IconInstaller

Tom Ekström

COLLABORATORS			
	TITLE : IconInstaller		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY	Tom Ekström	January 17, 2023	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

IconInstaller

1.1 CONTENTS

```
                        ICONINSTALLER 3.1

      Icon Installation/Management Tool

          Freely Distributable
       Copyright © 1995 Tom Ekström
          All Rights Reserved



           ~1.~~~INTRODUCTION~~~~~
             IconInstaller described

          ~1.1.~Requirements~~~~~
            What is needed to use this program

          ~1.2.~Installation~~~~~
            What to put where


          ~2.~~~USAGE~~~~~~~~~~~~
            About IconInstaller operation

          ~2.1.~Basic~Usage~~~~~~
            Installing icons manually

          ~2.2.~Directory~Mode~~~
            Installing whole drawers of icons

          ~2.3.~Script~Mode~~~~~~
            Using prewritten scripts

          ~2.4.~Configuration~~~~
            Changing the settings

           2.5. Writing Scripts
            Writing IconInstaller scripts
```

1.2 INTRODUCTION

        1.   INTRODUCTION

IconInstaller is a tool to aid in icon management. The
              basic operation
               of
IconInstaller is to copy the imagery from one icon to another. Both standard
icons and
              NewIcons
               are supported, including conversion from standard icons
to NewIcons, and vice versa. IconInstaller can also be used to copy other
attributes (such as tooltypes, position, default tool etc.) in addition to,
or instead of, the imagery.

IconInstaller has a mouse-driven graphical user interface, where icons can

be selected with simple dragging and dropping, or alternatively using
standard ASL file requesters. Unique to IconInstaller is a
                 script mode
                 ,
where icons are selected for installation according to a script.
IconInstaller can also be used to easily
                 install whole drawers
                  of icons
with a single command.

IconInstaller requires at least AmigaDOS release 2.04. For more info, there
is a detailed list of
                 requirements
                 .

For legal information, read the sections about
                 copyright
                  and
                 disclaimer
                 .

1.3 REQUIREMENTS

   1.1. REQUIREMENTS

IconInstaller requires at least AmigaDOS version 2.04. Specifically, the
following libraries are needed:

asl.library
dos.library
exec.library
gadtools.library
graphics.library
icon.library
intuition.library
layers.library
workbench.library

Additionally, newicon.library is is used for NewIcon support. However,
newicon.library is not required to run IconInstaller.

IconInstaller takes up about 50 KBytes of disk space. At least about 100
KBytes of free RAM is required to run IconInstaller. If the subdirectory
mode is used with many nested levels of directories, more memory is consumed.

1.4 INSTALLATION

   1.2. INSTALLATION

There is no special installation procedure for IconInstaller. Simply drag

the program wherever you want it. For example, The 'Tools' drawer on your
Workbench disk is a good place.

1.5 USAGE

                        2.   USAGE

IconInstaller is a very versatile tool, and can be used in many different
ways. There is a normal mode for installing icons one by one, a directory
mode for installing whole drawers of icons, and a script mode for selecting
icons for installation according to a script.

In its basic configuration, IconInstaller will copy the image from one icon
to another, but also other icon attributes, such as default tool, icon type,
position, etc., can be selected for installation.
                        NewIcon
                         users can
also change the configuration to copy NewIcon images, and convert between
standard icons and NewIcons.

                        ~2.1.~Basic~Usage~~~~~~
                          Installing and deleting icons manually

                        ~2.2.~Directory~Mode~~~
                          Installing to drawers

                        ~2.3.~Script~Mode~~~~~~
                          Using a script to install icons

                        ~2.4.~Configuration~~~~
                          Changing the settings

                        ~2.5.~Writing~Scripts~~
                          Making your own scripts

1.6 USAGE

                        2.1. BASIC USAGE

Start IconInstaller with a double click on its icon.

If you are using the
                NewIcons
                 system, a requester may appear asking you to
select from some image installation options.

IconInstaller will start as 'empty', with no icons selected. To install the
image from one icon to another, you'll need to open a source icon and a

destination icon. The easiest way to open an icon is to drag and drop it
into the IconInstaller window. Source icons can be dropped in the box
labeled
                New Image
                , destination icons in the box labeled
                Original Image
                .
You can also open icons using a file requester. Select
                Open Source Icon...
                and
                Open Destination Icon...
                , respectively, from the 'Project' menu.
Finally, the destination icon can also be opened by typing its name into the
gadget labeled
                Destination Icon
                .

When both source and destination icons are opened, you can click on the

                Install
                 gadget to replace the original image of the destination icon with
the new one from the source. If you prefer using menus, you can also select

                Install Icon
                 from the 'Project' menu.

IconInstaller can also be used to delete icons from files, leaving the file
associated with the icon intact. To delete an icon, open it as the
destination icon as described above. Then, select
                Delete icon...
                 from the
'Project' menu. IconInstaller will display a requester asking you to confirm
this action.

IconInstaller is also able to install multiple icons in one go, either a
whole
                directory
                 at once, or according to a
                script
                .

Furthermore, you can
                configure
                 the installation by selecting other icon
attributes to install from one icon to another, in addition to (or instead
of) the imagery, as well as change some other options.

1.7 USAGE

                    2.2. DIRECTORY MODE

Please read first the section describing
                basic~usage
                 of IconInstaller,

where opening source and destination icons is explaned, before proceeding
with this section.

To start directory installation, open both source and destination icons.
Then, click on the
                  Dir
                   gadget. A requester will appear, asking for the
directory (drawer) where the installation should take place. By default, this
is the directory where the destination icon is located.

IconInstaller will read the icons in the directory, one by one, and open
each icon in its turn as the destination icon. For each icon, you can select
to install the image from the source icon, or skip the icon and move on to
the next one. Click on
                  Install
                   or
                  Skip
                   to make your choice. This process
is repeated for all icons in the directory.

You can instruct IconInstaller to complete the directory without further
interaction by clicking on the
                  All
                   gadget. A requester will appear, asking
you to confirm this option.

You can also click on the
                  Quit
                   gadget at any point, to abort the directory
installation.

The 'Settings' menu contains options to change the behaviour of directory
installation. The options can be changed by setting and clearing subitems
for the
                  Directory
                   menuitem. The options available are 'Match Type',
'Create Icons' and 'Enter Directories'.
Match Type will accept for installation only those icons that are of the
same type as the source icon.
Create Icons will give you the option to create icons for files that don't
have icons.
Enter Directories means that the installation will proceed not only in this
directory, but all its subdirectories as well.

1.8 USAGE

                        2.3. SCRIPT MODE

Using IconInstaller in script mode is actually the easiest way to use it.
You don't even have to open icons for installation, as they are selected
automatically by the script.

You start a script by double clicking on the script's icon.

If you are using the
                NewIcons
                 system, a requester may appear asking you
to select from some options relating to the imagery installation.

The IconInstaller window will be opened as usual. IconInstaller will start
to open source and destination icons according to the script, presenting you
the new image from the source icon, and the original image of the
destination icon, side by side, before the installation. You can click on
these images to see the alternate imagery, if there is any.

For each icon you can select if you want to install the new image to the
destination icon, or keep the original image. Click on
                Install
                 or
                Skip
                ,
respectively, to make your choice. IconInstaller will then open the next
source and destination icons according to the script. This process is
repeated for each pair of icons, until the script is complete.

As in the
                directory~mode
                , you can instruct IconInstaller to complete the
script without further interaction, by clicking on the
                All
                 gadget. A
requester will appear asking you to confirm that you really want to do this.
You can also click on the
                Quit
                 gadget at any point, to abort the script
and reset IconInstaller to
                basic~usage
                .

Depending on the script, you may also be presented with messages and
requesters either to give you information, or prompt you to select from
some available options.

An IconInstaller script is a standard ASCII text file, that can be created
with any text editor, such as 'Ed' or 'Emacs'. For icon authors wanting to
distribute IconInstaller with their releases, or other users wishing to
automate the installation of a large number of icons, there is a section
about
                writing your own scripts
                , as well as a
                reference section
                 with all
commands and options available.

1.9 USAGE

                2.4. CONFIGURATION

In its default configuration, IconInstaller will copy the imagery from one
icon to another, while retaining the other attributes of the destination
icon, such as tooltypes, default tool, position, and so on. If you have the

              NewIcons
               patch running, the default action is to copy the NewIcon image,
while keeping the standard image as it is.

The default configuration can be changed with
              tooltypes
               in the IconInstaller
icon, so that every time you start IconInstaller, your preferred
configuration is active. To change the default configuration for a
script, tooltypes in the script's icon should be set instead.

While using IconInstaller, the configuration can be changed from the

              Settings
               menu.

The menu contains options for manipulating the
              Image
              , the
              NewIcon imagery
              ,

              Icon Type
              ,
              Default Tool
              ,
              Tool Types
              ,
              Position
               and
              Stack
              .

All of these attribute settings contain at least two options, 'Keep Original'
and 'Copy From Source'. The former means that the attribute in the destination
icon is not changed, the latter means that the attribute is copied from the
source icon to the destination icon.
The 'Image' and NewIcon settings also contain options to convert imagery from
the standard to NewIcon format, and vice versa. The 'Icon Type' and 'Default
Tool' settings contain many options to set the attribute explicitely.

In addition to the attribute settings, the 'Settings' menu also includes

              directory options
               to affect the behaviour of directory installation.
Finally, there is an option to
              quickly install icons
               with dragging
and dropping.

1.10 USAGE

2.5. WRITING ICONINSTALLER SCRIPTS

An IconInstaller script is a standard ASCII text file, that can be
created with any half-decent text editor, such as the 'Ed' editor
supplied with AmigaDOS.

