

Picticon

COLLABORATORS

	TITLE : Picticon		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		December 7, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Picticon	1
1.1	Main	1
1.2	Copying and Disclaimers	2
1.3	Introduction	2
1.4	Requirements	4
1.5	Usage	4
1.6	Tooltype Summary	4
1.7	Switches and Modes	9
1.8	A summary of Picticon's logic and behavior	9
1.9	Known Bugs	10
1.10	Stuff to do	11
1.11	Q&A	11
1.12	Author Info	12
1.13	Iconian	13
1.14	Registering a complaint	14
1.15	Newicons	14
1.16	mwb2ni.doc	15
1.17	Hiya!	17

Chapter 1

Picticon

1.1 Main

Picticon V1.3
May 24, 1995

This document describes and explains Picticon V1.3, a picture to icon convertor that requires OS3.x. Datatypes, palette sharing, and appicons are a few of the features.

All binaries, executables, source and documentation is Copyright ©1993-95 by Chad Randall. All Rights Reserved. Please read the section "Copying~and~Disclaimers" for more information.

Copying~and~Disclaimers~ Some legal information.

Introduction~ A brief introduction.

Requirements~ What you will need in order to run.

Usage~ How to utilize the various features.

Tooltype~Summary~ A complete listing of tooltypes and args.

Logic tree A summary of how Picticon acts.

Switch Modes A summary of interfering modes.

Known~Bugs~ A list of known bugs, and a to-do list.

Questions~and~Answers~ A small list of Q&A.

Notes on Newicons Some ideas, hints, and solutions.

Encountering a Bug What to do when you find one.

Credits~and~Author~Info~ About me, and some thanks.

This guide has not be edited much since version 1.0. Please refer to "history.txt" and "Picticon1_3.readme" files for new features and bug fixes.

1.2 Copying and Disclaimers

Copying and Disclaimers

Picticon was thrown together in a few days using bits and pieces of Iconian. If you haven't gotten a copy of Iconian, find one. It is the number one OS3.x icon editor you can get (IMHO).

Since Picticon is a patchwork program that's held together by spaghetti sauce, many bugs will probably show up. I will honestly claim that I can get consistent, satisfactory results at this time. If you do run across any, please let me know.
SEE: Bugs Registering a complaint

Everything that this archive contains that was created by me is copyrighted. You may not use this copyrighted material in your own programs, archives, or creations without my express permission.

Although this archive and it's contents are copyrighted, you may freely copy them if no charges are incurred. A nominal fee may be charged for media costs, plus 50 cents (US) for labor. This means a disk copy, so the entire disk may not be sold for more than the cost of the disk plus 50 cents (US). Of course, shipping is separate. If you receive a disk that you feel you paid too much for, please spread the word. I personally don't use PD houses, but many do. So warn your fellow Amigans about excessive prices.

There are several exceptions to this. Fred Fish may include this program, of course. He is the one spot any Amigan should be able to look to for quality material.

Aminet also has free distribution rights. This is another high-quality distribution network that should not be hampered in any way.

Any CD collection can include Picticon for free. CD's are a great source for programs. Publishers of CD's rarely (never?) take seem to take advantage of end users. (Unless they were to *only* include Picticon on a CD and then charge 19.95... yea right)

Magazines wishing to include Picticon on a coverdisk may do so on one condition. I'd like a copy of the specific issue mailed to me. My address is at the end of this guide. 10-30\$ isn't alot considering all the issues a magazine sells. Is it? (A big thanks to Amiga User International for asking for Iconian in the past.)

The source code "doloaddt.m" may be included into your E programs. Special copyrights and conditions that apply to these works appear in "doloaddt.readme".

1.3 Introduction

Introduction

Picticon is a conversion program that creates "thumbnail" icons from pictures. This is accomplished through datatypes, and therefore many various formats are supported.

Also, palette sharing routines are used to remap these scaled icons to the current Workbench's palette.

