulty that you associate with resizing to and from the 32 pixel square icon image. If you choose to work on oversized images and later "shrink" to icon size the following tips will help you along the way.

### **Using 32X32.BMP**

A bitmap image is provided with Icon Manager in the file 32X32.BMP which can help you create and size your icons. This image can be placed onto your drawing area by selecting "Paste From..." on the "Edit" menu and selecting the file 32X32.BMP from your "d:\ICONS" sub-directory. As when pasting an icon into the drawing area, immediately drag the pasted image from the upper left corner into a more suitable position in your drawing area. This bitmap consists of a black square surrounded by a dotted red square. The area INCLUDING the black line is exactly 32 by 32 pixels square. This represents the actual area of your icon image. The dotted red line represents the beginning of the area OUTSIDE of the icon.

The reason for providing two separate outlines becomes clear when you want to use Paintbrush's Pick tool if you are going to "Shrink + Grow." The clipping rectangle of the Pick tool does not work consistently across all four boundaries of the rectangle. You will find the the tool includes pixels which are contained underneath it's top and left boundaries, but does not include pixels which are directly underneath the right and bottom boundaries. The 32x32.BMP bitmap can help you position the crosshair cursor more accurately during a Pick operation.

### **Picking to Shrink + Grow**

To "Grow" and image, select the Pick tool and position the crosshair cursor directly over the top left corner of the dotted line, and then gently easing the

mouse one pixel down and one pixel left. When the crosshair is in place, gently depress the left mouse button to begin the Pick operation without disturbing the crosshair. Now drag the crosshair down to the bottom right corner, this time placing the left and upper extensions of the crosshair directly on top of the dotted red line (remember, these pixels will not be included in the pick operation). Now you can select "Shrink + Grow" from the "Pick" menu. Create a new, larger square by defining

a new image area with the cursor while holding down the SHIFT key to maintain a perfectly square area. When you release the left mouse button you should acquire a perfectly resized image.

Use a similar approach to reversing this operation. Carefully choose the upper left corner of your "from" pick and use the SHIFT key to cut out a perfect square. Line up the destination area within 32X32.BMP just as you would to pick an image from it (see instructions above). Shrinking an image almost always results in minor variances to the position of pixel colors when multiple colors appear close together or intermixed. You will always want to examine the icon image with the "Zoom In" feature to examine the image for areas which can be improved with a small "touch up."

### **Copying an Icon Out of Paintbrush**

Before you copy your icon to the clipboard, remember to consider preparing a transparent area for the icon. Most icons benefit from having their outer areas transparent, avoiding the "white square" often seen by users who select a color other than the default white for their application workspace background. You may also opt to create an inverse screen portion for your icon, although this is rarely used. Simply paint the transparent and inverse screen areas with colors that are not used in the rest of your icon. You can specify these colors later in Icon manager's Bitmap Tool as the screen and/or inverse areas to complete the construction of the icon.

Copying an icon out of Paintbrush is a simple operation. Choose the Pick tool and "clip" an area which includes your icon. There is no need to make a precise clip as long as the clip contains all of the image area for your icon. Choose "Copy" from the Paintbrush "Edit" menu. Now activate Icon Manager and select "Paste" from the Bitmap Tool menu. The image will appear in the Viewport. If the image is larger than 32 by 32 pixels, the scroll bar controls will activate to allow you to smoothly scroll the desired 32 pixel square into view.

# **REGISTRATION**

## **CHAPTER 7 Registration**

### **7.1 Licensing Icon Manager**

If you use this software and find it of value you may obtain a license for its use and support the continuing development of quality Windows shareware. Our licensing agreement appears on the following registration form along with pricing information for the software license and related materials.

### **7.2 Registration Form** (Printed on following page)

License request for Icon Manager  
Copyright (c) 1991-92 by Leonard A. Gray

Icon Manager is licensed on a per user basis. The license grants the user the right to install and use the software on one or more computers so long as the total number of users does not exceed the license quantity.

Please type or print clearly the following information:

\_\_\_\_\_  
Name

\_\_\_\_\_  
Company (if applicable)

\_\_\_\_\_  
Street

\_\_\_\_\_  
City State/Prov ZIP/Postal Code

\_\_\_\_\_  
Country Phone

\_\_\_\_\_  
Credit Card Number

VISA  MasterCard  Am Ex  
 Diner's Club  Carte Blanche

\_\_\_\_\_  
Expiration Name As Appears on Card

\_\_\_\_\_  
Signature

<b>Type</b>	<b>Qty</b>	<b>Price</b>	<b>Total</b>
-------------	------------	--------------	--------------

Basic Registration	___ x \$19.95	___	___
--------------------	---------------	-----	-----

Program Update Diskette 5.25 __ 3.5 (Extra Bonus! Over 300 original and enhanced icons).	___ x	\$6.50	___	<b>Disk Size:</b> ___
---	-------	--------	-----	-----------------------

\_\_\_\_\_  
Sub-total

\_\_\_\_\_  
California residents  
add 7.5% sales tax.

\_\_\_\_\_  
Outside U.S. and Canada  
add \$3.00 shipping for  
diskette.

**Grand total:**

\_\_\_\_\_



We accept checks and money orders (must be paid through a U.S. bank) in addition to payment by credit card.

**License requests from outside the United States:** Registrations will be processed upon receipt of the current equivalent of U.S. funds in check or money order (must be drawn or paid through a U.S. bank) or credit card request subject to the currency exchange rates used by your credit card company.

Mail your completed registration request to:

Impact Software  
12140 Central Avenue  
Suite 133  
Chino, CA 91710  
U.S.A.

Thank you for supporting the shareware software concept and encouraging the future development of Windows shareware!

Windows is a trademark of Microsoft Corporation

Impact Software  
Chino, California  
U.S.A.

Voice: (714)590-8522 Hours: 8:30 AM - 5:00 PM Pacific Time  
Data: (714)590-0500 24 Hours (BBS: eight bits, no parity, one stop bit)