

NewIcons

Philip A. Sauvageau

Copyright © 1999 Team NewIcons

COLLABORATORS

	<i>TITLE :</i> NewIcons		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Philip A. Sauvageau	August 9, 2022	

REVISION HISTORY

<i>NUMBER</i>	<i>DATE</i>	<i>DESCRIPTION</i>	<i>NAME</i>

Contents

1	NewIcons	1
1.1	NewIcons	1
1.2	Introduction	2
1.3	NewIcon Features	3
1.4	Installation	5
1.5	NewIcons	8
1.6	DefIcons	10
1.7	NewIconsPrefs	12
1.8	CopyNewIcon	14
1.9	CreateDefaultIcon	15
1.10	InjectBrush	15
1.11	KillNewIcon	16
1.12	PatchOpenWB	17
1.13	UpdateDrawers	17
1.14	ShowNI	17
1.15	Tips And Tricks	19
1.16	Color Scheme used with the supplied iconset	25
1.17	DefIcons Brainfile	27
1.18	Authors	30
1.19	Acknowledgements	31
1.20	Distribution	32
1.21	History	32

Chapter 1

NewIcons

1.1 NewIcons

The NewIcons Package
Version 4.6
brought to you by
Team NewIcons



Introduction

Features

Installation

The Main Programs:

NewIcons

DefIcons

The Support Files:

NewIconsPrefs

CopyNewIcon

InjectBrush

CreateDefaultIcon

KillNewIcon

PatchOpenWb

UpdateDrawers

ShowNI
Tips and Tricks
Used Color Scheme
DefIcons Brainfile
Authors
Acknowledgements
Distribution
Version History

1.2 Introduction

Greetings Amiga User!

The NewIcons Package is a revolutionary GUI enhancement system for the Amiga computer. It adds many new features to the Workbench that, we hope, will be standard in the next generation Amiga operating system. Specifically, it allows for a palette independent icon system whereby icons look color correct on any system, no matter what your Workbench palette is set to. It allows you to use icons in up to 256 colors, adds default icons for many, many file types, and works on any Amiga with OS3 and up.

If you have never used NewIcons before, please read the documentation carefully...it will save you from any trouble you should encounter. If you are a seasoned NewIcons user, also be sure to read the documentation carefully...there are a lot of new features in this latest version.

This archive is not "just another set of icons". This is a completely new concept, which we hope will revolutionize, and push in the right direction, the idea of GUI on the Amiga.

BACKGROUND INFORMATION

=====

Kickstart 3.0 was Commodore's first step towards ReTargetable Graphics. There isn't a device independent graphic engine yet, but there is a new pen arbitration system which allows applications to obtain a given color, or the one which is closest to it if there are not enough free pens.

It's simply ridiculous that, on a system capable of displaying a 256 color Workbench screen, and with a pen arbitration system like the one described above, icons are limited to 8 colors. Various solutions have been proposed, ranging from "use some colors and hope they are right" to "allocate some pens

and reserve them for icon display", but they all miss the point, which is:

Why should I, the user, have to care about the icon palette?

Why should I set the system palette in a way decided by someone else (the icon designer) and not the way `_I_` like it?

When you display a picture on the Workbench screen using MultiView, you don't have to care about the palette: the system chooses it for you. So why should one care about the palette used by icons? The system should take care of it, not the user.

Unfortunately, icons don't carry palette information. Therefore, Nicola had to define a new standard for icon files, and write a system patch to make the system recognize them. Several goals had to be reached:

- The new standard should be backward compatible, that is users not running the patch program should see the standard 4-color icons.
- Manipulation of icons by users not running the patch program should not cause the new icon information to be lost.
- Remapping of the icons to the Workbench palette should be as fast as possible.
- Data should be compressed in order to reduce disk usage.

All of the above goal have been reached. The new icon information is stored in the ToolTypes, so it is completely backwards compatible. A library, `newicon.library`, takes care of reading and writing the imagery in the ToolTypes; it also hides away the special ToolTypes so that when one asks for information about an icon only the standard ToolTypes are shown. A patch program, `NewIcons`, intercepts calls to `icon.library`, redirecting them to `newicon.library`, so that Workbench and all other applications transparently get the new icon imagery.

1.3 NewIcon Features

NewIcons

`NewIcons` is a revolutionary system patch which enhances your Workbench and gives the user much more power and configurability over the appearance of the GUI (Graphical User Interface). The `NewIcons` system works transparently, and provides for a standard icon system that will look the same on any system, no matter what the palette is set to.

DefIcons


`DefIcons` complements the `NewIcons` system. It works by recognizing a file's type, and instantly displaying the appropriate icon image for files without icons attached to them.

The system's features include:

- Allows up to 256 color icons on an AGA machine. (!)
 - Icon colors are color correct on an AGA system no matter what your palette is set to. They automatically appear in perfect color.
 - Default icons for dozens of file types automatically appear for files without icons (FAST!). Recognizes file types in a very efficient manner. If you click on a "fake" icon, your appropriate user-defined default tool will appear in the "Execute command" text field automatically!
 - Make a brush, convert it to an icon. It will appear on your workbench in the exact proper colors you created it in, no matter what your palette is set to. Imagine making gorgeous icons from your digitized pics, in more than 16 colors!
 - Icon images are compressed to maximize disk space. They are uncompressed faster than you can say "NewIcons"!
 - Fix that ugly Cross-DOS icon to look like a standard NewIcon.
 - Unique icons for RAM-Disks and CD-ROM's that don't have icons.
 - A complete set of beautiful, standardized Workbench icons, drawn by Phil Vedovatti. Hundreds of icons are also available separately on Aminet, some with a totally different style.
 - Comprehensive set of utilities to allow you to manipulate your NewIcons.
 - Totally FREE! Why would anyone want to pay cash for icons?
 - Dithering of icon images is selectable for users with fewer available Workbench colors. This is turned off by default.
 - Icons can be given a single tooltype, which will tell NewIcons to display a NewIcon image from elsewhere on the system. This feature has some
caveats
, but introduces a huge number of neat possibilities for configuring your Workbench, and saves on disk space and chip RAM. Imagine having your Workbench icons appear randomly different on EACH reboot!
 - The source for deficons.prefs has been included, so any enterprising programmer can add their own file identification types to the system!
 - User configurable, GUI enhanced Preferences program to configure all attributes of the NewIcons system (dither, pen precision, etc.)
 - The NewIcons patch is a commodity, which can be enabled and disabled on the fly, using Commodities Exchange.
-

- RTG (ReTargetable Graphics) mode, which allows graphics card users to store icon data in fast RAM instead of chip RAM. This has the advantage of saving your valuable chip RAM for other uses, and speeds up rendering of icons on the Workbench.
- Border removal option! Remember the Kickstart 1.3 days when there were no gray border boxes around the icons? This new option removes those silly looking boxes once again, and is user configurable.
- The system library is available in different versions, to ensure the best speed possible on every system configuration.
- NEW IN V4: Dragged icons can also be optionally made transparent.
- NEW IN V4: The text shown under the icons can be optionally rendered with a 3D or an outlined look.
- NEW IN V4: New utilities to make updating whole drawers or partitions with a specified drawer image, and a popup menu utility that displays icon images in a menu format -- could even be used as a program dock launcher with the appropriate scripts.
- NEW IN V4: New 32 color icon set. NewIcons is now sufficiently faster to allow fast rendering of higher bitplane icons.

1.4 Installation

To install the package, just double click on the "  Install_NewIcons" icon. The script recognizes if NewIcons is already running, so there is no need for a separate 'update' script for users of previous versions of the programs.