Unlike previous similar programs, many options are available to customize what these icons look like. These include background images, size text, centering, and sizes from real small to extraordinary large. SEE: Tooltype~Summary

A graphical status window is opened if run from Workbench or a Appicon. This status window will be opened and closed many times, but it's position is remembered until you turn the computer off. If you wish to set a permanent default position, then copy the file "Picticon.prefs" from ENV: to ENVARC:.

All visual information may be turned off by the QUIET~tooltypeQUIET~tooltype. This includes the status window and/or CLI output. (Note, the CLI QUIET switch isn't operational, so use the >NIL: pipe for now.)

New features of version 0.90 include aspect scaling, fractional scaling, faster rendering, HAM support, and a complete CLI interface.

New features of version 0.95 include newicon support and multiple CLI filenames and/or match-patterns.

New features of version 0.96 include aborting multiple picture conversions and disabling the AppItem.

New features of version 0.97 include greyscaling.

New features of version 0.98 include "addicon" and "overwrite" tooltypes. Plus the ability to convert an entire directory by dropping a drawer icon onto Picticon's appicon.

New features of version 0.99 include many HAM rendering related options.

There are three ways to run Picticon.

CLI: Recommended usage would be from a script. A sample script is included.

Workbench: Seleting multiple icons, then double-clicking Picticon's icon will invoke conversion of the selected pictures.

Appicon: If run from Workbench without passed files, an Appicon will be opened. Any "dropped" files will be processed. To remove Picticon, just doubleclick the appicon and select "Quit".

1.4 Requirements

Requirements

Any Amiga running at least Kickstart 3.0 and Workbench 3.0.

At least 1 megabyte of RAM is required. 2 megs of CHIP and 4 megs of FAST is recommended.

A 68000, but a faster processor is recommended.

The external librarys "iffparse", "diskfont", and "mathtrans" of at least version 37. All these are standard, so no problems should arise.

A strong knowledge of palettes, colors, and tooltypes/cli is recommended for full feature utilization.

A nice set of pictures. Come on, at least one. :)

1.5 Usage

Usage

Just double click the icon for an instant result. Then drop a picture icon onto the newly created appicon. If you need the picture to show up on Workbench, use the "Window-> Show-> All Files" Workbench menu option.

Read the section "Tooltype~Summary" for some better info on creating custom backgrounds, sizes, and overall looks.

The CLI accepts multiple files and/or match-patterns. Example:

```
picticon pictures:#?.gif store:gifs/#?.gif ram:(#?.iff|#?.ilbm|#?.pic)
```

1.6 Tooltype Summary

Tooltype Summary

x a numerical value
B a boolean value (only YES or NO, TRUE or FALSE are accepted)
\$ a string, usually a file with directory path
() What the option defaults to

ADDICON=B (NO)

If Picticon or the datatype library fails to create a scaled

image, an icon is still created if none exists. If WhatIs is present, then a user defined template is used, otherwise the system default project icon. You may force the creation of an icon by using the OVERWRITE tooltype.

```
APPICON=$ ('ENV:SYS/def_appicon') then,  
  (program's icon) then,  
  ('ENV:SYS/def_tool') then,  
  (default tool icon supplied by Workbench)
```

Names a user defined icon for the appicon image. X and Y positions are set to NO_ICON_POSITION unless overridden by APP_X_POS and/or APP_Y_POS.

```
APP_X_POS=x (NO_ICON_POSITION)  
APP_Y_POS=x (NO_ICON_POSITION)
```

Specifies where the appicon will appear, in typical unreliable Workbench fashion.

```
ASPECT_X=x (1)  
ASPECT_Y=y (1)
```

Defines your screen aspect ratios. If you are in hires, non-interlaced NTSC, then you could specify "ASPECT_X=2" and "ASPECT_Y=1". This states that there are 2 horizontal pixels to every 1 vertical.

```
BACKGROUND_ICON=$ (none)
```