An icon has to be created for the script, as currently the only way
to start a script is to double click on the script icon on Workbench.
The script icon type must be 'Project', and the default tool must
be 'IconInstaller'. The script icon can be conveniently created with
IconInstaller itself, for example.

IconInstaller will identify a script by examining the first line.
The first line has to start with the keyword

ICONINSTALLER
so that IconInstaller will know that this file is meant as a ↩
script.

Optionally, a version number may be included after the ICONINSTALLER
keyword. This guarantees that the script will not be executed with
an older version of IconInstaller. If you are using script commands
that are not available in early versions of IconInstaller, it could
be worthwhile to specify the version number. Version 3.0 can be
assumed as the lowest possible version number.

Empty lines and lines starting with a semi-colon (
;
) are ignored
by IconInstaller. This way, comments and extra spaces can be included
for better readability.

The command to install the imagery from one icon to another takes
always the following form:

FROM
 <Source Icon>
TO
 <Destination Icon> [OPTIONS]

If an icon name contains blank spaces, the name must be included in
quotes. The trailing .info extension for an icon file name may be included,
but it is not necessary.

For example, the command

FROM Icon_1 TO "Icon 2"

Will copy the imagery from Icon_1 to Icon 2.

There are seven options available for icon installation commands.

The options can be used alone or in any combination. Not all combinations
are very useful, though. The options are

>   ICONTYPE
>   Specify the type for the destination icon
>
>   QUIET
>   Suppress any 'Please insert volume...' requesters
>
>   FORCE
>   Create the destination icon if it doesn't exist
>
>   MATCHTYPE
>   Skip the destination icon if its type doesn't match the source
>
>   DIRECTORY
>   Install to all icons in a directory (drawer)
>
>   RAMSCRIPT
>   Offer to append a Ram Disk icon script to S:User-Startup
>
>   CHECKSYS
>   Look around the SYS: disk for the destination icon

In addition to the icon installation command, there are three more
commands available.

The level of automation can be specified with the SCRIPTMODE command.
The format for this command is

>   SCRIPTMODE
>   <AUTOMATIC|SKIP|MANUAL>

SKIP is the default mode, and provides good manual control. The AUTOMATIC
mode will install everything without any confirmation. It is NOT recommended
to set the AUTOMATIC mode for scripts that are distributed with icon
collections to the general public. The MANUAL mode gives the user the best
manual control, as even skipping of icons that are not found must be
confirmed.

It is also possible to display messages to the user. This is done with the
MESSAGE command. The format for this command is

>   MESSAGE
>   "<Show this text>"

A requester will be displayed, containing the text and a single 'OK'
gadget. If the text to be displayed contains any blank spaces, include it
in quotes. The special character combination ('\n') can be used to split
the text in the requester on several lines.

Some icons require a custom Workbench palette to make sure that the icons
appear correctly. This is because normal Amiga icons don't contain any
colour data, and depend on the Workbench palette for their looks. It is

possible to install a palette from within a script with the INSTALLPALETTE
command. The format for this command is

                    INSTALLPALETTE
                     "Palette file"

The argument is the name of the file that contains the palette to be
installed. A requester will be displayed, offering the user the option
to install the palette.

Two additional options,
                    IFNEWICONS
                     and
                    IFNOTNEWICONS
                    , can be used in
combination with the SCRIPTMODE, MESSAGE and INSTALLPALETTE commands.
If the IFNEWICONS option is used, the command will be executed only if
the NewIcons patch is installed on the user's system. The IFNOTNEWICONS
option has the opposite effect, causing the command to be executed only
if the NewIcons patch is NOT present.

If you are reading this documentation from the original IconInstaller
archive, there is a fully commented example script called 'Install_Icons'
included. Click here to take a look at the example script.

If you intend to distribute a script with an icon collection for the
general public, it is recommended that the
                    NOSETTINGS
                     and

                    ASKNEWICONS
                     tooltypes are set in the script's icon, but not any
other tooltype. This will guarantee that only the imagery will be installed,
and that the user will not be able to modify the settings in the middle of a
script. Read the
                    copyright
                     section for more distribution information.

1.11 REFERENCE

                    3.   REFERENCE

This chapter contains reference information for the IconInstaller window,
the menus, the tooltypes and the script commands. For each section, a
brief overview will first be presented, with more detailed description
for every item available by selecting from the overview list.

                    3.1. WINDOW
                        The elements in the IconInstaller window

                    3.2. MENUS
                     The menuitems and subitems in the menus

1.12 REFERENCE

3.1. WINDOW

This is a brief list of all elements in the IconInstaller main window.
This includes all gadgets and areas for icon dragging and dropping.
For a more complete description for each element, click on any item on
the list below.

SOURCE ICON BOX
 Displays the new icon image

DESTINATION ICON BOX
 Displays the original icon image

INSTALL GADGET
 Install the imagery to the destination icon

DIR GADGET
 Install to all icons in a directory

ALL GADGET
 Install rest of script/directory automatically

SKIP GADGET
 Skip the destination icon in script/directory mode

QUIT GADGET
 Quit IconInstaller (Abort script/directory)

DESTINATION ICON GADGET
 Displays the name of the destination icon

CLOSEGADGET
 Quit IconInstaller

ZOOMGADGET
 Shrink the IconInstaller window

DEPTHGADGET
 Put IconInstaller behind/in front of other windows

Many of the gadgets have also an equivalent menu item in the
            Project

menu.

1.13 WINDOW ELEMENTS

3.1.1. SOURCE ICON BOX

The source icon box is the left one of the two 'recessed' areas in the
IconInstaller window. The box is labeled 'New Image'.

The box shows what the destination icon will look like on the Workbench
after the installation.

Usually, this is the imagery of the source icon. If no source icon is
opened, the box will be empty. Depending on the settings, the source icon
box may also display something else than the source icon image. For example,
if the
                Image
                 option in the 'Settings' menu is set to 'Keep Original', the
destination icon image will be shown instead. This is because in this case,
the imagery will not be changed.
On computers with the
                NewIcons
                 system installed, the box will usually
display the NewIcon image of the source icon, but this also depends both
on the 'Image' and
                NewIcon settings
                 , and there are even more options than
without NewIcons.

Dragging and dropping icons into the source icon box is a convenient way
to open them as source icons. Source icons can also be opened with a file
requester, selecting
                Open Source Icon...
                 from the 'Project' menu.

Clicking on the icon image in the source icon box will display the
alternate image, if there is any. This is the image that is displayed
when the icon is selected on Workbench.

1.14 WINDOW ELEMENTS

3.1.2. DESTINATION ICON BOX

The destination icon box is the right one of the two 'recessed' areas in
the IconInstaller window. The box is labeled 'Original Image'.

The destination icon box displays the original imagery for the destination
icon. In other words, this is how the destination icon looks like on
Workbench before the installation. If the box is empty, there is no
destination icon opened.

If the
                    NewIcons
                     patch is installed, and the destination icon contains
NewIcon imagery, this will be displayed instead of the standard image.

Destination icons can be opened by dragging and dropping them into the
destination icon box. Other ways to open a destination icon is by typing
its name into the
                    Destination Icon
                     gadget, or with a file requester
by selecting
                    Open Destination Icon...
                     from the 'Project' menu.

Clicking on the image in the destination icon box will show the alternate
image, if the destination icon contains any.

1.15 WINDOW ELEMENTS

                        3.1.3. INSTALL GADGET

Clicking on the 'Install' gadget is used to perform the actual
installation. Usually, this means that the original imagery of the
destination icon will be replaced with the one from the source.

Depending on the
                    settings
                     , also other attributes in addition to
(or instead of) the imagery can be selected for installation. The
other attributes are icon type, default tool, tooltypes, position
and stack size. On computers using the
                    NewIcon
                     system, this includes
also the NewIcon imagery.

This gadget is enabled only when both source and destination icons are
opened.

The 'I' key on the keyboard can be used as a shortcut key for this gadget.

Also, the
                    Install Icon
                     menuitem in the 'Project' menu can be used to
perform the same action.

1.16 WINDOW ELEMENTS

                        3.1.4. DIR GADGET

The 'Dir' gadget is used to invoke the
                directory mode
                , making the
installation to a whole directory of icons at once, instead of just
a single icon.

A requester will be displayed, asking for the name of the destination
directory. By default, this is the directory where the destination icon
is located.

This gadget will be disabled if IconInstaller is already in directory
mode, or in the middle of a script. The gadget is disabled also if
either the source or the destination icon is not opened.

The
                Directory
                 menuitem in the 'Settings' menu contains some options to affect
the behaviour of directory installation.

The 'D' key on the keyboard can be used as the shortcut key for the 'Dir'
gadget.

The
                Install Directory...
                 menuitem in the 'Project' menu can also be used
to perform the same action.

1.17   WINDOW ELEMENTS

                3.1.5. ALL GADGET

The 'All' gadget is available only when IconInstaller is in
                directory
                 or

                script
                 mode. By selecting this gadget, the rest of the icons in the ←
                    script
or the directory are installed automatically, without further interaction
with the user.

A requester will be displayed, asking the user to confirm this action.

The 'A' key on the keyboard can be used as a shortcut for this gadget.

1.18   WINDOW ELEMENTS

                3.1.6. SKIP GADGET

The 'Skip' gadget is available only in

```
                    directory
                     or
                    script
                     mode, where
multiple destination icons are automatically opened for installation, one
after another.
```

The 'Skip' gadget is used when the user decides not to install the
presented icon, and instead move on to the next one. In that case, the
destination icon will not be changed at all.