Make sure you have some disk space on your SYS: partition. You should have at least 300K available if you install the entire package. If you don't have enough room on your partition and you still want to install the package, see the

tips and tricks
section

to learn how to make more room for the images. Alternatively, you can install

NewIcons
without the
DefIcons
files.

You will be prompted as to which parts of the distribution you want to install. For maximum performance, it is suggested that you install both NewIcons and DefIcons.

There are various flavors of the newicon.library. The installer will attempt to select the appropriate version for your system. If you have a higher end amiga (AGA or gfx card), you'll be given a choice

between the middle-end or high-end library. If you have a graphics card, or run your AGA Workbench in a higher number of colors, you should select the high-end library. If you require dithering of icon images (low bitplane screens), you can choose the middle-end library.

In this version of the package, we have included a new 32 color icon set. If you have a fast Amiga, you should have no problem with these icons. On 68000 Amigas, you may want to stick with 16 color icons and uncheck the box for installing 32 color icons. A separate archive containing the original 16 color icons is available on Aminet, or from the NewIcons Home Page. A 68030 should be fast enough for the 32 color icon set, and possibly 68020 machines as well.

The installer script will install the icon images on your system. The existing tooltypes of your icons will not be touched by the installation script, and the original icon images will not be removed. You will not see the new icon images until the NewIcons program is run. The installer script will launch NewIcons immediately, so you can view the new images right away. Since your original icon images are not actually deleted, it is easily possible to restore the icons to their original appearance, so don't be intimidated by the installation process.

The NewIcons program will be copied to your c: directory, the preferences editor will be copied to your Prefs/ directory, and newicon.library will be put in your Libs: directory. You'll be given the option of where you would like to put the other icon utilities. If you choose to install DefIcons (highly recommended), the program will be copied to your c: directory and the default icons and brainfile will be put in ENVARC:. Your old default icons will be backed up by the installer.

The program

```
PatchOpenWB
    will be copied to c:,
```

as well as the new ShowNI utility.

Two short commands will be added to your s:user-startup file to launch NewIcons and DefIcons whenever you boot your system.

With this installation is the option to install default drawer images to a user selected directory or partition. The installer will ask you to supply a directory, and then a drawer icon image to choose for that directory. All SUB-drawers of the selected drawer or partition will be affected. Be careful--if you select an entire partition, make sure that you want ALL those drawers changed. While it won't delete any icons or tooltypes, it will wipe out any old Newicon drawer images in those selected subdrawers with the new image. You can continue to update more drawers until you are happy with the look. Wash, rinse, repeat. :)

Also available is an option to install user selected disk icon images to your various hard drive partitions. Simply select the partition, then click on the icon image from the popup menu window, and you're done. You can repeat the process for all your other devices.

Users of old versions of NewIcons! IMPORTANT!:

Since version 3.0, NewIcons requires a change in how it is run in the user-startup script. For technical reasons, it must now be RUN'ned like this:

```
RUN <>NIL: c:NewIcons
```

If you installed NewIcons previously with the installer script, the installer will seek out the ;Begin Newicons line in the user-startup, and will make the change for you automatically. If you have altered your user-startup and removed the ";Begin Newicons" section, or added the NewIcons command elsewhere in your startup, you will have to remove the old NewIcons command from the user-startup script.

If your user-startup script launches NewIcons the old way (without "RUN <>NIL: c:NewIcons"), your Amiga will freeze during bootup.

Note that if you are running a workbench replacement such as ScalOS or Directory Opus Magellan in Workbench replacement mode, these programs directly support the NewIcons system without requiring the NewIcons executable to be run, so you should skip this step in the installation.

Kickstart 3.x users! IMPORTANT:

You need to add a line to your s:startup-sequence to fix a bug in Kickstart 3.x. Add the following line to your startup-sequence, before IPrefs and after SetPatch:

```
C:PatchOpenWB
```

Note that this might not be required if you already have another patch doing the same job, such as the one included in MCP.

Once the installation is complete, reboot your computer and you're ready to have the best looking GUI anywhere!

Don't forget to run the
 Preferences
 program to fine tune your setup.

The installation program only installs images for the standard WB, so some files will likely not be "NewIconized" by the installer script. Don't worry; you can add images to these icons later using the

```
    CopyNewIcon  
    utility supplied in this archive.
```

NewIcons works by installing compressed images of the new icons into tooltype lines of the .info files. Once NewIcons is launched, you will not see these tooltypes, so they will not interfere with your work. If for some reason you are not happy with NewIcons, you can use the utility

```
    KillNewIcon  
    to remove these images from the .info  
files, restoring them to their original appearance.
```

There is a de-installation program which will remove the NewIcons system entirely from your system, EXCEPT for the def_icons contained in envarc:sys. Simply delete them by hand and install your backups of the original default icons (you DID back them up when the installer asked you if you wanted to, right?).

 The big advantage of this system is that when the NewIcons program is run, you see the new icon images. When NewIcons has not been started, you will see the boring old icon you viewed before you received this package. Therefore, icons that are transferred to users not running NewIcons will still be visible as a plain, old-style icon.

1.5 NewIcons

NEWICONS

=====

This program is the actual patch that allow your system to display NewIcons, and is launched from your user-startup. It must be RUN'ned from the user-startup like this:

```
RUN <>NIL: c:newicons
```

It accepts three parameters, and its template is as follows:

```
CX_PRIORITY/N/K,CX_POPKEY/K,PE=PREFSEEDITOR/K
```

CX_PRIORITY/N/K - This is the standard commodity hotkey priority. This parameter prevents conflicts with hotkeys between different programs. Consult your Workbench manual if you don't understand this parameter. The default setting for this parameter is 0, and most likely you won't need to change it.

CX_POPKEY/K - NewIcons is now a commodity, and pressing the specified hotkey will open a window displaying information about NewIcons, the current settings for NewIcons, and displays a gadget to launch the prefs program. The default hotkey is Lalt Shift N.

PE=PREFSEEDITOR/K - This defines where NewIcons should look for its preferences program. By default, it looks for Sys:Prefs/NewIconsPrefs. If you have placed the prefs editor in another location, use this parameter so NewIcons can find it.

Examples:

```

RUN <>NIL: C:NewIcons (most people will use this)
RUN <>NIL: C:NewIcons CX_POPKEY "lalt shift P"
RUN <>NIL: C:NewIcons PE Work:Prefs/NewIconsPrefs

```

The idea behind NewIcons is that we shouldn't need a fixed color scheme for icons anymore. NewIcons will automatically remap icon images using the colors available on the Workbench screen. A specific palette is therefore not required (icons look reasonable even on a 2-color Workbench) but it is suggested to obtain best results. If you use a Workbench with ≥ 64 colors, you can ignore the palette completely. The file "palette.preset" is the palette I use on my Workbench. Note I have changed the pens used by bright and dark edges to have smoother button gadgets, and that color 3 is the same as color 0 so that the shell cursor is of the correct color. This palette uses 7 out of the standard 32 colors used by the NewIcons. The remaining 25 are dynamically allocated at run time.

ECS machine users:

If you are using an ECS machine, it helps to set up a suitable palette. The recommended palette as used by the original 14-colors iconset is:

	Dec		
	R	G	B
	---	---	---
Black	0	0	0
Gray1	2	2	2
Gray2	5	5	5
Gray3	8	8	8
Gray4	10	10	10
Gray5	12	12	12
White	15	15	15
Green	2	8	2
Yellow	13	11	3
Red	13	2	2
Purple	9	4	10
Light Brown	12	9	6
Dark Blue	0	5	9
Light Blue	5	8	11

Of course you can sort them anyway you like, since NewIcons will pick the correct colors from the above palette for you. A suitable palette is supplied in the Extras drawer of this distribution. Note that the iconset included in V4 uses a 32-color palette. However, you can still use it on a screen with 16 colors or less - the icons will be remapped to look as good as possible to fit with your current palette. If you are using the new 32 color icon set with an ECS machine, you won't be able to display all of the colors on your Workbench. The new palette does use the above 14 colors in the palette however, so the icons should look reasonable. Turning dithering mode on might make them look even better.