Names a background you wish to overlay the picture over. There is no default. If the named icon does not exist, a blank workspace will be used.

```
CENTER=B (NO)
```

Centers the image in the PIC__POS/PIC__SIZE rectangle. Otherwise, it'll be upper-left justified.

```
CHUNKYMODE=B (NO)
```

Turns on chunky image conversion. This may allow certain types of graphic boards to save correct looking icons. The conversion process is a bit slower, but much safer.

```
DISCARDERROR=B (NO)
```

When active, and used in conjunction with RENDERHAMx, then any ham-errors are discarded. This results in less dithering noise, but slightly more fringing. This option was meant to deal with HAM6's

4-bit palette limitations. Experiment.

DITHER=B (YES)

Turns error-diffusion dithering on or off.

FREE_ICON_POS=B (NO)

If on, all saved icons will be "unsnapshot". If off, then the template icon's position will be used.

FORCE_EIGHT=B (NO)

Saves the icons with 8 planes, always. This will only help with icons created with 3 planes (8 colors), and a MagicWB style palette setup. Hopefully, everyone has switched to "shovecolors" by now, and won't need this option... :)

FULLHAMBASE=B (NO)

When this option is turned on (yes/true), then all available pens are used when selecting base colors. On default, only the first 4 pens are used. When left off, the resulting icons will appear closer to correct when displayed on other systems.

GREYSCALE=B (NO)

Enables the greyscale creation of icons. Best used with the NEWICON switch, but it also works with normal mode. The grey level is calculated using equal parts. You may wish to try the LUMSCALE switch on certain pictures to determine which you prefer.

The purpose of this switch is to create thumbnails that require fewer pens when using Newicons. Even when combined with the QUANTIZE tooltype, the icon files will remain at almost the same size.

HAMTHRESHOLD=x (64)

Alters the HAM thresholding formula. The nearer this value is to zero (0), the less base-colors are used, meaning more fringing. When this value approaches 760, the more base-colors are used, meaning less fringing.

HIGHPEN=x (256)

Specifies the highest pen to use. A value of 4 or 8 is useful for cross-computer displaying.

LUMSCALE=B (NO)

Similar to the GREYSCALE switch, but a luminance formula is applied to the red, green, and blue registers.

LOWPRI=B (NO)

Turn this on to allow Picticon to run at -1 priority. Useful for background processing of many images.

MAXIWIDTH=x (128)
MAXIHEIGHT=x (100)

This is the workspace you wish to allocate to the processing engine. It ranges from 50 to 1000 in size. If you specify too small of a number, your icons may come out chopped off. The default values Usually work fine.

NEWICON=B (NO)

Saves newicons. Original image is replaced with a "dot" image. The maximum size is reduced to 92x92 pixels and is enforced. Any background templates are ignored, as these are not color-mapped yet. See: Newicons

NOAPPITEM=B (NO)

Disables the creation of an AppItem (Workbench's Tools menu).

PIC_X_POS=x (0)
PIC_Y_POS=x (0)
PIC_X_SIZE=x (size of buffer (128))
PIC_Y_SIZE=x (size of buffer (100))

Creates a rectangle to place the scaled image into. This can be used along with BACKGROUND_ICON to create frames. See my examples. Also, see the CENTER tooltype.

OVERWRITE=B (NO)

When used in conjunction with the ADDICON tooltype, this tells Picticon it is okay to overwrite any icon (non-picture) it stumbles upon. Use with care.

QUANTIZE=N (256)

Defines how many greyscales to use when greyscale or luminance modes are activated. If the image has fewer than your specified value, then it will be used. Meaning, if you specify 8 shades of grey, and the picture only has 4 colors, then 4 shades of grey will be used for that picture, only.

QUIET=B (NO)

Turns off all status output, including messages and the status window. Not tested much, so take care.

RENDERHAM6/RENDERHAM8=B (NO)

Enables HAM-rendering mode for icons. Use in conjunction with a HAM6/HAM8 Workbench. You can find out how to hack a HAM Workbench by downloading "WBHacksAGA.lha" in the "gfx/aga" directory of Aminet.