In
```
                    basic mode
                    , when installing icons manually, this gadget is
disabled.
```

The 'S' key on the keyboard can be used as a shortcut key for this gadget.

1.19 WINDOW ELEMENTS

```
                    3.1.7. QUIT GADGET
```


The 'Quit' gadget is used to quit IconInstaller. A requester will
be displayed, asking you to confirm the quitting.

Clicking this gadget will reset IconInstaller to an empty state with
no icons opened. The configuration will be reset to the initial values.
Thus, this gadget can also be used to abort a
```
                    script
                     or a
                    directory
                    installation.
```

The 'Q' key on the keyboard can be used as a shortcut key for this gadget.

The
```
                    Close
                     gadget and the
                    Quit...
                     option in the 'Project' menu perform
similar functions. Their function is different from the 'Quit' gadget in
that they don't reset IconInstaller.
```


1.20 WINDOW ELEMENTS

```
                    3.1.8. DESTINATION ICON GADGET
```


The destination icon gadget displays the name of the destination icon.
If no destination icon is opened, this gadget is empty.

This gadget can be used to open the destination icon, by typing its
name into this gadget. Other ways to open a destination icon are
to drag it into the
                destination icon box
                , or using a file requester
by selecting
                Open Destination Icon...
                 from the 'Project' menu.

If this gadget is active (ie. if a square cursor is in this gadget)
the name will be displayed in full, including the directory path.
If the gadget is not active, only the filename part will be displayed.

If IconInstaller is currently reading a the next icon in a script or a
directory, the text 'reading...' will be displayed.

The shortcut key 'N' can be used to activate this gadget. The gadget
can also be activated by clicking in it with the mouse.

1.21   WINDOW ELEMENTS

                3.1.9. CLOSE GADGET

This is a standard Workbench gadget in the upper left corner of the
window.

This gadget is used to quit IconInstaller. A requester will be displayed,
asking the user to confirm this action.

This gadget has similar function to the
                Quit
                 gadget and the
                Quit...
                menuitem in the 'Project' menu.

1.22   WINDOW ELEMENTS

                3.1.10. ZOOM GADGET

This is the left one of the two gadgets in the upper right corner
of the IconInstaller window. This is a standard Workbench gadget.

Clicking on this gadget will shrink the IconInstaller window to just
a narrow strip displaying only the window's title bar and the
                Close
                ,
'Zoom', and
                Depth
                 gadgets. The rest of the GUI will be hidden from
view, but the

```
               menus
                will still be available.
```

This gadget can be used to temporarily hide the IconInstaller window
if the Workbench screen is very cluttered.

1.23   WINDOW ELEMENTS

```
   3.1.11. DEPTH GADGET
```

This is a standard Workbench gadget, the right one of the two gadgets in
the upper right corner of the IconInstaller window.

With this gadget, you can arrange the visibility of the IconInstaller
window respective to other windows on the Workbench screen, by moving
IconInstaller in front of or behind the other windows.

1.24   REFERENCE

```
            3.2. MENUS
```

IconInstaller has two menus, 'Project' and 'Settings'. Items in the
'Project' menu are used to perform the different actions of IconInstaller.
The 'Settings' menu can be used to affect how these actions are performed.

```
        PROJECT
         The available actions

        SETTINGS
         The configuration options
```

1.25   MENUS

```
        3.2.1. PROJECT MENU
```

Below is a list of all items in the 'Project' menu, with a brief description
of each. You can click on any menuitem for more details.

```
        ~Open~Source~Icon...~~~~~
          Open the icon containing the image to be installed

         Open~Destination~Icon...~
           Open the icon that will receive the new imagery
 ---------------------------
```

```
            ~Install~Icon~~~~~~~~~~~ ~
              Install the destination icon back to disk

            ~Install~Directory...~~~~
              Install a whole drawer of icons at once
  --------------------------

            ~Delete~Icon...~~~ ~~~~~ ~
              Delete the destination icon
  --------------------------

            ~About~~~~~~~~~~~~~~~~~~ ~
              Display information about IconInstaller
  --------------------------

            ~Quit...~~~~~~~~~~~~~~~~ ~
              Quit IconInstaller
```

1.26 PROJECTMENU ITEMS

                    3.2.1.1. OPEN SOURCE ICON MENUITEM


```
The 'Open Source Icon...' is used to open the source icon using a
file requester. The file requester will only display icon files and
directories.

An alternative way to open the source icon is to drag and drop it
into the
                source icon box
                .

This menuitem can also be selected with the [Right Amiga]-S hotkey.
```

1.27 PROJECTMENU ITEMS

                    3.2.1.2. OPEN DESTINATION ICON MENUITEM


```
The 'Open Destination Icon...' menuitem is used to open the destination
icon with a file requester. The file requester will only display icon
files and directories.

An alternative way to open the destination icon is to drag and drop it
into the
                destination icon box
                , or by typing its name into the

                Destination Icon
                 gadget.
```

This menuitem can also be selected with the [Right Amiga]-D hotkey.

1.28   PROJECTMENU ITEMS

3.2.1.3. INSTALL ICON MENUITEM

The 'Install Icon' menuitem is used to install the destination icon to
disk.

With the default IconInstaller
                configuration
                , this means that the original
imagery of the destination icon will be replaced by the imagery from the
source icon.

This menuitem is enabled only when both source and destination icons are
opened.

The [Right Amiga]-I key on the keyboard can be used as a hotkey for this
menuitem.

The
                Install
                 gadget performs the same action.

1.29   PROJECTMENU ITEMS

3.2.1.4. INSTALL DIRECTORY MENUITEM

The 'Install Directory' menuitem can be used to start the
                directory mode
                ,
making the installation to a whole directory of icons at once, instead of
just a single icon.

A requester will be displayed, asking for the name of the destination
directory. By default, this is the directory where the destination icon
is located.

This menuitem is disabled if IconInstaller is already in directory mode
or in the middle of a script. The gadget is disabled also if either the
source or the destination icon is not opened.

The
                Directory
                 menuitem in the 'Settings' menu contains some options to affect
the behaviour of directory installation.

The
                Dir

gadget can be used to perform the same action.

1.30 PROJECTMENU ITEMS

3.2.1.5. DELETE ICON MENUITEM

This menuitem is used to delete icons. If there is a file associated
with the icon, it will remain as it is.

To delete an icon, it must be opened as the destination icon. This
can be done, for example, by dragging and droppping it into the

destination icon box
.

When this menuitem is selected, a requester will be displayed asking
to confirm the deletion.

This menuitem is disabled if the destination icon is not openend.

The [Right Amiga]-E key on the keyboard can be used as a hotkey for this
menuitem.

1.31 PROJECTMENU ITEMS

3.2.1.6. ABOUT MENUITEM

Selecting this menuitem will display a small message containing the current
IconInstaller version number, as well as brief information about the author,
copyright, and distribution.

1.32 PROJECTMENU ITEMS

3.2.1.7. QUIT MENUITEM

This menuitem is used to quit IconInstaller. A requester will be displayed,
asking the user to confirm this action.

This menuitem has similar function to the
Quit
and
Close
gadgets.

1.33   MENUS

                    3.2.2. SETTINGS MENU

The 'Settings' menu contains options for selecting what will happen to the
different attributes of the destination icon when it is installed. Changes
made to the 'Settings' menu will be lost when IconInstaller is quit. To
change the default settings, include
                tooltypes
                 in the IconInstaller icon
(or in the case of scripts, the script icon).

In addition to the attribute settings, the menu contains also options for
directory installation and quick icon installation.

Below is a list of all items in the 'Settings' menu, with a brief description
of each. You can click on any menuitem for more details.

                ~Image~~~~~~~~~~~~~~
                  Options for selecting the new imagery

                ~NewIcon~~~~~~~~~~~~
                  Options for the NewIcon imagery

                ~Icon~Type~~~~~~~~~~
                  Options for setting the icon type

                ~Default~Tool~~~~~~~
                  Options for selecting a default tool

                ~Tool Types~~~~~~~~~
                  Options to change or keep the tooltypes

                ~Position~~~~~~~~~~~
                  Options for the destination icon position

                ~Stack~~~~~~~~~~~~~~
                  Options for the destination icon stack

                ~Directory~~~~~~~~~~
                  Settings to affect the directory mode

                ~Drag~&~Drop~~~~~~~~
                  Option for quick installation

A special tooltype,
                NOSETTINGS
                , can be set to disable the 'Settings' menu.
This can be used as a safety measure to prevent inexperienced users from
accidentally changing the settings to undesirable values.

1.34   SETTINGSMENU ITEMS

                        3.2.2.1. IMAGE MENUITEM

The 'Image' menuitem is used to control what will happen to the imagery
of the destination icon when it is installed.

There are four available options.

Keep Original

   The original imagery of the destination icon will not be changed.

Copy Source Image

   The imagery of the destination icon is replaced by imagery from the
   source icon.

Copy Source NewIcon

   The

               NewIcon
                imagery from the source icon is copied as the standard
   imagery of the destination icon
   The image will be remapped to the current Workbench colours and depth.
   The colour information will be lost in the conversion, as the standard
   icon imagery depends on the Workbench palette for its colours.
   If the source icon does not contain any NewIcon image, the standard
   image from the source icon will be copied instead.
   This option is disabled if newicon.library is not found in the LIBS:
   directory.

Make Dot

   The imagery of the destination icon is replaced with a small dot. This is
   probably useful only for icons that contain both standard and NewIcon
   imagery, since in that case the original image will not be shown anyway.
   As the icon (.info) file will be much smaller, this can save quit a bit of
   disk space. However, this setting is recommended only for those who use
   the NewIcons patch all the time.
   As with the 'Copy NewIcon' option, if newicon.library is not found, this
   option will be disabled.