Another useful feature is the DEFAULTIMAGE tooltip. Any icon that does not have NewIcon image data in it can be given a single tooltip to cause NewIcons to "borrow" an image from another NewIcon icon. To use this feature, you simply add the following tooltip to the icon:

```
DEFAULTIMAGE=<location of image to "borrow">
```

For example, you could direct all your drawer icons to a single image on your harddisk or RAM: disk. Once this is done, you can change the image of all your drawers by changing the single "loaner" icon. You can read more about using this feature

here

.

Workbench replacements users:

If you aren't using the standard Workbench but one of its replacements (such as Directory Opus 5.x or Scalos), running the NewIcons executable might not be required, since your Workbench replacement could have built-in support for NewIcons through newicon.library. In that case, you won't need to run the NewIcons executable. Please refer to your Workbench replacement documentation for more details.

1.6 DefIcons

DEFICONS

=====

DefIcons is run as a single command in the user-startup. It takes no parameters, and does not need to be RUN'ned, since it automatically detaches from the CLI.

The DefIcons patch will magically transform Workbench's silly fake icons (displayed when you pick the menu item Window/Show/All Files) into more meaningful ones. The program not only recognizes files such as projects and tools, but identifies various other file types:

- IFF, JPEG, ANIM, GIF and PNG files
- 8SVX, SMUS, DCMS, Med, and PT/ST mods
- Handlers, Kickfiles, and Filesystems
- ASCII, TeX, AmigaGuide and Postscript files
- Source codes of various languages
- Keymaps, Datatypes, Libraries, and Printer Drivers
- Various crunched filetypes
- Bitmap and Outline Fonts

...and several more.

Just like any "fake" icon, you can't double click on fake icons to start the associated tool immediately, but if you have designated a default tool for the deficon (located in ENVARC:sys), the "Execute

Command" text field will automatically have the default tool entered for you. This feature is very handy for things like text files and picture files that don't have icons attached to them.

Additionally, you can transform a fake icon into a real icon using the

```
CreateDefaultIcon  
utility.
```

DefIcons has a brainfile called `deficons.prefs`, which is copied to `ENVARC:` by the `install_NewIcons` script. Unfortunately, it is currently not possible to easily edit the brainfile, so you can't add your filetypes, but the source code for `deficons.prefs` is included. With a little programming knowledge, you can create your own `deficons` classes. When you install `NewIcons`, the Installer script copies the default `def_xxx` icon images to `ENVARC:.` These icons are used by `DefIcons`.

Apart from fake icons, `DefIcons` has another wonderful feature: if a volume doesn't have a `disk.info` icon, it will replace the standard `def_disk.info` icon used by `Workbench` with a user-defined one. For example, it will use `def_CD0disk.info` for a CD in `CD0:`, and `def_RAMdisk.info` for the Ram Disk. If you did the old 'copy `ramdisk.info` to `ram:disk.info`' trick, you can remove that command from your `startup-sequence`.

`DefIcons` works well with multiple file systems and `MFS`. When a disk without `disk.info` is inserted in `DF0:`, it first searches for an icon named `def_DF0disk.info`; if it can't find it, it looks for an icon named after the file system (`def_DOS0disk.info`, `def_MSD0disk.info`, `def_PFS0disk.info`...) so you can have different icons for different file systems. Latest versions of `CrossDos` use an icon of their own for `MS-DOS` disks. Since the icon is really ugly, you may want to use your `def_MSD0disk.info`. To do that, load `L:crossdosfilesystem` into a hex editor, search for the string `"disk.info"` and replace it with something else, for example `"dusk.info"`.

The disk trick will also work for non-`DOS` disks. In this case, the `basename` is the one displayed by `Workbench` under the icon. For example if `Workbench` shows `"DF0:NDOS"`, the icon named `def_NDOSdisk.info` will be used. `DF0:????` is a special case: `def_BADDisk.info` will be used.

`DefIcons` uses a tree structure when identifying files. For a `JPEG` picture, for example, `DefIcons` will search in order first for `def_jpeg.info`, `def_picture.info` and `def_project.info`. `IFF` files are special, and `DefIcons` will search for `def_XXXX.info`, where `XXXX` is the `FORM` type of the file. If it can't be found, `def_iff.info` is used.

For archived files there are three families: `def_archive` (`lha`, `zip`,...), `def_filearchive` (for `uuencode`, `compress`, `gzip`,...) and `def_diskarchive` (`DMS`). These families can have sons (i.e, `def_lha`, `def_dms`, etc) if you create and add them to your `ENVARC:sys` directory.

`Workbench` replacements users:

`DefIcons` will probably not work properly if you are using a `Workbench`

replacement such as Directory Opus or Scalos. This is not a bug in DefIcons neither in the Workbench replacement! DefIcons was designed as a patch for the standard Workbench. Workbench replacement will have to implement their own system to provide a similar feature. Please refer to your Workbench replacement documentation for information on this.

1.7 NewIconsPrefs

NewIconsPrefs
=====

Since NewIcons V3, various features can be configured through the preferences editor. There are three ways of launching the preferences program for your convenience:

- 1) Start the Commodities Exchange program and select NewIcons. Click on the "Show Interface" gadget, or:
- 2) Double-click on the NewIconsPrefs icon from the Workbench, or:
- 3) Just hit the NewIcons hotkey (default is Left-Alt+shift+N) NewIcons will open a status window showing how many NewIcons are currently open, and all the current options. From here you can click on "OK" to close the window, or click on "Prefs" to launch the preferences program.

Using NewIcons Prefs

There are eight configurable options from the Preferences program:

DITHERING - (default: OFF) - Normally NewIcons does not dither the icon images. This generally results in better looking icons for AGA systems, since the icons appear exactly as they were created. For users with a limited number of available colors on the Workbench (such as those with <32 colors, you can try the DITHER option to improve the appearance of the icons.

Won't be available if you are using the High-End version of newicon.library.

NO BORDERS - (default: OFF) - Using this option removes the bounding icon boxes around your icons. This option MUST be enabled to be able to use the next one:

TRANSPARENT - (default: OFF) - This will remove all the gray area around the icons, dramatically changing the appearance of your WB. This option is recommended primarily for die-hard NewIcons users who have done away

with all those non-NewIcon images on their system. It can make non-NewIcon icons look strange at times, but can transform your Workbench into a visually stunning interface. Color zero is made completely transparent, so the backdrop will fill where the box was.

TRANSP. DRAG - (default: OFF) - This option will have the same visual affect as the NO BORDER and TRANSPARENT options but on icons that are being dragged.

RTG MODE - (default: OFF) - This feature has been added for the benefit of graphics card users only, and if you have a graphics card, you can set this mode to ON. RTG mode allows NewIcons to render the icon images in Fast RAM instead of Chip RAM. It has several advantages, including saving your precious chip RAM for other uses, and speeds up icon rendering, since access to fast RAM is, well, faster! :) If you don't have a graphics card, keep this mode OFF or you will meet the Guru, most likely.

NOTE: RTG mode is only available with the high-end version of the library. It also requires a proper graphic system that allows you to use FastrAM for the graphic data, such as Cybergraphics or Picasso96.