SHOWSIZE_NORMAL=B (NO) This overrides: SHOWSIZE_OUTLINE=B (NO)

The size text can be 1 of three styles. It defaults to a shadow look, but a normal font or outline style can be specified. NORMAL overrides OUTLINE. Use OUTLINE if you expect the text to be displayed over a confusing background.

SHOWSIZE_TALL=B (NO)

Tells the show size renderer to use the 8 pixel high font instead of the 6 pixel high. Useful for interlaced displays.

SHOWSIZE_X=x (no display)

SHOWSIZE_Y=y (no display)

Will display the image size using one of two built in fonts. These coordinates specify the upper-left position, not the baseline. The height is 6 unless SHOWSIZE_TALL is turned on, in which case it's 8. Use a X coordinate of -1 to center the display under the scaled image.

STRETCH=B (NO)

Will reduce the picture to fit exactly into the dimensions specified with MAXIWIDTH and MAXIHEIGHT.

This tooltype is a bit misleading, as it will not enlarge images to fit the specified buffer. If both dimensions of the picture fall below the buffer dimensions, then no scaling is done at all.

All aspect information is ignored (for obvious reasons).

```
TEMPLATE_ICON=$ ('ENV:sys/def_picture') then,  
                ('ENV:sys/def_project') then,  
                (default project icon supplied by Workbench)
```

Names a icon to get tooltypes and positions from. Only the "meat" is used, not the image. To specify an image, see the tooltype BACKGROUND_ICON.

1.7 Switches and Modes

Mode and switch incompatibilities

As the number of options grew inside Picticon, the problems arising between conflicting rendering methods also grows. Listed here are several combinations you will want to avoid.

Greyscale and quantization

These two were written to support the NewIcon mode. But, they will also work during "normal" rendering with limited effects.

Greyscale and Lumscale will try and pick out greys from the Workbench palette, but will not limit itself to them. This in effect degrades image quality, but rarely reduces the icon file size. It is not advised to use these switches without the NewIcon switch.

Without the Greyscale and NewIcon switches, the Highpen parameter will be lowered to the Quantize parameter. This may go undetected, so beware.

Dithering while using NewIcons

Dithering will only slow down the NewIcon rendering, unless Greyscale is activated. Activating the Greyscale, Dither, and NewIcon switches may improve the image quality. Of course, the Remap switch is ignored altogether when rendering NewIcons.

Backgrounds vs. NewIcons

Background images should be ignored when using the NewIcon switch. If this was enabled, the background would be remapped twice resulting in very undesirable results. Besides, with a 92x92 image, there isn't much room left over.

1.8 A summary of Picticon's logic and behavior

What does Picticon do with icons, pictures, templates and switches?

As the development of Picticon progresses, I've started losing track of what Picticon does **exactly**. So included here is my memory of what Picticon should do. But be warned that it may not be 100% accurate.

We'll start with the routine "processicon". This is where the meat of the icon handling is done.

1. Allocate bitmaps. No rendering will be done outside the user-defined dimensions. Therefore, this is the maximum size the final icon can be.
2. Attempt to get a "template" icon. This will be used for tooltypes, default tools, position, etc.
 - 2a. Use template name if supplied.
 - 2b. Use any exsisting icon for the file.
 - 2c. Use a WhatIs defined icon, if WhatIs is present.
 - 2d. Attempt to open ENV:SYS/DEF_PICTURE.
 - 2e. Ask icon.library for the system default project icon.
(If any of the above fails, then the next in line is attempted.)
3. If a background image is specified, attempt to open it and copy it to the bitmaps. If NewIcon mode is activated, ignore it.
4. Call the datatype scaler.
5. If the datatype fails, and "addicon" is activated, create a default icon, (but only if no icon exsists OR "overwrite" is active)
 - 5a. If WhatIs is installed, ask it for a suitable icon based on the filetype.
 - 5b. Ask icon.library for the system default project icon.