The default setting is to 'Copy Source Image'. The default setting can be
changed with the
               IMAGE
                tooltype.

Notes to NewIcon users: This menuitem always controls the standard imagery
of the icon. The NewIcon settings are specified with the
               NewIcon menuitem
                        .
If the NewIcons patch is installed, the default setting for the Image
attribute will be set to 'Keep Original Image'.

1.35   SETTINGSMENU ITEMS

                              3.2.2.2. NEWICON MENUITEM

The 'NewIcon' menuitem is used to control what will happen to the

                NewIcon
                  imagery of the destination icon when the icon is installed.
The settings for the standard icon imagery are set separately with
the
                Image
                  menuitem.

This menuitem is disabled if the newicon.library is not found in the
LIBS: directory. Note that only the library is required, the NewIcons
patch itself does not have to be installed. However, if the patch is
not installed, these settings will not affect the imagery shown in the
box labeled
                New Image
                  . This is because the box always shows what the
destination icon will look like on the Workbench screen after the
installation, and if the patch is not installed, any NewIcon imagery in
the icon will be hidden.

There are five options available.

Keep Original

   The NewIcon imagery of the destination icon will not be changed.

Copy Source NewIcon

   The NewIcon imagery from the source icon will be copied as the
   destination icon's NewIcon imagery.
   If the source icon does not contain any NewIcon imagery, the standard
   imagery from the source icon will be copied instead. The action will
   be as in the 'Copy Source Image' option below.

Copy Source Image

   The standard imagery from the source icon will be copied as the
   NewIcon imagery of the destination icon.
   The converted NewIcon image will be an exact match of the standard
   icon image as it is displayed on the Workbench. However, only those
   colours that are actually used in the imagery are included in the
   NewIcon, making the icon files small. The resulting NewIcon image
   will contain colour information and look good independently of the
   Workbench palette. Images up to 85 x 85 pixels are supported, larger
   images are cropped to the maximum size.

Remap Original Image

   The destination icon's own standard imagery will be copied as the
   icon's NewIcon imagery.

Erase

   Any NewIcon imagery in the destination icon will be deleted.

If the NewIcons patch is installed, the default setting is to 'Copy
Source NewIcon'. Otherwise, default is to 'Keep Original'.
The default setting can be changed with the
                  NEWICON
                   tooltype.

1.36   SETTINGSMENU ITEMS

                     3.2.2.3. ICON TYPE MENUITEM

The 'Icon Type' menuitem is used to determine what the type of the
destination icon will be when it is installed. This option is the
most likely cause of problems for novice users, so it is advisable to
change this setting only if you know what you are doing.

There are eight possible alternatives.

Keep Original

   The type of the destination icon will not be changed.

Copy From Source

   The destination icon type will be copied from the source icon.

Disk

   The type of the destination will be set to 'Disk'. This is the type
   of all disk icons on the Workbench screen, be it floppy disks, hard
   disks or Ram disks.

Drawer

   The type of the destination will be set to 'Drawer'. This is the type
   of all directory icons.

Tool

   The type of the destination will be set to 'Tool'. All applications
   and programs should have their icons of this type.

Project

   The type of the destination will be set to 'Project'. All data files
   created with an applications program are set to this type.

Trashcan

   The type of the destination will be set to 'Trashcan'. The only icons

of this type are the trashcan drawers on each disks. It is probably
not worthwhile to use this type for any other icon.

Kick

   The type of the destination will be set to 'Kick'. This is the type
   of any inserted disk that is not recognized by AmigaDOS. This includes
   Kickstart disks. This type is probably useful only for the
   'ENVARC:Sys/def_kick.info' icon, that Workbench will display for any
   unrecognized disk that is inserted.

The default setting is to 'Keep Original' type, unless the
                 ICONTYPE
                 tooltype is set to something else.

1.37   SETTINGSMENU ITEMS

                    3.2.2.4. DEFAULT TOOL MENUITEM

The 'DefaultTool' menuitem in the 'Settings' menu is used to control
what will happen to the default tool of the destination icon when the
icon is installed.
This menuitem is disabled if the
                 IconType
                  settings are set to something
else than 'Project', 'Keep Original' or 'Copy From Source'.

There are seven options available.

Keep Original

   The default tool of the destination icon is not changed.

Copy From Source

   The default tool of the destination icon is replaced by the
   default tool from the source icon.

MultiView

   The default tool for the destination icon will be set to 'MultiView'.
   'MultiView' can be replaced with some other preferred tool by setting
   the tooltype
                 TOOL1
                 .

AmigaGuide

   The default tool for the destination icon will be set to 'AmigaGuide'.
   'AmigaGuide' can be replaced with some other preferred tool by setting
   the tooltype
                 TOOL2
                 .

```
Display
```

```
    The default tool for the destination icon will be set to 'Display'.
    'Display' can be replaced with some other preferred tool by setting the
    tooltype
                TOOL3
                .
```

```
More
```

```
    The default tool for the destination icon will be set to 'More'.
    'More' can be replaced with some other preferred tool by setting the
    tooltype
                TOOL4
                .
```

```
Other Tool...
```

```
    A requester with a string gadget will be displayed, asking the user
    to specify a default tool for the destination icon. The default
    string to be used can be specified by setting the tooltype
                OTHERTOOL
                .
```

```
The default setting is to 'Keep Original' default tool, unless the
                DEFTOOL
                tooltype is set to something else.
```

1.38 SETTINGSMENU ITEMS

```
                3.2.2.5. TOOL TYPES MENUITEM
```

```
The 'Tool Types' menuitem in the 'Settings' menu control what will happen
to the destination icon tooltypes when the icon is installed.
```

```
Two choices are available.
```

```
Keep Original
```

```
    The tooltypes are kept as they are in the destination icon.
```

```
Copy from Source
```

```
    The tooltypes from the source icon are copied to the destination
    icon.
```

```
The default setting is to 'Keep Original' tooltypes, unless the
                TOOLTYPES
                tooltype is set to something else.
```

1.39 SETTINGSMENU ITEMS

3.2.2.6. POSITION MENUITEM

The position for the destination icon can be controlled with options
under the 'Position' menuitem in the 'Settings' menu. The position of an
icon on Workbench is related to the upper left corner of the window where
the icon is located.

There are three available options.

Keep Original

   The position for the destination icon will not be changed.

Copy From Source

   The position for the destination icon will be copied from the source
   icon.

Free

   The destination icon position will be cleared, which means that
   it is left up to the Workbench to find a suitable position for the
   icon. This has the same effect as selecting 'UnSnapshot' for an
   icon from the Workbench 'Icons' menu.

The default setting is to 'Keep Original' position, unless the
               POSITION
               tooltype is set to something else.

1.40 SETTINGSMENU ITEMS

3.2.2.7. STACK MENUITEM

The options under the 'Stack' menuitem control what is done to the
stack size in the destination icon when the icon is installed.

There are two options.

Keep Original

   The stack size is kept as it is in the destination icon.
   This is the default setting, unless the STACK tooltype is set
   in the IconInstaller icon to something else.

Copy From Source

   The stack size is copied from the source icon.

The default setting is to 'Keep Original' stack size, unless the
                STACK
                tooltype is set to something else.

1.41   SETTINGSMENU ITEMS

                       3.2.2.8. DIRECTORY MENUITEM

The options related to directory installation can be found under the
'Directory' menuitem in the 'Settings' menu. These options can be used
in any combination.

There are three available options.

Match Type?

   The icons in the destination directory are compared with the source icon.
   If the types of the icons are the same, the destination icon is accepted
   for installation. If they are different, the destination icon will be
   skipped and IconInstaller will move on to the next icon.
   If the
                Icon Type
                 setting is set to something else than 'Keep Original'
   or 'Copy From Source', the type of the destination icon will be matched
   against the specified type instead of the source icon's type.
   This option can be used to install new imagery only to project icons in a
   drawer, for example.
   This option is enabled by default.

Create Icons?

   If files are encountered that don't have icons, IconInstaller will give
   you the option to create icons for them. In a case like this, a suitable
   default icon is used for those attributes that are not copied from the
   source icon.
   This option is initially disabled.

Enter Directories?

   Normally, new imagery is installed only to icons in the specified
   directory. When this option is set, IconInstaller will also enter all
   subdirectories that it encounters, installing icons in a whole hierarchy
   of directory levels in one go. Use this option carefully, as it is very
   easy to mess up a lot of icons this way.
   This option is disabled by default.

1.42   SETTINGSMENU ITEMS

                       3.2.2.9. DRAG & DROP MENUITEM

There is one option to affect the dragging and dropping of icons into
the IconInstaller window.

Quick Install?

      When this option is set, the destination icon will be installed
      as soon as it is dropped into the destination icon box, without
      having to click on the
                  Install
                   gadget. This makes the
      installation a bit faster, especially if many icons have to be
      installed to get the same attributes.
      This option is disabled by default.

1.43 REFERENCE

                        3.3. TOOLTYPES

In its default configuration, IconInstaller will only copy the imagery from
the source icon to the destination icon, leaving the icon position, default
tool etc. unchanged. You can change the default configuration by including
tooltypes in IconInstaller's icon, so that your preferred settings are
active every time you start IconInstaller.

If you start IconInstaller with an accompanying script, the tooltypes are
read from the icon of the script instead of IconInstaller's icon. This way,
each script can have its own, preferred settings, too.

Below is a brief listing of all available tooltypes for IconInstaller. More
complete description of each tooltype is available by clicking on the name
of the tooltype.