DEPTH PROMOTE-(default: NONE) - This option is really only useful when using the TRANSPARENT option. If you have non-NewIcon images that look particularly strange, you can try this option. Turning on this option causes all non-NewIcon images to be promoted to 8 bitplanes, and can sometimes fix those wierd looking non-NewIcon images. The drawback of using this mode is that it eats more memory. There are 3 modes available with this cyclegadget: NONE, which will not promote icons (default), ONLY WORKBENCH, which will promote only Workbench icons, and ALL, which will promote appicons and other images like dock icons that use icon images. Depth promote may look better or worse, depending on your OS/configuration. Play with the settings and find the best effect for you.

IMPORTANT! If you are doing some editing on non-NewIcons with an icon editor, be sure to turn the DEPTH PROMOTE option to NONE or ONLY WB. If you don't you'll be playing with an icon copy, and your editor won't save it properly. You have been warned.

TEXT OUTLINE - (default:None) - This option will allow you to change the way the text located under each icon is displayed. "None" will display the text as it's usually displayed, "Outline" will draw an outline around it, and "Shadow" will display it with a 3D look. The color

used to draw the outline/shadow is the one you've set as "Field" in your Font preferences.

NOTE: You must also set the icon text to "Text + Field" in your Font preferences for this to work.

PEN PRECISION-(default:16) - This option allows you to set how exact NewIcons will select the colors for displaying the icons. The default value is 16, which is reasonable if you have a limited number of pen colors available on your WB. It won't steal all the colors from your other programs, but may not display everything with exact precision. The lower the PRECISION parameter, the greater the precision when picking colors to map the icons. Zero is very high precision, -1 is the highest, and is recommended for graphics card users or those with lots of available WB pens.

The four other gadgets in the Preferences window are fairly self explanatory:

- SAVE - Saves the currently set configuration permanently.
- USE - Keeps the currently set configuration until the next reboot.
- CANCEL - Cancels any changed parameters.
- NewIcons Logo - Click here for version and author information.

NewIconsPrefs uses the ClassAct GUI system, and requires a few ClassAct gadgets to be installed on your system. ClassAct is NOT included in the NewIcons archive. It can be obtained from Aminet, or from the Finale Developments web site, located at <http://www.finale-dev.com>.

1.8 CopyNewIcon

CopyNewIcon will likely be the most used of the support utilities.

This command is used to copy the NewIcon image from one icon to another. You can use it to add NewIcon images to your other programs/files. ONLY the NewIcon image is copied. The standard image and the other properties of the destination icons are not affected.

You'll need this command to use the icons contained in the Icons/Misc drawer. DO NOT USE THE USUAL 'COPY' COMMAND TO COPY THOSE ICONS. You should consider the Icons drawer as a sort of "catalog" of images, which you can pick IMAGES, and not true icons, from.

From CLI:

```
Copynewicon srcicon desticon1 [desticon2...]
```

From Workbench:

```
click on CopyNewIcon
shift-click the icon you want to copy from
shift-click the icons you want to copy to
shift-double click the last icon
```

Or, double-click on CopyNewIcon to be prompted for a source and a target icon through file requesters.

You can also add this command to the Tools menu using ToolManager or ToolsPrefs.

1.9 CreateDefaultIcon

```
                This will convert a fake icon to a real icon. Of course it is ←
                much more
useful if
                DefIcons
                is running.
```

It also replaces real icons with standard icons, that is with the ones which would be used as fake icons if there wasn't a real icon. I hope I've been convoluted enough to confuse you ;-)

Anyway, the interesting part is that if you use this command on a real icon, the original properties will not be lost (position, drawer dimensions, tooltypes, default tool - unless the def_icon has a default tool by itself). This command is VERY handy to quickly replace all the drawer icons in your hard disk.

```
Usage is the same as
                KillNewIcon
                :
```

Click on CreateDefaultIcon, then shift-click on the icons you want to make into a real icon. Double-click on the last icon selected. If you have a standard default tool for certain files (like Multiview for your AmigaGuide® files), you can add that tooltype to your default icon (Envarc:sys/def_guide.info) and any AmigaGuide® files you use CreateDefaultIcon on will have the correct default tool associated with it. This technique is useful for picture and sound files, and ASCII text files as well.

1.10 InjectBrush

This command is used to create a NewIcon. It can be used only from CLI. It takes the name of the icon to NewIcon-ize and the names of two ILBM brushes. The brushes should be 36x40 in size, and use the standard palette. In the Extras/BUILDING_BLOCKS directory you find some standard elements to use in creating NewIcon images.

Template:

```
ICON/A,NORMALIMAGE/A,SELECTEDIMAGE,FORCE/S
```

```
Example: injectbrush myicon.info brush1.iff brush2.iff
```

You can leave off the ".info" extension if you like. Injectbrush is smart enough to know you mean the icon file, not the file itself.

Note that the original image is not removed from the icon. It stays there so that users not running NewIcons can see an old-style icon.

Color 0 of the brush used to create a NewIcon is always considered the transparent color. That is, it will always be remapped to color 0 of the screen.

The newicon.library does not optimize the color data. Thus, if you use an 8 bitplane brush to create an icon, and it uses only 16 colors of the 256 available, the icon created will be larger than if you used a 4 bitplane brush. The best thing to do is to always work on a 16 color screen, using the building_blocks/platform provided to start from.

Injectbrush expects the brushes to be of the standard size and color palette. You will be prompted if the brushes are not 36x40, or if the palette colors are non-standard. This is NOT implemented to force you to use a particular icon size or palette. Rather it is included as a feature for those who want to keep their icons the same palette and size, consistent with the included icon set.

This feature can be overridden by using the Force option.

1.11 KillNewIcon

This is the complementary of InjectBrush. It removes the NewIcon and restores the icon to its original image.

From CLI:

```
Killnewicon icon1 [icon2...]
```

From Workbench:

```
click on KillNewIcon
shift-click the icons you want to remove the NewIcon from
shift-double click the last of the icons
```

Or, double-click on KillNewIcon to be prompted for a target icon through a file requester.

You can also add this command to the Tools menu using ToolManager or ToolsPrefs.

1.12 PatchOpenWB

Kickstart 3.x has a serious bug concerning Workbench screen opening. If the Workbench screen is closed and a program causes it to be opened, either by calling `OpenWorkBench()`, or `OpenWindow()`, or `LockPubScreen()`, the system may hang. Specifically:

- If the program has priority less than 0, it will always hang.
- If the priority is greater than 0, it will never hang.
- If the priority is 0, it may or may not hang; it depends on when task switching happens.

This program is a workaround for the bug. It will intercept `OpenWorkBench()`, `OpenWindow()` and `LockPubScreen()` calls and temporarily raise the priority to 1, thus avoiding the deadlock. This patch is necessary when running `NewIcons`, otherwise the system will always hang when the Workbench screen is closed and reopened.

To install the patch, just execute the program `PatchOpenWb` in your startup-sequence, preferably before `IPrefs`, and after `SetPatch`.

Users of `MCP` and `MultiCX` can skip this installation as these programs have their own `OpenWB` patch (just make sure to enable it).

1.13 UpdateDrawers

This program will allow you to replace ALL drawer icons in a given path/partition (excluding this path's icon, i.e. specifying "`SYS:Prefs/`" will NOT change your `SYS:Prefs` icon), copying a specified `NewIcon` to them. This will include ALL sub-directories in the selected path. If, as an example, you specifying `SYS:`, then all drawers on that partition will be updated.

Note that if a drawer doesn't already have an icon (normal or `NewIcon`), `UpdateDrawers` won't affect it.

From CLI:

```
UpdateDrawers SOURCE DESTINATION
```

From Workbench:

```
click on UpdateDrawers
shift-click the source icon to use for the drawers
shift-click the target drawers/drives in which all sub-drawer icons
will be replaced.
shift-double click the last target icon.
```

Or, double-click on `UpdateDrawers` to be prompted for a source and a destination through file requesters.

