

IconBar 2.00β

by Dan Porrey Copyright 1994 Expert Systems

This is a preliminary release of this version of the software. I have release this way because it has not yet been tested, nor has the help file been completed. Anyone who does programming in Windows can appreciate the length of time a good help file takes. Sometimes longer than the program itself. I ask that while I'm creating the help file, you look for bugs and/or problems. I may be able to squeeze in extra features before the release, but anything to involved will have to wait for a later release.

Unlike the prior version I am requiring a \$15.00 registration fee for IconBar 2.00. Anyone who registered version 1.00 prior to July 10th, 1994, is automatically registered for this version. Make all checks or money orders payable to **Expert Systems** and send them to:

**Expert Systems
9810 W. 153rd St. 1W
Orland Park, IL. 60462-3625**

Please note that this is a very stable version of IconBar. You should have no problem using this program in place of version 1. Note also that registering for the beta version automatically transfers to the release version when it is available.

Installation

Simply copy IconBar to the directory you wish it to reside. Place the dll file into your **system32** directory (for example: c:\winnt\system32). Make an icon for it and your ready to go.

Using Icon Bar

Below I have listed the added features with brief descriptions on how to use them. The final release of this software will include a detailed help file explaining all new features as well as old ones. I did not go into any great detail since it we take almost as much time to write this as it would to complete the help file. This file is just intended to get you started. I hope that most or all of the features are self explanatory or at least are simple enough to figure out.

Main Menu: (bring up the main menu by clicking the main icon with the right mouse button)

Always On Top

Enabling this will keep the main icon and all program icons on top of all other windows.

Options

This dialog box contains main program settings such as:

- 1) Display of hand cursor when moving an icon
- 2) Disabling the align property when an icon is moved.
- 3) Centering dialog boxes
- 4) Enabling 3D Effects
- 5) Set grid spacing in pixels
- 6) Setting icon dragging threshold (in pixels). This option works in conjunction with option #2 above.

Add New Icon

Click here to add a new undefined icon.

Delete Icon

Delete icons by selecting them from a list and then clicking the delete button. There is also an icon in the shape of an x. Drag this x over the icon you wish to delete and drop it. You will then be prompted on whether or not you wish to delete this icon.

Arrange

A sub menu will pop-up allowing you to align the icons in one of four edges of the screen. Only those icons with their "align icon" property set will be aligned.

Reorder

Select an icon from the list in this dialog box and then move it up and down in the order using the "Move Up" and "Move Down" buttons. The order of the icons is used in two places.

- 1) When the icons are loaded after the app IconBar is started.
- 2) When the arrange icons command is selected.

Mark All

Globally mark settings on all of the icons at one time. This is useful if you wish to align all icons at one time. First mark them all with the align property set, and then issue an arrange command.

The check boxes in this dialog box are three-state boxes. When the dialog box first appears, the check boxes will either be checked, unchecked, or gray. Checked indicates that ALL the icons have this particular property enabled. Unchecked indicates that ALL of the icons have this property disabled. If the check box is gray, then some icons have this property enabled, and some have this property disabled.

To set the icons: place a check in the box to set this property on all of the icons to enabled. Leave the box unchecked to set this property on all of the icons to disabled. To leave the property on each icon as is, click the check-box until it is gray.

Save

This saves the icons information and all of the options. Throughout the running of the program, the icons are saved whenever they are changed, created, or deleted. At times

some or all the information may not be immediately saved to the registry. Choosing this command forces all info to be stored in the registry database. This command is also performed when the program is shut down.

Help

Opens the help file (which will be available shortly).

About IconBar

Opens the about dialog box.

Unload IconBar

Exits IconBar or unloads it from memory. At this time you will be prompted to save the programs settings. These are only the settings in the Options dialog box. The icon information is always saved regardless of this options setting.

Logoff

Logs the current user off the system. This command also prompts you to save the settings.