                  ~IMAGE~~~~~~~~~
                    Options for installing the icon imagery

                  ~NEWICON~~~~~~~
                    Options for installing the newicon imagery

                  ~ICONTYPE~~~~~~
                    Options for installing the type of the icon

                  ~DEFTOOL~~~~~~~
                    Options for installing the icon's default tool

                  ~TOOLTYPES~~~~~
                    Options for installing the icon tooltypes

                  ~POSITION~~~~~~
                    Options for installing the icon position

                  ~STACK~~~~~~~~~
                    Options for installing the stack size in the icon

```
            TOOL1
             Specify an alternative for 'MultiView'

            TOOL2
             Specify an alternative for 'AmigaGuide'

            TOOL3
             Specify an alternative for 'Display'

            TOOL4
             Specify an alternative for 'More'

            OTHERTOOL
             Specify an alternative default tool


        ~ASKNEWICONS~~~
          Display a requester asking for some newicon options

        ~SIMPLEREFRESH~
          Open the IconInstaller GUI as a simplerefresh window

        ~NOSETTINGS~~~~
          Disable the 'Settings' menu
```

1.44 TOOLTYPES

```
              3.3.1. IMAGE TOOLTYPE
```

With the IMAGE tooltype you can specify the default setting for what happens
to the destination icon's image.

There are four possible options to this tooltype. More detailed information
for each option is available for the corresponding
              Image
               menuitem.

KEEP

   The destination icon's imagery is kept as it is.

COPY

   The imagery from the source icon is copied over to the destination icon.

NEWICON

   The
              NewIcon
               image from the source icon is copied as the standard icon
   image for the destination icon. If newicon.library is not found in the
   LIBS: directory, the default setting will be used instead.

DOT

   The imagery of the destination icon is replaced with a small dot.
   As with the NEWICON option, if newicon.library is not found, the default
   setting will be used.

If the IMAGE tooltype is not included at all, the default setting is to COPY
the image from the source icon. If the NewIcons patch is installed, the
default setting is to KEEP the standard image. This is because in this
case, the NewIcon imagery will be copied instead by default.

Example:

IMAGE=NEWICON

1.45 TOOLTYPES

                        3.3.2. NEWICON TOOLTYPE

With the NEWICON tooltype you can specify the default setting for what happens
to the destination icon's NewIcon image. Note that this is separate from
the standard icon image.

If the newicon.library is not installed in the LIBS: directory, this tooltype
is ignored. Note that the
                NewIcons
                 patch itself does not have to be
installed.

There are five possible options to this tooltype. More detailed information
for each option is available for the corresponding
                NewIcon menuitem
                 .

KEEP

   The NewIcon image is not changed.

COPY

   The NewIcon image is copied from the source icon. If the source icon
   contains no NewIcon image, the standard image from the source icon is
   copied instead, as in the IMAGE option below.

IMAGE

   The standard image from the source icon is copied as the destination
   icon's NewIcon image.

REMAP

   The destination icon's own standard image is copied as the NewIcon image.

ERASE

    Any NewIcon image in the destination icon is erased.

If the NewIcons patch is installed, and the NEWICON tooltype is not set, the
default setting is to COPY the NewIcon image. Otherwise, default is KEEP.

Example:

NEWICON=REMAP

1.46 TOOLTYPES

                         3.3.3. ICONTYPE TOOLTYPE

With the ICONTYPE tooltype you can change the default setting for the type
of the destination icon when the icon is installed.

These settings can be changed while using IconInstaller with the
corresponding
                 Icon Type
                  menuitem.

There are eight possible options to this tooltype.

KEEP

    The destination icon type will be kept as it is.

COPY

    The type for the destination icon will copied from the source icon.

DISK

    The destination icon type will be set to 'Disk'.

DRAWER

    The destination icon type will be set to 'Drawer'.

TOOL

    The destination icon type will be set to 'Tool'.

PROJECT

    The destination icon type will be set to 'Project'.

TRASHCAN

    The destination icon type will be set to 'Trashcan'.

```
KICK
```

   The destination icon type will be set to 'Kick'.

If the ICONTYPE tooltype is not included at all, the default setting is to
KEEP the original icon type.

Example:

ICONTYPE=PROJECT

1.47 TOOLTYPES

                          3.3.4. DEFTOOL TOOLTYPE

The DEFTOOL tooltype is used to change the default setting for what
happens to the default tool of the destination icon.

These settings can be changed while using IconInstaller with the
corresponding
                  Default Tool
                   menuitem.

There are seven possible options to this tooltype.

KEEP

   The default tool in the destination icon is not changed.

COPY

   The default tool is copied from the source icon.

TOOL1

   Sets the default tool to 'MultiView'. Using the additional
                TOOL1
                    tooltype, 'MultiView' can be changed to some other preferred ←
                      tool.

TOOL2

   Sets the default tool to 'AmigaGuide'. Using the additional
                TOOL2
                    tooltype, 'AmigaGuide' can be changed to some other preferred ←
                      tool.

TOOL3

   Sets the default tool to 'Display'. Using the additional
                TOOL3
                    tooltype, 'Display' can be changed to some other preferred tool ←
                      .

```
TOOL4
```

> Sets the default tool to 'More'. Using the additional
> >   TOOL4
> > >   tooltype, 'More' can be changed to some other preferred tool.

```
OTHERTOOL
```

> Sets the default tool to the tool specified by the
> >   OTHERTOOL
> >   tooltype.

The default setting is to KEEP the original default tool, if the DEFTOOL
tooltype is not specified.

Example:

```
DEFTOOL=OTHERTOOL
```

1.48 TOOLTYPES

                          3.3.5. TOOLTYPES TOOLTYPE

With the TOOLTYPES tooltype you can change the default setting for what
will happen to the tooltypes of the destination icon.

These settings can be changed while using IconInstaller with the
corresponding
                Tool Types
                 menuitem.

There are two possible options to this tooltype.

```
KEEP
```

> The destination icon tooltypes are kept intact.

```
COPY
```

> The tooltypes are copied from the source icon.

If the TOOLTYPES tooltype is not set, the default setting is to KEEP
the original tooltypes.

Example:

```
TOOLTYPES=KEEP
```

1.49 TOOLTYPES

3.3.6. POSITION TOOLTYPE

With the POSITION tooltype you can change the default setting for the
position of the destination icon when it is installed.

The settings can be changed while using IconInstaller, by accessing the

Position
 menuitem in the 'Settings' menu, where the options are
described in detail.

There are three possible options to this tooltype.

KEEP

   The destination icon position is kept unchanged.

COPY

   The icon position is copied from the source icon.

FREE

   The destination icon position will be set free.

If the POSITION tooltype is not set, the default setting is to KEEP
the original icon position.

Example:

POSITION=FREE

1.50 TOOLTYPES

3.3.7. STACK TOOLTYPE

With the STACK tooltype you can change the default setting for what
will happen to the stack size in the destination icon.

These settings can be changed while using IconInstaller with the
corresponding
Stack
 menuitem in the 'Settings' menu.

KEEP

   The stack size is kept as it is in the destination icon.

COPY

The stack size is copied from the source icon.

If the STACK tooltype is not set, the default setting is to KEEP
the original stack size.

Example:

STACK=KEEP

1.51 TOOLTYPES

                        3.3.8. TOOL1 TOOLTYPE

The TOOL1 tooltype can be used to specify an alternative name instead
of 'MultiView' as the first preset default tool.

If the
                DEFTOOL
                 tooltype is set to TOOL1, this tool is used as the
default default tool.

If this tooltype is set, the name given will be shown instead of 'MultiView'
as an option under the
                Default Tool
                 menuitem in the 'Settings' menu.

Example:

TOOL1="IconInstaller"

1.52 TOOLTYPES

                        3.3.9. TOOL2 TOOLTYPE

The TOOL2 tooltype can be used to specify an alternative name instead
of 'AmigaGuide' as the second preset default tool.

If the
                DEFTOOL
                 tooltype is set to TOOL2, this tool is used as the
default default tool.

If this tooltype is set, the name given will be shown instead of 'AmigaGuide'
as an option under the
                Default Tool
                 menuitem in the 'Settings' menu.

Example:

```
TOOL2="GRAPHICS:FastJPEG"
```

1.53 TOOLTYPES

```
                    3.3.10. TOOL3 TOOLTYPE
```

The TOOL3 tooltype can be used to specify an alternative name instead
of 'Display' as the first preset default tool.

If the
                DEFTOOL
                 tooltype is set to TOOL3, this tool is used as the
default default tool.

If this tooltype is set, the name given will be shown instead of 'Display'
as an option under the
                Default Tool
                 menuitem in the 'Settings' menu.

Example:

```
TOOL3=Visage
```

1.54 TOOLTYPES

```
                    3.3.11. TOOL4 TOOLTYPE
```

The TOOL4 tooltype can be used to specify an alternative name instead
of 'More' as the first preset default tool.

If the
                DEFTOOL
                 tooltype is set to TOOL4, this tool is used as the
default default tool.

If this tooltype is set, the name given will be shown instead of 'More'
as an option under the
                Default Tool
                 menuitem in the 'Settings' menu.

Example:

```
TOOL4="PPMore"
```

1.55 TOOLTYPES

3.3.12. OTHERTOOL TOOLTYPE

The OTHERTOOL tooltype can be used to specify an alternative preset
default tool. The difference between the OTHERTOOL default tool and other
preset default tools is that this one can be changed with a requester
while using IconInstaller. This is done by accessing the 'Other Tool...'
option under the
                Default Tool
                 setting in the 'Settings' menu.