1.14 ShowNI

This is a small CLI command that will take a list of specified NewIcons and display them in a window as buttons. Clicking on them will let you see the alternate (selected) imagery.

IMPORTANT: ShowNI requires ClassAct. More specifically:

```
images/drawlist.image
images/glyph.image
images/bevel.image
gadgets/layout.gadget
gadgets/button.gadget
window.class
```

Template: LABEL/K, TITLE/K, S=SELECT/S, ICONS/M/A

LABEL Optional string for the label shown while in Select mode. Default is "Select an icon:".

TITLE Optional string for the window titlebar. Default is "ShowNI".

SELECT Switch. If present, ShowNI exits at the first selected icon, returning its filename.

ICONS A list of icons to show in the window. A maximum of 15 icons are supported.

Here's a short example on how to use ShowNI to let the user choose between a list of icons, and copy the one the user chooses over a destination:

```
---
;This is the icon we're going to change.

copy SYS:prefs.info ram: QUIET

;We're going to copy the icon selected by the user in ShowNI over it.

copynewicon `ShowNi sys:prefs sys:tools sys:utilities SELECT` ram:prefs
---
```

This small script will copy an icon in your Ram Disk, and then ask the user to chose between three different images. The selected one will get copied over the icon present in the Ram Disk.

Feel free to include ShowNI with your Installer scripts if you want to use it. An example of use could be to have a drawer containing various drawer icons, and having the user choose which one he wishes to use. ShowNI can also be used to display colorful user queries.

The source code has been included as an example on how to use newicon.library to decode a NewIcon and remap it into a ready-to-use

Intuition Image.

1.15 Tips And Tricks

Here are some tips to help you make the most of the Newicons system:

Installation

Q: I just installed NewIcons version 4. Now my computer seems to freeze up during the boot process. What the hell?!

A: Hopefully we have placed this information in enough places in the archive that users won't have this problem. If you have updated from a previous release of NewIcons, and didn't originally install everything with the installer, you may have to make a change in the user-startup. Since V3, NewIcons requires a change in the command line that launches it. NewIcons MUST now be launched with

```
RUN <>NIL: c:NewIcons
```

If your old command line in user-startup still says

```
c:NewIcons
```

you must change it. If you installed NewIcons previously with the Amiga Installer program, the version 4 installation should take care of this automatically.

Q: When I ran the installer script, it didn't convert every single icon on my hard drive. Why not?

A: Since everyone's system can be configured differently, the installer can't possibly configure for all setups. The installer is designed to install the typical Workbench System file icons. To convert the rest of your icons, use the CopyNewIcon utility to copy images to other icons. You can quickly convert lots of drawer images using the UpdateDrawers utility.

Q: I really don't want a great looking Workbench... How can I deinstall it?

A: There is a script in the archive that will deinstall the package for you. To remove NewIcon images from icons by hand, just run KillNewIcon on them.

Q: The NewIconsPrefs editor won't start, complaining about a missing gadget/image. Where can I get this gadget?

A: The ClassAct gadgets can be found here:

```
http://www.finale-dev.com/download.html
```

Just get the latest archive from there, and install it on your system via its supplied Install script.

NOTE: ClassAction is NOT the same thing! Do not confuse them.

Q: I clicked on Install and got an error message from Installer saying it couldn't compile the script.

A: Your Installer version is outdated. Get V43.3 or better from Aminet.

Settings

Q: I'm using the borderblank option and some of my old non-NewIcon images look very strange. What's wrong?

A: Workbench 2.04 and up was not designed to have the borders removed around the icons. With the borderblank option turned on, color zero is now transparent. Any area of the icon with color zero will now show the backdrop through the icon. This can cause some icons to appear strange. The options for fixing it are 1) change your backdrop to a somewhat neutral gray color 2) don't use the borderblank option, or 3) get rid of those old icons altogether, and use NewIcon images instead.

Another note about the borderblank option: some non-newicon icons will really look bizarre, especially the secondary image if the alternate image is just a complemented color. You can try toggling the DEPTH PROMOTE option in NewIconsPrefs on or off. Depending on whether you use a graphics card, this option might help either way.

One further note about the borderblank option: If you use colorfonts for your icon font, these can look weird also with non-NewIcon icons. It appears to be a quirk in the OS. The solution is the same as above: 1) don't use borderblank, or 2) use only NewIcon images on the Workbench. Some icon authors also have forgotten to use color zero for the background image of their icons. The backdrop will look gray instead or transparent.. To fix these images they have to be converted back to brushes and have the background color changed to color zero.

Q: Can I use RTG mode if I don't have a graphics card?

A: No. You will end up with garbage displayed instead of icons, if not plainly crashing your system.

Q: If I have a graphics card, must I use the RTG mode?

A: You don't have to, but there's certainly no reason not to, unless you like eating up all your chip RAM, or your graphics card software doesn't support using FastRAM for image data (like the older Picasso-II software). Both CyberGraphX and Picasso96 support RTG mode.

Q: When I enable the RTG mode, I get garbage for my icons. I do have a 24-bit card, since I have a Picasso II. Why won't it work?

A: RTG Mode uses Fast RAM for allocating icons. You need to use a RTG software that properly patches graphics.library to allow blitting to FastRAM. The old Picasso-II native software didn't. Either use Cybergraphics, or the

new Picasso96 instead of your older software.

DefIcons

Q: Can I run Deficons without running NewIcons?

A: Absolutely, if you only want to use the default icon feature. In this case you will only have the standard icon images to view, however. NewIcons can be run without Deficons also, although you won't get default icon images for "fake" icons.

Q: If I want to make my own default disk icons, do I have to make them a disk type icon?

A: No. can use drawer type icons for disks or drawers in most cases. It will also use any type of icon for the "kick" type icons. Therefore, if you make new default icons for your ENVARC:sys drawer, you don't usually have to convert them to a disk or kick type icon. This applies to the custom icons only, such as def_RAMdisk, etc. The standard def_disk.info and def_kick.info icons still must be of the proper type.

Q: When I double click on an iconless file, sometimes the "Execute Command" gadget doesn't get filled in with the default tool like it's supposed to.

A: This is due to Intuition, not Deficons. If you open a drawer and click on an iconless AmigaGuide file for example, Multiview should be selected in the "Execute Command" field. However, if you open a second drawer or click on other icons in a different drawer than the first, then try to open that same AmigaGuide file, the default tool will not appear. Sorry, there is no easy solution to this problem; it's a fault of the OS. If you select Update Window from the Workbench menu and try again, all will be well.

Q: I'm running DefIcons, but my Ram Disk won't use the supplied def_RAMdisk.info icon. What's wrong?

A: Make sure your def_RAMdisk.info icon is of Disk or Drawer type. Also, check that you aren't copying a disk.info icon in your user-startup, overriding the default icon.

Creating and Editing NewIcons

Q: I used CreateDefaultIcon to convert my icons into NewIcons. Now my tooltypes and default tool settings are missing from the icons. Help!

A: Go back and read the documentation. If the default icon in Env:sys associated with the icon you created has a default tool setting in it, it will be used, as well as the tooltype settings. For general use in converting existing icons to NewIcons, use CopyNewIcon instead. CreateDefaultIcon is for adding icons to files that don't have .info files!

Q: I want to make my own icons that are different sizes and colors from the supplied icons. How can I get injectbrush to work with the brushes I use to make the images?

A: All the supplied icons use the same 31 color palette to preserve the number of available colors on the Workbench. If you choose, you are certainly not limited to these colors and you can make icons as complex as your heart desires. Icons also do not have to be the 36 x 40 pixel size either. Using the FORCE option in InjectBrush will allow you to use nonstandard palettes and sizes (up to 93 x 93 pixels). Make sure your brushes are the same size and palette, or injectbrush won't allow you to make them into icons.