Icon menu: (bring up the icon menu by clicking the desired icon with the right mouse button)

Parameters

Set the following icon parameters:

File Name

Startup Directory

Command Line Options

Icon File Name and Icon Index (if more than one icon in file)

Icon Name - This is the name of the icon that will appear in the reorder and delete dialog boxes. This name MUST be unique. The program does not check for a unique name. It will check for a name of zero characters which is also not allowed. The name is also used to store the icons parameters in the registry database. I have included this option because it makes it easier to identify the icons.

Environment

Set the environment variables for the application assigned to the icon. You must also click the enable check box to use the strings in this list. Click Default to get a list of the default string that would be assigned if this option was not set. You can then edit the default settings to customize them to your needs.

App Startup

This dialog box gives you options to set the start up state of the application assigned to the icon. You can control the state of the initial window (i.e. Maximized, Minimized, or Normal). You may also specify the position of the startup window by giving upper left corner positions in pixels. Further you may set the height and width of the window in pixels. This option must also be enabled. To the size of the window as a reference do the

following:

- 1) Launch the application
- 2) Choose the App Startup item from the icon menu
- 3) Drag the find window icon over the caption bar of the window
- 4) and drop it. The dialog size values will automatically populate with the current values.

Please note that some applications will override this option. This option is especially useful for console apps.

Icon Position

With this option you can control the position of the icon and those options effecting it's position. The following options are included: Snap To Grid, Align Icon, and Lock icon position. Hopefully these are self explanatory. Another option included is the ability to lock an icon to one of the four edges of the desktop. With this property set the icon can be moved along the edge but not off the edge. Playing with these options should reveal their characteristics.

Cancel

Cancel simply closes this menu easily.

Hidden Features

Drag and Drop

There are two ways to use drag and drop features in IconBar. First, drop files from the File Manager onto the main icon to create new icons. IconBar try to assign as many options as possible to the icon. It will also try to resolve file association. The second way is to drop files on an existing icon. If the extension of the file is DLL, ICO, or EXE the icon will change the Icon File to the file that was dropped. If the file does not have these extensions, then the file(s) dropped will be used as the command line. In both instances the current values are replaced. If multiple files are dropped, all non icon files are placed on the command line. In the case of multiple icon files, the last one in the list is used for the icon file. You may also drop more than one file on the main icon to create many new icons.

Grid

The icons will now snap to a grid set in the options menu and enabled in each icon under Icon Position.

3D Effects

You can now enable or disable 3D Effects while the program is running.

Double Clicking the main icon

Double click the main icon to:

Open the about box;

(while holding the Control key) save the settings. Same as save option in the menu;

(while holding down the Shift) rebuild the icon list in the registry. If the program aborts unexpectedly (either by user intervention or the operating system) it is possible for the icon list in the registry to become corrupted. This will be evident when some or all of the icons do not load when IconBar is started. If this happens, simply start IconBar, double-click the main icon while holding down the shift key, and then unload and restart IconBar.

Launching an App

Launching an app can now be done either by a single click or a double click. Set this in the Options menu item.

Using Environment String in the Parameters

In the parameters dialog box, you can use environment strings in within the arguments. IconBar will try to resolve them if they exist. You cannot however access the environment variables being set by the program since they do not exists until the program is launched. An example is: for an icon you can have %COMSPEC% as the program name, and %HOMEDRIVE%%HOMEPATH% for the startup directory.

Known Problems

- 1) If an icon file (a dll or exe) contains a large number of icons (>100), it will take a considerable amount of time to load. I am currently seeking a solution for this.
- 2) I am looking for a way to register the icon you choose for the program to so it shows up in the task switching dialog and also when the app is minimized. All DOS apps will show the standard icon when they are minimized.
- 3) I have one program that will not launch from IconBar, yet it launches fine from program manager. Obviously Microsoft® takes additional (or different steps) to launch a program. I launch a program in the following way:
 1. A call to SetCurrentDirectory() is made.
 2. A call to CreateProcess() is made. I use the lpszCommandLine argument for both the appname and the command line parameter and leave lpszImageName as NULL.

If you can help with these problems I would appreciate the information.