Setting this tooltype will provide an initial default tool to be used in
the requester. If this tooltype is not set, an empty string will be used
initially.

If the DEFTOOL tooltype is set to OTHERTOOL, this tool is used as the
default default tool.

Example:

OTHERTOOL="This Tool Can Be Changed While Using IconInstaller"

1.56 TOOLTYPES

3.3.13. ASKNEWICONS TOOLTYPE

This tooltype is used to facilitate the setting of the Image and NewIcon
options, if the
                NewIcons
                 patch is present. This tooltype is especially
useful in combination with the NOSETTINGS tooltype, as it gives the user
some control over the imagery installation even if the 'Settings' menu
is disabled.

A requester will be displayed when starting IconInstaller, asking the user
how the imagery should be installed. There are three options, 'NewIcon',
'Dot' and 'Standard'.
'NewIcon' means that the NewIcon image is copied from the source icon to
the destination icon, while the standard imagery is not changed.
'Dot' is the same as 'NewIcon', except the standard image is replaced
with a small dot.
Finally, 'Standard' means that the standard imagery is copied, and any
NewIcon imagery in the destination icon is erased.

If the NewIcons patch is not installed, this tooltype is ignored.

Example:

ASKNEWICONS

1.57 TOOLTYPES

3.3.14. SIMPLEREFRESH TOOLTYPE

By default, IconInstaller will open its window in smart refresh mode.
This means that the user interface will always be up to date. Setting this
tooltype will save some memory when running IconInstaller, but updating
the window will be a bit less efficient.

Example:

SIMPLEREFRESH

1.58 TOOLTYPES

3.3.15. NOSETTINGS TOOLTYPE

The NOSETTINGS tooltype is used to disable the
                Settings
                 menu. This
effectively means that the user cannot change the configuration while
using IconInstaller.

This is useful as a safety measure, to prevent novice users from
accidentally changing the settings, as some of the options could have
quite undesirable results. It is recommended to include this tooltype with
scripts that are included with distributed icon collections, to guarantee
that only the icon images will be changed by the script.

A useful companion to this tooltype is the
                ASKNEWICONS
                 tooltype. Using
this combination gives the
                NewIcon
                 user some control over the imagery
installation, even if the 'Settings' menu is disabled.

Example:

NOSETTINGS

1.59 REFERENCE

3.4. SCRIPT COMMANDS

This is a list of all keywords available when writing IconInstaller scripts.
As this section is intended for reference, it is advisable to read first the
introductory section about
                writing~scripts

.

The list on this page is only a brief overview; more detailed information
for each keyword is available by clicking on its name.

The keywords used for icon installation commands

                ~FROM~~~~~~~~~~~
                  Specify the name of the source icon

                ~TO~~~~~~~~~~~~~
                  Specify the name of the destination icon

The options available for icon installation commands

                ~ICONTYPE~~ ~~~
                  Explicitely set the type for the destination icon

                ~QUIET~~~~~ ~~~
                  Skip the destination icon if volume is not mounted

                ~FORCE~~~~~ ~~~
                  Create the destination icon if it doesn't exist

                ~MATCHTYPE~ ~~~
                  Compare source an destination icons for same type

                ~DIRECTORY~ ~~~
                  Install to a whole directory instead of a single icon

                ~RAMSCRIPT~ ~~~
                  Offer to write the Ram icon script to S:User-Startup

                ~CHECKSYS~~ ~~~
                  Search some default directories for the destination icon

The general commands

                ~MESSAGE~~~~~~~~
                  Display a message to the user

                ~INSTALLPALETTE~
                  Offer to install the specified palette to Workbench

                ~SCRIPTMODE~~~~~
                  Set the level of automation for the script

The options available for the general commands

                ~IFNOTNEWICONS~
                  Execute the command only if NewIcons is not installed

```
                ~IFNEWICONS~~~~
                  Execute the command only if NewIcons is installed
```

The remaining keywords and identifiers

```
                ~ICONINSTALLER~~~
                  The script identifier on the first line

                ~;~~~~~~~~~~~~~~~
                  The first character of a comment line
```

1.60   SCRIPT KEYWORDS

```
                3.4.1. INSTALLATION COMMAND : FROM KEYWORD
```

The FROM keyword is used to specify the source icon for an icon
installation command. The source icon is the icon that contains
the imagery to be installed. The FROM keyword must always be used
together with the
                TO
                  keyword, that specifies the destination icon.

A number of options may be used in icon installation commands,
to affect the installation procedure. The options are as follows.

```
                ICONTYPE

                QUIET

                FORCE

                MATCHTYPE

                DIRECTORY

                RAMSCRIPT

                CHECKSYS
                Examples:
```

FROM SourceIcon TO DestinationIcon

FROM SourceIcon TO DestinationDrawer/ DIRECTORY MATCHTYPE

1.61   SCRIPT KEYWORDS

```
                3.4.2. INSTALLATION COMMAND : TO KEYWORD
```

The TO keyword is used to specify the destination icon for the
installation command. The destination icon is the icon that will
receive the new imagery from the source icon, as specified by the
FROM keyword. The
                FROM
                  and TO keywords must always be used in
combination.

A number of options may be used in icon installation commands,
to affect the installation procedure. The options are as follows.

                ICONTYPE

                QUIET

                FORCE

                MATCHTYPE

                DIRECTORY

                RAMSCRIPT

                CHECKSYS
                If the DIRECTORY keyword is included in the command, the argument
for the TO keyword will be interpreted as a destination directory
instead of a destination icon.

Examples:

FROM SourceIcon TO DestinationIcon

FROM SourceIcon TO DestinationDrawer/ DIRECTORY MATCHTYPE

1.62 SCRIPT KEYWORDS

                3.4.3. ICONTYPE OPTION

With the ICONTYPE option, it is possible to change the type for
the destination icon. The type is given as the argument for the
ICONTYPE keyword. There are eight options:

 KEEP
 COPY
 DISK
 DRAWER
 TOOL
 PROJECT
 KICK
 TRASHCAN

KEEP means that the type is kept as it is in the destination icon, COPY
means that the type is copied from the source icon. The other options
set the type for the destination icon explicitely.

This option applies only to the single icon installation command it is
included with. If the ICONTYPE keyword is not included in a command, the
default type will be used. Normally, the default type is KEEP, but it can
be changed by setting the
                ICONTYPE
                 tooltype in the script's icon.

Examples:

FROM Icons/Disk TO SYS:Disk ICONTYPE DISK

FROM Icons/DrawerIcon TO ENVARC:Sys/def_drawer ICONTYPE=DRAWER FORCE

1.63 SCRIPT KEYWORDS

                        3.4.4. QUIET OPTION

Normally, if the destination icon (as specified with the TO keyword) is
on a volume or device that is not mounted, the usual 'Please insert
volume...' requester is displayed. If the QUIET keyword is included in
the command, the requester will be suppressed and the installation
command is simply skipped.

This is useful, for example, when installing new imagery to the hard
disk icons. Since it is often impossible to know beforehand how many
hard disks the user has, or indeed if they are mounted as HD0: or DH0:,
or whatever, non-existing volumes are conveniently skipped.

The QUIET keyword overrides the
                FORCE
                 keyword. Non-existing volumes
are skipped even if the
                SCRIPTMODE
                 is set to MANUAL.

Examples:

FROM HardDiskIcon TO HD0:Disk QUIET FORCE TYPE=DISK

FROM MusicHD TO MUSICHD:Disk QUIET

1.64 SCRIPT KEYWORDS

                        3.4.5. FORCE OPTION

Normally, if the destination icon cannot be found in an installation

command, the command is simply skipped. If the FORCE option is
included, IconInstaller will try to create the new icon from scratch.
In this case, an appropriate default icon will be used for those
attributes that are not copied from the source icon. If an accompanying
file exists for the icon to be created, it is taken into account when
selecting what type of default icon to use.

If a certain type of the icon is preferred, the
                    ICONTYPE
                     keyword can
be included as well to explicitly set the icon type.

Examples:

FROM Icons/HardDisk TO HD0:Disk FORCE QUIET ICONTYPE=DISK

FROM Icons/Icon TO SYS:ThisIconWillBeCreated FORCE

1.65 SCRIPT KEYWORDS

                    3.4.6. MATCHTYPE OPTION

The MATCHTYPE option is used when it must be certain that the source icon
and the destination icon are of the same type.

This keyword is especially useful in combination with the
                    DIRECTORY
                    keyword, if the imagery from the source icon should only be ←
                       installed to
certain types of icons in the destination directory, and it is possible
that the directory may contain many types of icons. Of course, the
MATCHTYPE keyword can be used without the DIRECTORY keyword, too.

In addition, if the type for the destination icon is specified with
the
                    ICONTYPE
                     keyword, the type(s) of the existing destination icon(s)
are matched against that type, instead of the type of the source icon.

This option can be used, for example, to change the imagery for drawer
icons only in a given directory, even if the directory would contain
other types of icons as well.

Examples:

FROM Icons/DrawerIcon TO SYS: DIRECTORY MATCHTYPE

FROM Icons/ProjectIcon TO SYS:FooBar ICONTYPE=PROJECT MATCHTYPE

1.66 SCRIPT KEYWORDS

3.4.7. DIRECTORY OPTION

The DIRECTORY option is used when the same imagery is meant to be
installed to all icons in a given directory, but it is not possible to
know in advance what these icons will be, or even how many they will be.

For example, icons in the SYS:Prefs/Presets/ drawer, the DataType or
DOSDriver icons, or perhaps icons in some Graphics/Pictures/ drawer are
good candidates for the use of this option.

In this case, the argument for the TO keyword is interpreted as the
destination directory instead of a single icon. The imagery from the
source icon will be installed to all icons in this directory.