1.9 Known Bugs

Known Bugs

- (1) ANNOYANCE A small bug that makes usage annoying.
 - (2) VISUAL A bug which makes visual output unattractive.
 - (3) GENERAL A general performance lack of a feature.
 - (4) SERIOUS Will not crash the machine, but will require a reboot.
 - (5) CRITICAL This is a spectacular crash.
- (4) Sometimes the JPEG datatype will freeze and then refuse to respond to all future requests. I'm not sure about the entire nature, but the only solution is a reboot. It will affect *ALL* programs afterwards, so it is probably the JPEG datatype's fault. ?
- Sending the Picticon process a signal via XOper usually restarts the Jpeg datatype. Any good Amiga user will have XOper sitting on his/her hard drive. Just send a \$FFFFFFFF signal to Picticon's process. Repeat if neccessary.
- (3) Any picture that's smaller than the domains (pic_x_size, pic_y_size) will be ignored by the aspect scaler. This is not too bad, as small images don't scale well.
 - (2) The GIF datatype I'm using sometimes stops decoding the image, resulting in a large, black rectangle at the bottom of the

icon. This usually happens on images encoded with the ppmto gif. Best solution is to re-encode the image with AdPro.

1.10 Stuff to do

To-do

- o Cleanup scaler's multiple mode handling.
- o A GUI preference program/window to make easy option changes? (in the works)
- ? Maybe a frame around the image, to distinguish it from the background?

1.11 Q&A

Questions and Answers (Q&A)

Q - When loading JPegs, it takes an awfully long time...

A - JPegs are slow, even on fast computers. Just run them in the background (LOWPRI~~~tooltype) and do something else. (Wow, multitasking.) Better yet, obtain the JFIF datatype. It's much faster and user-configurable! Make sure to send some money to Christoph for his efforts!

Q - The icon doesn't appear different after running Picticon on it.

A - If an exsisting icon is created with the NewIcon switch, and then run a second time without, the image will not appear to change. You must physically delete the offending icon, close any window in Workbench with it, then try again.

Q - Certain switches don't seem to work correctly when other switches are activated.

A - Exactly. See the list of incompatible switches.
Switch Modes

Q - When rendering HAM pictures into NewIcons, the image comes out muddy and lacking color. How do I fix this?

A - Well, the rendering routine simply uses the base palette for HAM pictures. Sometimes this is enough, but more often than not it isn't. To get a better picture, place Picticon in a grey scale mode. Or wait until I write a median-cut routine.

1.12 Author Info

Author Info

The author:

US.Snail: Chad Randall
229 S.Washington,
Manchester, Michigan
48158-9680 USA
Internet: crandall@garnet.msen.com
Phone: (313)-428-0113

Thanks to:

Jay Miner. For creating such a unique, wonderfull machine. May
he rest in peace.

Wouter van Oortmerssen. For "E" in all it's glory. E 3.0 is
GREAT!

Lionel Vintenat. The French locale catalog, for both Iconian and
Picticon.

Mark 'theMUG' Rose. For Picticon's icon. Hope he don't mind me
ripping it from his archive, and modifying
the app_icon a bit.

Colin Bell. My datatype routines are partially based on his.

David Larsson. For King-CON, of course!

D.L. McPaul. For the AmigaGuide Writer, used to create parts of
this guide.

Iconian betatesters. For finding so many bugs, I still haven't
gotten them all. I3 is on it's way!

Commodore (RIP). For all their mismanagment, non-marketing,
unfulfilled promises, and overall
empty-headedness. :P PPPPHTTT

Any company willing to hire me. Hey, I'll do free-lance work!
Mail me.

Other works by C.Randall

Previous Aminet released works include:

Iconian The premier 3.x icon editor that almost does it all, for now.