Note: Although you can create up to 256 color icons, this is not really recommended, since you will very likely eat up all the available colors for the workbench, and programs like Multiview will not be able to dynamically allocate the correct colors for viewing pictures and so forth (unless you have a gfx card). Note that icons with large palettes take up a larger amount of disk space.

Q: IconEdit doesn't work with my NewIcon images. What gives?

A: Do NOT use standard icon editing tools to manipulate NewIcon images. Some programs could cause serious damage to the icon data, saving the NewIcon image as an old-style image, therefore losing palette information. The best way to create NewIcon images is to use a paint program; create a brush, and then use injectbrush; there are NewIcon editors out there as well, such as Iconian. Probably the best way to make NewIcon images is with a good paint program such as Personal Paint or Deluxe Paint. That's how we do it.

Q: I prefer the magic workbench style of icons. Can I get the benefits of using NewIcons with mwb?

A: Absolutely. Just convert the images to brushes and change them into NewIcons with InjectBrush. From now on you'll never have to worry about what your palette is set to. There is also a utility to convert mwb icons to NewIcons available on Aminet, called MWB2NI.

Q: With 2 images in each icon, my icon sizes are a bit bigger than standard icons. Is there anything I can do about this?

A: If you decide that you really like the NewIcons system, and make it a permanent part of your Workbench, you can make the standard icon images very small, i.e. a few pixels large. This is done quite easily with the included utility "Killoldicon", by Michael Barsoom. This changes the old icon images permanently, so don't do this procedure unless you are sure you won't need them back. To use, simply shift-click on killoldicon, then shift-click on the icons you wish to convert. While holding shift, double-click on the last icon to convert. Since NewIcon images are compressed, the resulting final icon can be considerably smaller than a standard icon with the equivalent number of colors. Also these icons load considerably faster with the old image removed. One drawback of this method, however is that if you share these icons with people not running NewIcons, they will only see the tiny pixel icon.

Q: How can I make my NewIcon icons into brushes for editing?

A: There's a utility on Aminet (gfx/conv/newicontobrush) which will do this

for you.

Q: Where can I get more icons in the isometric/McVey style?

A: There are loads of them on Aminet (well over 1000 images) in the pix/nicon directory. Try Roger McVey's NewIconsXTRA and NI-V2 packages (16 color only), and the AESicons series (16 and now 32 colors).

Q: Are there any backdrop sets for NewIcons?

A: Yes! Look in the pix/nicon directory on Aminet. Roger McVey has uploaded literally hundreds of NewIcons compatible backdrops. They look great!

Q: How do I use the DEFAULTIMAGE tooltip?

A: Use it with icons that have NO NewIcon image data in them. For example, to make all your drawers use the image of an icon called Work:MyDrawer.info, add this tooltip to your non-NewIconized drawers:

```
DEFAULTIMAGE=Work:MyDrawer.info
```

Now, you can change the look of ALL your system drawers just by changing the MyDrawer icon's image. Pretty cool! You could even set up your user-startup to randomly copy a different image at each bootup to Env:sys, and make your Workbench look different every time you start your computer!

Using this feature slows down loading time slightly, because two icons have to be loaded (the main one and the one whose image is to be borrowed), but has several advantages, especially with icons which are used frequently, like drawer icons:

- o 1) It saves disk space, because the image is stored only once.
- o 2) Saves memory when the icons are on screen, because the Tooltypes containing the compressed image are loaded, decompressed, and freed, whereas in the normal method they stay in memory until the icon is freed.
- o 3) If you get bored of an image, just replace the default one, and all the others will change automatically!

Remember that the image stored in the icon has precedence. Therefore, you have to run KillNewIcon on a NewIcon to have the DEFAULTIMAGE ToolType work.

WARNING!!!: This trick is recursive, that is, if DEFAULTIMAGE points to an icon which has a DEFAULTIMAGE itself, a third icon will be loaded, and so on. Pay attention not to create loops which refer back to another icon. If you do, you will likely meet the big GURU. You have been warned.

Miscellaneous

Q: Sometimes when I change screenmodes, the colors of my NewIcon appicons don't look right. What's going on?

A: Intuition doesn't reset the colors properly for appicons, unfortunately. You'll have to switch back to the screenmode you were using, or restart the program that created the appicon. Sorry, this is a fault of the OS. For Toolmanager 2.1 Docks, this shouldn't be a problem with NewIcons version 4. A fix for Toolmanager docks only has been implemented. This might break with future versions of Toolmanager.

Q: My icon colors don't look the right colors.

A: The NewIcons system dynamically allocates colors as needed when an icon appears on the Workbench. Most likely if you are having a problem with the icon colors, you don't have enough free colors available on your Workbench screen. Unless you are using a back-drop with a large number of colors, 32 or 64 colors is usually plenty with the supplied icons. Increasing the number of bitplanes in the screenmode settings should fix this problem. Alternatively, you can set the DITHER mode in NewIconsPrefs to ON (Note that dithering does not work with the high-end library installed).

You can check how many free pens you have currently by pressing LAlt-shift-N while the NewIcons program is running.

Q: I'm running DOpus 5.5, and my NewIcon icons keeps changing colors! What's wrong?

A: The NewIcons executable is meant to be a patch for the original Workbench, to allow it to use NewIcon icons. Unfortunately, it won't properly work with DOpus 5.5, unless you ran DOpus on the Workbench screen. Future Workbench replacements should directly support NewIcons via the newicon.library to avoid such problems. (Note: DOpus Magellan supports NewIcons fully).

Q: I'm running ToolManager 3, and my docks using NewIcon icons won't appear. What can I do?

A: ToolManager 3 relies on datatypes to display its dock imageries. Either convert your NewIcons into IFF brushes, or install a NewIcon datatype. Or, go back to ToolManager 2.1 (which is what we, Eric and Phil, did :)

Q: If I change the Workbench screenmode, all the displayed icons that use the DEFAULTIMAGE tooltype show weird colors.

A: This is a bug in the DEFAULTIMAGE handling. With prior versions, the system would plainly crash. The only way we could "fix" it was to simply not attempt to reload and remap these again.

Q: Sometimes at boot time, some of my disk icons don't show as NewIcons.

A: Your Workbench must be loading too fast - NewIcons isn't fully installed yet that Workbench is already loading and displaying disk icons :) Either move the line running NewIcons higher in your Startup-Sequence, or add this line right below the line running NewIcons: C:Wait 1

Feature Requests

Q: How about adding built-in pictograms?

A: This was originally planned for 4.5. In fact, the code was even written, but then problems caused by the NI design itself (NI was never meant to be extended) appeared, forcing us to drop the idea for now.

Q: How about eliminating the 93x93 size limitation?

A: This is another problem inherited from the original design. Increasing this limitation would cause various problems, most serious one being that people still running the older versions of the software would crash their system as they would run into one of those larger icons. Anyway, keep in mind that these are file ICONS. If you need to put a 320x200 screenshot of your program, please put it elsewhere than in a Workbench drawer window.

Q: How about (insert your favorite feature or bug fix here) for DefIcons?

A: DefIcons is no longer being supported or developed. It's included in the NewIcons archive purely because it's simpler than making a separate release (and having to release the default icons in both archives). If someone updates the deficons brainfile, send it to us and we'll stick it on the website (the source is in the NewIcons archive).

Q: I need an icon for (insert your favorite program name here).

A: Phil often fulfills such requests if you ask nicely. Drop him an Email, and he'll see what he can do.

Q: How about: animated icons? 16-bit or 24-bit icons?

A: Such things cannot be implemented just by patching icon.library. They require a Workbench rewrite. So implementing these is quite simply impossible for us.