This keyword can be used in combination with the
                FORCE
                 keyword, if
new icons should be created even for files that don't have any
accompanying icon (.info) file. The
                MATCHTYPE
                 keyword can also be
used to automatically discard those icons in the destination drawer
whose type doesn't match that of the source icon.

A shorter version DIR is also available as a synonym for DIRECTORY.

Examples:

FROM Icons/PresetIcon TO SYS:Prefs/Presets/ DIR FORCE TYPE=PROJECT MATCH

FROM Icons/DevsIcons/DataType TO DEVS:DataTypes DIRECTORY QUIET

1.67 SCRIPT KEYWORDS

3.4.8. RAMSCRIPT OPTION

The RAMSCRIPT keyword is a handy option when installing a new icon for
the Ram Disk. As the Ram Disk is created from scratch after every reboot,
the new icon must be created as well.

For this purpose, a copy of the icon must be saved to the user's system
disk, from where it can be copied to the Ram Disk every time the computer
is powered up. The copying process can be made automatic by appending a
small script to the 'S:User-Startup' file (that is executed automatically
at every reboot).

This option applies only to the single icon installation command it is
included with. Furthermore, this option is ignored if the user decides
to skip this icon.

If the user decides to install the new icon for the Ram Disk, a requester
will be displayed, explaining the special case with the Ram Disk icon,

and offering to perform the actions described above to make the new icon
appear automatically after every reboot. The user can select from 'OK' or
'Cancel'.

If the
                  SCRIPTMODE
                   is set to AUTOMATIC, the requester will not be
displayed. instead, IconInstaller will try to perform the installation
automatically.

It is recommended to use the RAMSCRIPT option in combination with the

                  FORCE
                   and
                  ICONTYPE
                  =DISK options, as this will guarantee that the icon
will really be created, and that it will actually be a disk type icon.

The shorter RAM keyword can be used instead of RAMSCRIPT.

Examples:

FROM Icons/RamDisk TO RAM:Disk RAM FORCE ICONTYPE=DISK

FROM Icons/Disks/Ram TO RAM:Disk RAMSCRIPT

Technical details:

A duplicate icon for the Ram Disk will be created as
'ENVARC:Sys/def_ramdisk.info'. Then, the following script
will be appended to the 'S:User-Startup' file.

```
;BEGIN IconInstaller
If EXISTS ENVARC:Sys/def_ramdisk.info
   Copy ENVARC:Sys/def_ramdisk.info RAM:Disk.info
EndIf
;END IconInstaller
```

IconInstaller will detect if the script has already been included
in the 'S:User-Startup' file, in which case the script will not
be appended a second time.

1.68 SCRIPT KEYWORDS

   3.4.9. CHECKSYS OPTION

The CHECKSYS option can be included with icon installation commands
to search some default directories on the SYS: volume for the
destination icon. This is especially useful when installing icons
for standard Workbench files, that may have been moved to somewhere
else from their original location.

For example, if the user has moved the RexxMast program from the
SYS:System drawer to the SYS:WBStartUp drawer, the script will still

find it with the command

```
FROM Icons/RexxMast TO SYS:System/RexxMast CHECKSYS
```

First, IconInstaller will try to find the icon in the specified
directory for the destination icon. If the icon is not found there,
it will also look in the following directories (in this order)

```
 SYS:
 SYS:WBStartUp
 SYS:System
 SYS:Tools
 SYS:Utilities
```

Example:

```
FROM Icons/System/Shell TO SYS:DOSTools/Shell CHECKSYS
```

```
FROM Icons/Tools/Commodities/Blanker TO SYS:Blanker CHECKSYS
```

1.69 SCRIPT KEYWORDS

3.4.10. MESSAGE COMMAND

The MESSAGE command is used to display messages for the user. The
actual text to be displayed must be included, in quotes, after the
keyword itself. Messages up to about 500 characters are supported.
The special character combination '\n' is interpreted as a newline
character.

A requester will be displayed, containing the message, with a single
'OK' gadget.

The
               IFNEWICONS
                and
               IFNOTNEWICONS
                keywords can be used in combination
with this command.

Examples:

MESSAGE "This will be all on the same line."

MESSAGE "This text\nwill be split\non three lines."

1.70 SCRIPT KEYWORDS

3.4.11. INSTALLPALETTE COMMAND

As standard Amiga icon files don't contain any colour data, they
will often need a specific palette for the Workbench screen to
make the icons appear correctly. This is especially true with eight
colour icons. The INSTALLPALETTE command is used for this purpose.

The INSTALLPALETTE keyword must be followed with the name of the
palette file to be installed. The file can be either an IFF ILBM
file with a CMAP chunk or an IFF PREF palette file. The former is
created by most Amiga paint programs when you save a palette, the
latter is created with the Palette preferences editor, by selecting
the 'Save As...' menuitem.

A requester will be displayed, informing the user that the set of
icons to be installed requires a custom palette to make the icon
colours appear correctly. The requester prompts the user to select
from three options; 'Save', 'Use' and 'Cancel'.
'Save' means that the new palette will be installed, and will be
retained even if the computer is powered down.
'Use' means that the palette will be installed, but after the
next reboot, the original palette will be restored again.
'Cancel' means that the palette is not changed at all.

If the
                SCRIPTMODE
                 is set to AUTOMATIC, the requester will not be
displayed at all. Instead, the 'Save' option will be selected
automatically.

The
                IFNEWICONS
                 and
                IFNOTNEWICONS
                 keywords can be used in combination with
this command.

The PALETTE keyword can be used as a synonym for INSTALLPALETTE.

Examples:

INSTALLPALETTE IGfx_Palette

PALETTE "My Icon Palette"

Technical details:

This command will try to execute the SYS:Prefs/Palette program (or the
PREFS:Palette program, if the former isn't found), giving the specified
palette file and the selected action (ACTION=SAVE or ACTION=USE) as
arguments. If neither PPREFS:Palette or or SYS:Prefs/Palette are found,
an error message will be displayed.

1.71 SCRIPT KEYWORDS

3.4.12. SCRIPTMODE COMMAND

This command is used to specify the level of automation for the script.

There are three possible options for this command:

AUTOMATIC

   As the name implies, every action is made as automatic as possible.
   All icon installation commands (FROM .. TO) are executed automatically.
   If the destination icon (as specified by the TO keyword) is not found,
   the command is simply skipped.
   If the
                  RAMSCRIPT
                   keyword is encountered as part of an installation
   command, IconInstaller will try to append a small script for Ram Disk
   icon creation to the S:User-Startup file without displaying any
   requester.
   If the
                  INSTALLPALETTE
                   command is encountered, no requester will be
   displayed, and the 'Save' option will be selected automatically. Of
   course, any IFNEWICONS or IFNOTNEWICONS keyword is also taken into
   account.
   Messages invoked by the
                  MESSAGE
                   command and disk based error
   messages will be displayed as usual.

SKIP

   This is the default mode used by default, if SCRIPTMODE is not specified
   at all. In this mode, the user has to confirm all icon installation
   (FROM .. TO) commands, before the script can proceed, either by
   installing or skipping the icon. However, if the destination icon (as
   specified by the TO keyword) is not found, the command will be skipped
   as in the AUTOMATIC mode.
   The RAMSCRIPT and INSTALLPALETTE requesters will be displayed as usual,
   forcing the user to make the choice manually in each case.
   All other messages are displayed as well.

MANUAL

   This mode gives the user the best manual control. The only difference
   from the SKIP mode is that even those icon installation (FROM .. TO)
   commands where the destination icon is not found have to be confirmed
   by the user. This gives the user the option to open the destination
   icon manually, or confirm the skip and proceed with the script. This
   mode could be handy for icon collections where the author wants the
   script to display every icon but can't know in advance where the
   destination icons will be located.

The SCRIPTMODE keyword can be included many times within a script,
changing the level of automation in the middle of a script.

The
                    IFNEWICONS
                     and
                    IFNOTNEWICONS
                     keywords can be used in combination with
this command.

You can also use the shorter version MODE as a synonym for SCRIPTMODE.

Examples:

MODE AUTOMATIC

SCRIPTMODE=MANUAL

1.72 SCRIPT KEYWORDS

                        3.4.13. IFNOTNEWICONS OPTION

The IFNOTNEWICONS keyword, (like its counterpart,
                    IFNEWICONS
) is a
simple way to control the execution of the
                    MESSAGE
                    ,
                    INSTALLPALETTE
                    ,
and SCRIPTMODE
patch is installed on the user's computer or not.

If this keyword is present, the command will be executed if (and only if)
the NewIcons patch is NOT installed. If the NewIcons patch is installed,
the command is ignored. The command will also be executed if the
                    NewIcon
                    configuration is set to 'Erase', even if the NewIcons patch is ←
                        installed.

This is useful, for example, to display different messages to NewIcon users,
or to skip the palette installation.

This keyword cannot be used with an icon installation (FROM .. TO)
command.

Examples:

INSTALLPALETTE "IGfx_Palette" IFNOTNEWICONS

IFNOTNEWICONS MESSAGE "Hmmm...\n I see you are not using NewIcons!"

1.73 SCRIPT KEYWORDS

                              3.4.14. IFNEWICONS OPTION

The IFNEWICONS keyword (like the
                    IFNOTNEWICONS
                     keyword), is used as
a simple way to control the execution of the
                    MESSAGE
                    ,
                    INSTALLPALETTE
                    ,
and
                    SCRIPTMODE
                     commands depending on if the script is executed on
systems with the
                    NewIcons
                     patch installed or not.