MWB2NI Converts a "magic-wb" icon into a "newicon".
Play KiSS Supports the KISS/GS format, and upto 256 colors.
kissutils CLI based programs to convert ppm<->kissdata.
Randomcopy Copies one file from a pattern list to a destination file.
Half-n-Half Fakes 1 meg of "fast" memory from 2 megs of "chip".
Fake20 Hacks execbase into thinking you have a '20 installed.
ColorSlash A boot picture based on the OS3.x system disks.

Yet to be released Amiga programs:

Iconian3 A complete rewrite of Iconian 1.98. (aka I3)

Other notable ancient projects:

Pit Bomber 4 mini-games in one. PET
Bust Out Break out clone. PET

Land Unique game where 2 players fight for land. VIC=20
Cartoon DBase Keep track of Saturday cartoons. VIC=20

To Hell&Back Ultima clone. C-64
Land Wars Updated Vic game. C-64

Un-named Unfinished Wizardry/Ultima clone. (Too big for me) IBM
B-Paint Monochrome paint program. IBM
C-Paint 256 color paint program with mouse support. IBM
Chess-4 A four player chess program based on Hoyles rules. IBM
Tetra Tetris clone. IBM
Petra Well-tris clone. IBM
Tank Wars Tank trajectory, landscape battle thingy. IBM
Gems 1,2 Two versions of Columns clone. IBM

Chess-4 IBM conversion for AmigaBASIC. Amiga
VCR database Simple movie indexer. Amiga
PopEd Program to create Populous gfx landscapes from IFFs. Amiga
Z-Mix Set of 2080 color display programs for OLD chipset. Amiga
Gems 3-6 4 versions of Columns clone. Amiga

btw, you can ask to see any of the above, but don't expect much.
Most of it's buried and long forgotten.

1.13 Iconian

Iconian

A great (IMHO) icon editor that deserves a look at. Features abound:

Features of Iconian v1.98

Support for sizable windows and custom fonts.
Multiple undos
Appwindows
Custom magnification levels

- Upto 256 colors
- 3.x palette sharing
- RGB color editing
- Integer scaling of oversized brushes
- Optional remapping of IFFs to screen colors
- Can save 8 plane icons which stay "normal" regardless of screen depth.
- Full range of drawing tools including beveled boxes
- Recolors 1.3 icons
- Clipboard support
- Extensive keyboard support
- Ability to cut and paste brushes
- Drawing with text, including selectable font and styles
- 100% system supported, no reqtools/explode/iff/powerpacker required

Look for it on Aminet in the gfx/edit directory. It requires OS 3.x.

" link

1.14 Registering a complaint

If and when you come across a bug, *please* let me know. Don't assume that someone else is going to complain about it, because they probably won't. In the past, I've found about bugs by visiting Usenet. Someone will post there complaining about something or other. If this user has Usenet access, then I'm sure he can EMail me! Picticon is not shareware, therefore I'm not committed to making updates. But I have in the past, and will probably do so in the future. Therefore, I'd like to know what the problem is so I can promptly fix it.

This goes for suggestions, too. Other than a GUI, I'm not really sure what to do with Picticon next. Any ideas? If so, then E-MAIL me! It's not like I'm asking for money, just a minute or two for some words.

1.15 Newicons

Newicons

First off, we must first clarify the two (2) parts of Newicon that exist. There's the icon collection, and then there's the system patch. I personally don't care for the iso-icons, so I won't be dealing with them. We are interested in the Newicon system-patch, exclusively.

To enable saving of newicons, you need to specify a "NI" on the CLI line, or a "NEWICON=YES" tooltype from Workbench.

Some features of Picticon become disabled when this is active. You can't make icons larger than 92x92 pixels, as Newicons won't allow it. No background image, as the palettes cannot be merged into the datatype palette. Dithering is not noticable, as the scaling function

uses the original palette. This means displaying a newicon image will result in considerably less quality than a dithered one. But of course, the trade off is that the newicon will look better on a different screen depth and/or palette, whereas an original icon will only look good on it's original screen format. A newicon file will usually be a bit larger, no matter how many colors are used. Consider all these trade-offs, and decide which is best for you. Time to experiment, huh?