Q: I'd like to be able to use a "theme concept" kinda like what Win95 allows (being able to switch between different iconsets in an easy way).

A: There is various ways to do it. The best one IMHO is to use the DEFAULTIMAGE tooltype in your icons, loading icons from a THEME: assign. Put your theme icons in separate directories, then create a THEME: assign in your user-startup, pointing to the directory containing the desired icons. Just moving the THEME: assign will let you use a whole different iconset, as DEFAULTIMAGE will always obtain the icons from the currently set assign.

1.16 Color Scheme used with the supplied iconset

NewIcons comes with a standard set of icons. Note that you are by no means limited to this iconset.

The included iconset uses a 32-color palette:

R	G	B
---	---	---

-00:	165	168	168	Transgray
-01:	045	040	158	DeepBlue
-02:	000	102	153	DkBlue
-03:	051	119	170	MedBlue
-04:	102	136	187	LtBlue
-05:	012	097	255	SkyBlue
-06:	036	093	036	DkGreen
-07:	051	136	051	Green
-08:	134	178	061	LtGreen
-09:	012	227	000	LCDGreen
-10:	077	158	142	Aqua
-11:	130	000	000	DkRed
-12:	221	051	051	Red
-13:	219	101	057	DkOrange
-14:	219	142	065	LtOrange
-15:	221	187	068	Yellow
-16:	239	231	020	LCDYellow
-17:	130	097	077	Brown
-18:	204	153	119	Tan
-19:	255	166	130	Peach
-20:	153	085	170	Purple
-21:	178	008	121	Maroon
-22:	255	000	255	FunkyPurp
-23:	255	255	255	White
-24:	204	204	204	Gray1
-25:	187	187	187	Gray2
-26:	170	170	170	Gray3
-27:	136	136	136	Gray4
-28:	102	102	102	Gray5
-29:	077	077	077	Gray6
-30:	051	051	051	Gray7
-31:	000	000	000	Black

Icon Platform Color Scheme:

The included iconset is drawn in an isometric style, with platforms which can aid the user in identifying file types quickly by the color of the platform.

The platform colors are divided primarily by file type function. If you plan to make your own icons in the style similar to these, consider using the color scheme below, to help maintain consistency with other icons. Of course you are not bound to this color scheme, or icon style for that matter. The platforms are indexed in the following manner:

System files (including Locale)	Red
DEVS files (including disk icons) \$^1\$	Tan
Prefs programs	DkOrange
Graphics and Gfx programs	Aqua
Music/Sound files and programs	LtBlue
Networking programs	Yellow/Tan dither
Text programs	Purple/Tan dither
Games	Purple
Local Network (Parnet, etc)	Maroon
Emulation programs	Red (like System)

All other Tools, Projects

Green

$\1 Disk icons are commonly platformless as well.

While Green is mainly for most Tools and Projects, feel free to use other colors in the palette, particularly for groups of programs that carry a similar theme (e.g. a group of files that comes with a package). These guidelines are fairly loose...the important thing is to be creative and be consistent (and don't forget, color zero is the transparent one). Now get going and make more NewIcons! :)

1.17 Deflcons Brainfile

The tree setup is important to understand how deficons works. If a file has failed to be matched its PARENT filetype will be used. So, if you wished to use the same icon & default tool for a filetype as the one used for its parent, you could simply omit it.

The icon name to be searched will be def_#?.info, #? being the name listed in the tree.

Brainfile's tree:

```
Project
|_ ascii
  |_ doc files matching #?(.doc|.txt|readme#?)
  |_ html
  |_ amigaguide (bin)
  |_ usenet news
  |_ email
  | |
  | |_inemail incoming email
  | |_outemail outgoing email
  |
  |_ postscript
  |_ fd
  |_ mountlist
  |_ filearchive
  | |
  | |_uuencode
  |
  |_ installer
  |_ src
  | |
  | |_ c
  | |_ cpp
  | |_ h
  | |_ i
  | |_ asm
  | |_ basic #?.bas
  | |_ e amigae code
  | |_ m amigae module
  |
```

```
|_ tex
|_ make a compiler's makefile
|_ rexx
|_ script amigados script
```

Tool

```
|_ device
|_ library
|_ keymap
|_ font
|_ printer
|_ sfx lha executable
|_ filesystem
|_ handler
|_ loadmodule
```

iff

```
|_ sound
|   |_ 8svx 8svx,made to have sound as parent object
|
|_ video
|   |_ anim anim,made to have video as parent object
|
|_ music
|   |_ smus sums,made to have music as parent object
|
|_ fax
|   |_ faxx these are common gpfx files
|   |_ fax3
|
|_ ilbm
|   |_ brush iff grab(brush),made to have ilbm as parent object
|
|_kickstart iff kickstart
```

empty

object compiled .obj

lib compiled.lib

archive

```
|_ lharc
|_ lzx
|_ arj
|_ zip
```

diskarchive

```
|_ dms
```

picture

```
|_ gif
|_ jpeg
|_ png
|_ bmp
|_ pcx
|_ targa
```

```
    |_ tiff

video
  |_ quicktime
  |_ flic
  |_ mpeg
  |_ avi

sound
  |_ wav
  |_ mp2
  |_ sunau
  |_ voc

music
  |_ med
  |_ mod  general #?.mod|mod.#?|#?.mod.#? pattern
  |_ s3m  screamtracker
  |_ xm   Extended Module (FastTracker)

kickstart grabbed kick image

src
  |_ basic
    |_ amos amos basic,made to have proper parent objects

font

outlinefont

PREF this is the oldie system-configuration file.

key  pattern for keyfiles

ascii these are incase our TXT is crunched
  |_ doc
  |_ amigaguide

filearchive
  |_ powerpacker
  |_ imploder
  |_ xpk
  | |_ tool tools packed with xpk will get correct icons
  | |_ iff for crunched iff's
  | | |_ sound
  | | | |_ 8svx
  | | |
  | | |_ video
  | | | |_ anim
  | | |
  | | |_ music
  | | | |_ smus
  | | |
  | | |_ ilbm
  | | | |_ grab
  | |
  | |_ sound for crunched sounds.
  | | |_ wav
```

```
|      |_ sunau
|      |_ voc
|
|_ rnc
|_ compress
|_ gzip
```

1.18 Authors

Team NewIcons
=====

The NewIcons project is the result of many months of idea-exchanging, late-night IRC sessions, numerous reboots and collaboration among people who have never once met each other in person. The NewIcons package was created purely for the enjoyment of its users.

Official Web site: <http://www.amiganet.org/NewIcons/>

Please send any bug reports to Eric or Phil.

Programming for Newicons V3 and up:

Eric Sauvageau
3535 Masson street
Apprt. #2
Montreal, QC
Canada
H1X-1S1

IRC: RMerlin
E-mail: merlin@linux.tc3net.com

Documentation and installer, NewIcons V1.0 - 4.6, and artwork for Newicons V3 and up:

Phil Vedovatti
4129 Ashworth Ave. N.
Seattle, Washington 98103
USA

IRC: Aes
E-mail: vedovatt@u.washington.edu

The deficons.prefs file:

Ariel Magnum
IRC: Ariel
E-mail: arielmag@actcom.co.il

The original icon (V1) artwork was done by

Roger McVey

Original concept, and programming for versions 1.0 - 2.0:

Nicola Salmoria
Via Piemonte 11
53100 Siena ITALY

E-Mail: MC6489@mclink.it

1.19 Acknowledgements

Acknowledgements

=====

Many thanks go out to our beta testers, who risked crashing their systems for the sake of NewIcons:

Steve Bara (IRC: Frotz)
Simon Dick (IRC: SimD)
Jonathan Larson (IRC: Multitask)
Giacomo Magnini (IRC: Swatch)
Petter Nilsen (IRC: Mitchman)
Andrew G. Robson (IRC: Greebo)
Chris Townsend (IRC: Vidar)
Joshua B. Wingell (IRC: Shape)

Thanks to Chris Green, whose chunky to planar routine was based upon. (Chris based his routine on one by Richard Addison).