If this keyword is included, the actual command is executed if (and
only if) the NewIcons patch IS installed. IF the patch is NOT installed,
the command will be skipped.

One use of this command could be to display display certain messages only
to NewIcon users.

This keyword cannot be used with the icon installation (FROM .. TO)
commands.

Examples:

IFNEWICONS MESSAGE "You are using the NewIcons system."

MESSAGE "Can be after the command keyword too" IFNEWICONS

1.74 SCRIPT KEYWORDS

   3.4.15. ICONINSTALLER KEYWORD - SCRIPT IDENTIFIER

The first line of each script has to start with the keyword

ICONINSTALLER

so that IconInstaller will know that this text file is meant as
a script. If you try to start a script and get the error message
'File '...' is not a valid IconInstaller script', this keyword is
missing from the first line.

Optionally, a version number may be added after the keyword.
This guarantees that the script is only executed on a version of
IconInstaller that supports all commands that are used in the
script. If someone tries to execute the script on an older version,
an error message is displayed. If no version number is given, any
IconInstaller version will do. Version 3.0 can be assumed to be the

minimum version possible.

Examples (each must be the first line of a script):

ICONINSTALLER

ICONINSTALLER 3.1

1.75 SCRIPT KEYWORDS

3.4.16. SCRIPT COMMENT CHARACTER

The semi-colon (';') is used as a comment identifier in IconInstaller
scripts. The semi-colon must be the first character on the line.
IconInstaller does not currently support comments starting from
the middle of a line. Commands and comments must be on separate lines.

Lines that start with the semi-colon are ignored by IconInstaller.
Empty lines are ignored as well, and can be used for extra spacing
to improve script readability.

Examples:

;This line is a comment

 ;This line will produce an error message

1.76 MISCELLANEOUS

4.   MISCELLANEOUS

This section contains legal information, description about previous
IconInstaller versions, plans for future versions, some information
about NewIcons, and about the author, too.

4.1. Copyright

4.2. Disclaimer

4.3. History

4.4. Future

4.5. NewIcon Information
  4.6. Author

1.77 MISCELLANEOUS

4.1. COPYRIGHT

IconInstaller is freely distributable and copyright © 1995 by Tom Ekström.
There is no charge for using this software. This is generally known as
FreeWare.

IconInstaller can be distributed in two ways:

1. As the complete IconInstaller archive.

   The original archive must be distributed in its entirety, without
   additions, deletions or modifications of any kind. The original archive
   contains the following files:

   ```
   IconInstaller               IconInstaller.info
   IconInstaller.guide         IconInstaller.guide.info
   Install_Icons               Install_Icons.info
   Icons.info                  Icons/def_harddisk.info
   Icons/def_preset.info       Icons/def_ramdisk.info
   Icons/IGfx_Palette          Icons/IGfx_Palette.info
   Icons/PrefsDrawer           Icons/PresetDrawer
   Icons/Font.info             Icons/IControl.info
   Icons/Input.info            Icons/Locale.info
   Icons/Overscan.info         Icons/Palette.info
   Icons/Pointer.info          Icons/Printer.info
   Icons/PrinterGfx.info       Icons/PrinterPS.info
   Icons/ScreenMode.info       Icons/Serial.info
   Icons/Sound.info            Icons/Time.info
   Icons/WBPattern.info        Redistribution.info
   Redistribution/IconInstaller.guide
   Redistribution/IconInstaller.guide.info
   ```

2. As an installation tool with icon collections.

   Icon authors may include IconInstaller with their freely distributable
   icon collections. In this case, the following files must be included:

   ```
   IconInstaller       IconInstaller.guide
   IconInstaller.info  IconInstaller.guide.info
   ```

   It is recommended to include the 'light' version of IconInstaller.guide
   from the 'Redistribution' drawer instead of the full version as most
   users don't need the script writing and the reference sections anyway.
   Optionally, the icon for the script ('Install_Icons.info') may be
   included as well. Tooltypes in the icon files may be changed, otherwise
   the files may not be altered in any way.
   IconInstaller may not be distributed separately from its documentation
   (IconInstaller.guide or Redistribution/IconInstaller.guide) under any
   circumstances.

In addition, the following two conditions must be met:

No profit may be made for distributing IconInstaller. A reasonable handling
and copying charge is acceptable.

No part of IconInstaller may be included in a commercial product without
my prior written consent. No prior permission is required to distribute
IconInstaller on Internet/FTP (ie. Aminet), public access BBS's, FidoNet,
and the Aminet, Fish series and other CD-ROMs containing only freely
distributable software. IconInstaller can also be included on a magazine
coverdisk, provided that a copy of that particular issue is sent to me.

If you want to send any feedback, complaints, bug reports etc., feel free
to send
                me
                 a note.

1.78 MISCELLANEOUS

   4.2. DISCLAIMER

THE AUTHOR MAKES NO WARRANTIES, EITHER EXPRESSED OR IMPLIED, WITH
RESPECT TO THE PROGRAM DESCRIBED IN THIS DOCUMENTATION, ITS QUALITY,
PERFORMANCE OR FITNESS FOR ANY PARTICULAR PURPOSE. THE PROGRAM IS
DISTRIBUTED "AS IS". THE ENTIRE RISK AS TO ITS QUALITY AND PERFORMANCE
IS WITH THE USER. IN NO EVENT WILL THE AUTHOR BE LIABLE OF DIRECT,
INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGE TO OR LOSS OF DATA OR
EQUIPMENT RESULTING FROM THE USE OF THIS PROGRAM.

1.79 MISCELLANEOUS

   4.3. VERSION HISTORY

Version 3.1 - September 7, 1995
             - First stand-alone release
             - Corrected bug with INSTALLPALETTE script command
               'Instert volume PREFS:' was displayed if SYS:Prefs/Palette
               wasn't found and PREFS: wasn't mounted
             - Corrected bug with MutualExclude menuitems
               Would clear even non-existing menuitems
             - Corrected bug when opening NewIcons
               Tried to RemapChunkyImage() even when there was no chunky
               image
             - Corrected bug with script version checking
               New version more robust, revisions now handled correctly
             - Corrected nasty bug when checking current DOS version
               Horrors! Did not work at all. Probably crashed the program
               on all AmigaDOS versions previous to 3.0.

Version 3.0 - August 22, 1995
             - Full NewIcons support:
               NewIcon options in the 'Settings' menu
               Conversion from standard icons to newicons, and vice versa
               NEWICON and ASKNEWICONS tooltypes

```
                    IFNEWICONS and IFNOTNEWICONS script keywords
              - Position settings and tooltypes
              - new SCRIPTMODE script command
              - MESSAGE script command extended to 500 characters
              - Configuration for a script is now read from the script's icon
                tooltypes instead of IconInstaller's icon
              - Blocking requesters, for WB 3.0+ with busy pointers
              - TOOL1, TOOL2, TOOL3 and TOOL4 tooltypes
              - Other Tool is can now be requested using the 'Settings' menu
              - 'Smart' filename handling in the destination icon gadget

Version 2.1 - June 13, 1995
              - Released with Iconographics 1.1 and 1.2
              - Corrected bug with the FORCE keyword in scripts
                Icons were not really created (eg. Ram Disk and def_xxx icons)

Version 2.0 - May 26, 1995
              - Released with Iconographics 1.0
              - Rewritten in Modula-2
              - Fixed (hopefully) a nasty NewLookMenus bug that made
                the menu text invisible on Workbench 2.x systems.
              - Fixed bug with QUIET (MUTEVOLUME) script keyword that appeared
                in combination with directory installation
              - New 'Settings' menu
              - Default configuration in icon tooltypes

Version 1.0 - 1994
              - Released with IconPack 1.0 as Iconizer
              - Coded in assembler
```

1.80 MISCELLANEOUS

```
                    4.4. FUTURE
```

Here's a list of some improvements that I've thought of, but haven't had the
time to implement yet (in no particular order):

- Import imagery from an IFF brush
- Optional custom palette for NewIcon conversion instead of screen palette
- Resizable GUI
- CLI interace
- ReqTools.library support (making Asl optional)

If you think of something else that you would find useful, let
                me
                 know.
Don't ask for an icon editor though, as there already exist other tools for
that purpose. And I think it's much easier to design icons in a paint
program anyway.

1.81 MISCELLANEOUS

4.5. NEWICON INFORMATION

NewIcons is a new standard of icon (.info) files that makes icons look
good independently of the Workbench palette. This is accomplished by
including new icon imagery with colour data in the icon tooltypes, and
using a system patch to read this NewIcon imagery and display it instead
of the standard icon image. When the patch is in use, the NewIcon image
information is also hidden from the icon tooltypes.
In other words, NewIcons contain both standard and NewIcon imagery. When
the patch is not in use, the standard imagery is displayed as usual.

Because the NewIcon images contain colour information, they are remapped
to the current Workbench screen colours, and will therefore look good
on a screen with any palette.

To be able to use NewIcons, you'll need the NewIcons patch and the
accompanying newicon.library. These are included in the NewIcons package.
The package is not included with IconInstaller, but can be obtained from
eg. Aminet (pix/icon/NewIcons.lha), as well as good PD/Shareware suppliers.

The NewIcons standard is developed by Nicola Salmoria.

1.82 MISCELLANEOUS

4.6. AUTHOR

I'm currently studying at Tampere university in Finland, majoring in Library
and Information science.

I can be reached by

Mail   : Tom Ekström
         Tiilentekijänkatu 16 as.10
         FIN-20810 Turku
         FINLAND (Europe)

E-mail : litoek@uta.fi

I reply to all e-mail I get. The e-mail address will be valid until autumn
1996, the street address is valid long into the future.