When Picticon saves a newicon, it creates a "dot" icon for the original icon. This means if you trade the icon to someone who doesn't have newicons installed, they will only see a very tiny icon.

For my real thoughts of the NewIcons package, read: mwb2ni.doc

1.16 mwb2ni.doc

...(documentation about mwn2ni removed to save space)...

some personal notes on "newicons", nicola salmorcia, and roger mcvey...

read carefully. contains flames. these are OPINIONS and not to be taken seriously. each person in the world should be ENTITLED to his/her own OPINION.

first off, newicons (the program) is very nice. but, nicola's attitude put me off working with him on joint projects, very quickly. and i won't even mention that i personally dislike those iso-icons, nor trying to insist that everyone use them.

i'm not sure how i first met up with nicola, but we started exchanging email/files concerning iconian and newicons. i had offered to support newicons inside iconian. he supplied some developer material and ideas. i wrote some quick hacks and i had a version of iconian that read and wrote 256 color newicons. i gave it to nicola, and he flew off the handle! he wanted things a certain way, and would refuse to listen that i had made the newicon support a HACK. at that point, i gave up the idea of working with newicons and nicola.

i've saved all my email conversations with him, and i'd like to share a few highlights of my conversation. to try and point out why i have my opinions like i do...

nicola:

"The program is ready. It has been ready for months, now. It's only missing documentation and icons. Roger McVey is very busy at the moment." -- April 24, 1994

jeez..it was finally released in November of '94. had you released it a year earlier, magic workbench wouldn't have been so widespread. maybe you wouldn't have to dis it to try and get your package installed...anyways,

nicola:

"It IS possible, however, to directly access the data stored in the NewIcons - I can provide you routines to do so. Iconian should use a 256 color custom screen, however, not the Workbench screen, otherwise it may lose information. And should allow to modify the palette, of course."

ok, then why did you later tell me this:

nicola:

"Using a custom screen shouldn't be necessary. A 12 color icon can easily be edited on a 32-color Workbench, if you do things right. The new RemapChunkyImage() fucntion in newicon.library could be helpful (you did receive it, didn't you?)"

i know you say i can use the remapchunkyimage() function, but after EVERY time i modify the image? i think not...(overhead)

me:

"I'm trying to keep Iconian backwards compatible. Therefore, I can't really always use your newicon functions to read/write icons. Not everyone has or will have newicon.library (well, no one does...)"

nicola:

"Watch carefully the code I sent you. Those functions fall back to icon.library if NewIconBase is NULL. Using NewDiskObject internally is supposed to make things EASIER for you, since you don't have to handle NewIcons as a special case."

yea, right. your library makes handling 4 buffers instead of 2 invisible to iconian. dream on. i still have to handle newicons as a special case, so why not do it as a special case? jeez

nicola:

"Open NewIcon" and "Save NewIcon" must disappear. All editing should be performed via "Open" and "Save".

oh, is that right? iconian is my program, not yours. (of course, this insistent could be language translation problems...hhh)

nicola:

"Actually, it would be very nice to have a good icon editor distributed with NewIcons: it would make things much easier."

WAIT A SECOND you want to include iconian (a full blown, 140k program) with a SYSTEM PATCH?!? i think not! you need get your priorities straight. i'll include and promote your patch with my program, but not the other way around. i personally know that my program must have taken 10x the amount of work your 3k library took.

nicola:

"Let me know what you think. I really need an injection of enthusiasm to finally finish this project which is almost one year old." -April 24, 1994

wow. it was released in November. that makes it 1.5 years old. older than i thought.

to sum up. fuck nicola's and roger's idea of an isometric world. i say convert all your glorious magic-workbench icons to the newicon format just to piss them off! ;) later guys...

1.17 Hiya!

huh? What's this? Go away...I'm trying to sleep.