Thanks to Federico Giannici who pushed Nicola to make the color remapping faster, and faster, and faster...

Thanks also to Michael Barsoom, who kindly allowed the inclusion of his program, "killoldicon" in the package.

Børge Nøst for optimizing the ASM DecodeLine routine.

Christopher Aldi, Tim Aston, Osma Ahvenlampi and Petter Nilsen for ClassAct.

Georg Steger and Christopher Naas for their help and routines for the transparent icon dragging.

Daniel Nyberg for the original icon text outline routine.

All the folks on ANet/EFNet/IRCnet IRC channel #amiga who begged us to finish this release.

And thanks to all the NewIcons fans out there!

1.20 Distribution

This software is FreeWare. It is freely distributable as long as the archive remains intact, and only a nominal fee is charged for its distribution.

This software is provided "AS IS" without warranty of any kind, either expressed or implied. By using it, you agree to accept the entire risk as to the quality and performance of the program.

1.21 History

Version History:

V1.0:

Original public release.

V2.0:

In the previous version of the docs I forgot to mention that the chunky to planar routine was based on a routine by Chris Green (who based his routine on one by Richard Addison).

fixed bug in DefIcons, was causing "please insert volume dir c" after doing execute command - dir c:

fixed bug in DefIcons, would hang if it couldn't find deficons.prefs

fixed bug in DefIcons, was causing a disk icon to appear for files called "disk".

fixed bug in DefIcons, would show a drawer as disk icon for disks containing a directory called "disk" (which is a bad idea anyway).

The GetDiskObjectNew() patch now returns NULL if the given name doesn't exist (like the standard function)

now does dithering (optional)

some file types added to deficons.prefs.

bug fix: createdefaulticon now preserves the tooltypes of tool icons and of project icons whose default icon have no default tool.

bug fix: newicons would check the argarray after FreeArgs(). Thanks to Nicola Soggia for discovering it and not reporting it. Thanks to Moreno Magnani who did report it.

bugfix, arj files were not recognized

bug fix: DefIcons now uses Stricmp() to avoid problems with international sorting

DEFAULTIMAGE special ToolType added

Fixed installer: no longer copies over the original def_#? icons in Envarc:sys. It now just copies them to a safe place.

Localized installer and documentation files added

Replaced "ReplaceImage" utility with "KillOldIcon", since ReplaceImage required NewIcons not to be running to use.

Changed a few of the icons from the original distribution archive

CHANGES since last version from Nicola (Release 2, V38.3)

3.0

===

- Development is now done by
 Phil Vedovatti
 and
 Eric Sauvageau
 .

- A much requested feature: you can now have borderless icons on your Workbench, just like under Kickstart 1.x :) Note that it is really a hack, so it might break under future versions of Workbench, and it has some minor drawbacks. Make sure you read the
 NewIconsPrefs
 section
about it before enabling it.

- New RTG Mode which uses Fast RAM instead of Chip on systems with graphics cards.

- Library (V39.1) now comes in two versions:
 - o 68020 and up, OS 3.x (Required for RTG Mode)
 - o 68000, OS 2.x

The library base has also changed, so application programmers should check the updated libraries/newicon.h include file. Note that things are still fully backward compatible).

Various speed optimisations have been done, and the tootype decoding routine has been rewritten by Børge Nøst. We're hoping to also have the C2P routine rewritten in a future version :)

- NewIcons has been rewritten in E, and is now a commodity which can be enabled/disabled. The hotkey will let you see some information about the current settings, and allow you to directly launch the external
 preferences
 editor.

- NewIcons will no longer auto-detach from the CLI so it must be Run <>NIL:. It also has a totally new template:

```
'CX_PRIORITY/N/K,CX_POPKEY/K,PE=PREFSEEDITOR/K'
```

Hopefully NewIcons will auto-detach again in a future version.

- The former Newicons settings have been moved to a separate prefs editor. Makes it easier for the user to install/configure, and gives you more freedom to experiment with the various settings without needing to reboot.

This editor uses the ClassAct BOOPSI package for its GUI (all the required classes are included in this archive.)

- Added a "kludge" so that icons used by ToolManager docks get properly remapped on screenmode changes.

3.1

===

- Fixed a bug in the PutIcon() patch that caused various problems with some programs such as SwazInfo or DOpus.
- Added Autodocs for newicon.library, allowing third party developers to directly support NewIcons through the library.

4.1

===

- New official WWW support site: <http://www.amiganet.org/NewIcons/>
 - Improved installation procedure.
 - Extended the standard 14-colors palette to 32-colors.
 - Complete new 32-colors iconset included. People who prefer the older ones for speed or resources issues can still use them.
 - Library (V40.1) now comes in three flavours:
 - \textdegree{} Low-End, for 68000 and OS 2.04 users and up.
 - \textdegree{} Middle-End, for 68020 and 3.0 users and up.
 - \textdegree{} High-End, for 68020 and 3.0 users and up, without the dither option and with RTG Mode support.
 - New library call: GetDefNewDiskObject(objtype).
 - Preferences are now loaded in the library init code instead of in the NewIcons executable, allowing easier use of the library without the main executable.
-

- Made Transparent and NoBorder options separate.
- Added Transparent Dragging option.
- Added Outlined/Shadowed iconcontext option.
- Rewrote the NoBorder/Transparent patches, made them cleaner.
- Changed the preferences file format. The new format is now documented in the Includes.
- The preferences editor is now able to detect what features are available in the installed library, and will display the library version in the About requester.
- Added ShowNI, a CLI-based NewIcons viewer/selector.
- Added UpdateDrawers, a new tool that will allow you to recursively replace all drawer icons within a given path.
- All Workbench-based tools will allow you to select the arguments through filerequesters if launched from Workbench without any argument.
- CopyNewIcon will report if it can't write the destination icon.
- Updated InjectBrush to support the new 32-colors palette.
- Updated the DefIcons brainfile, added various new filetypes.
- DefIcons now uses utility.library's Stricmp() instead of SAS/C's own routine. Should fix the problems experimented by some persons using a localized Workbench.
- A lot of code optimization, resulting in general speed improvements.
- Recompiled the libraries and the utilities with SAS/C 6.57, and optimized the 020+ libraries for 68060 users.

4.5

===

NewIcons:

- This version prevents crashes occurring if the user would change screenmode while an icon with DEFAULTIMAGE was displayed. Sadly I can't solve the problem itself, as it's a side-effect of the very idea of DEFAULTIMAGE (it's been there since Nicola's days). I did the best I could do, which was to at least prevent it from crashing (I simply no longer remap those icons).
 - Eliminated the crawling ants artifacts
 - Now requires Kickstart 3.0. 2.04 users: this is your wake-up call.
-

- Various changes to the No Border and Text Outline patches, speeding them up a bit.
- An Enforcer would happen if an icon was loaded while no screen was open (i.e. before WB and any bootpic screen would open). This one should be fixed now.

NewIconsPrefs:

- Bumped from 40.3 to 40.5
- Updated About to reflect V4.5

CreateDefaultIcon:

- removed a debugging KPrintf() I had forgotten.

4.6

===

NewIcons:

- eliminated Cyberguard/Enforcer hits.

NewIconsPrefs:

- fixed a small recoverable alert which could occur if certain Class Act gadgets were missing.

InjectBrush:

- The wrong version was supplied with NewIcons 4.5, which contained some beta code. Use the version supplied here instead.
-