

PubScr

Hans Bühler

COLLABORATORS

	<i>TITLE :</i> PubScr		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Hans Bühler	August 9, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	PubScr	1
1.1	PubScr1.0 Guide (w)26.1./25.2.1994 by Hans Bühler	1
1.2	Create a public screen of your own !	1
1.3	Requirements & Installation	2
1.4	How to start up PubScr...	2
1.5	PubScr on the run...	3
1.6	The Use of the settings window	4
1.7	Settings window menus	5
1.8	Defining the screen values:	5
1.9	Defining the control window	7
1.10	Setting a font	7
1.11	Defining the shell window and the startup-file	7
1.12	How to use the reqtools-file-requester	9
1.13	How to use the reqtools-font-requester	10
1.14	How to use the reqtools-screen mode-requester	11
1.15	Settings gadgets that invoke basic functions:	13
1.16	Looking at this file by menu: View doc	13
1.17	PubScr icon handling:	14
1.18	Gadgets used within PubScr:	14
1.19	Public screens - a new OS2.0 feature !	14
1.20	PubScr again and again...	15
1.21	Copyright (c)1994 by CodexDesign:	16
1.22	Amiga on its way to paradise	18

Chapter 1

PubScr

1.1 PubScr1.0 Guide (w)26.1./25.2.1994 by Hans Bühler

·C·O·D·E·X· ·D·E·S·I·G·N· ·S·O·F·T·W·A·R·E·
presents

PubScr1.01 - "create a public screen of your own !"

This guide should give you further information about what this program does, how to use it and about all I want you to know.

Select one of the following topics:

Introduction
System requirements & installation
Start PubScr
Play with PubScr
Settings window
Past & future
About PubScr, the author and some copyrights

In fact, I suppose, PubScr is mostly easy to understand, but who should know ?

Have fun, anyhow - Hans Bühler
Codex Design Software.

1.2 Create a public screen of your own !

PubScr concept:

Approximately two weeks ago I started a program which allows the user to define the public screen on which it should open its windows. But as I started to watch out for a program that opens a user-defined public screen, I wondered:

Not much programs CAN create public screens - and most of them just clone the workbench screen's preferences.

Thus I decided to develop a program which just opens a public screen - but every kind one may ever need !

Using PubScr you will be able to set up all types of public screens you have ever dreamt of - a SUPERHIRES/LACED 417x317x5 screen with times/65 font etc. is now possible !

All system-known possibilities are available !

I really hope, someone finds this program useful - I do !

1.3 Requirements & Installation

Requirements:

PubScr runs under Kickstart 2.04 (V37) and higher (coded with Kick3.0), won't need any hardware extras but expects the following libraries:

diskfont.library	Amiga OS
amigaguide.library	Amiga OS
gadtoolsbox.library	PD, (c)Jaba Development
reqtools.library	PD, (c)Nico François

Installation:

Double-Click at 'PubScr.install' to install PubScr using the Commodore System Installer. If you want to do it by hand, follow these points:

1. Copy the four libraries mentioned above from '<pubscr>libs/' to LIBS:..
2. Copy the file icons/DefPubScr_project.info into a directory called 'ENVARC:PubScr'. You may change this icon because my ability in creating nice graphics is not that wonderful !
3. Copy the file 'PubScr' into a directory of your own choice, e.g. 'SYS:C/', 'SYS:System/', ...
You may also copy the icon file PubScr.info into the same directory.
4. Copy this file, 'PubScr.guide'~into one of your document directories (e.g. 'DOC:AmigaGuide/') and use the AmigaDOS command 'SetEnv' to help PubScr to find this file:
Type 'SetEnv ENVARC:pubScr/HelpFile DOC:AmigaGuide/pubscr.guide' or something else.
You can also locate this file later while PubScr runs.
5. Now run PubScr and pray.
Note the About PubScr... chapter !

1.4 How to start up PubScr...

You want to start PubScr ? Choose one of these possibilities:

Direct workbench start:

Double-click on its icon.

PubScr will start, look for a tooltype 'STDSET=<set filename>', load the <set filename> - if possible - and will bring up the screen. If STDSET had not been defined, PubScr will use its defaults.

Indirect workbench start:

Double-click onto one of PubScr's project-icons.

CLI/Shell start:

Call it from the CLI by 'PubScr <set filename>'.

If you don't pass a <set filename>, PubScr will behave as described under point a).

Note: PubScr does NOT detach itself from the CLI, so you may better start it by 'RUN >NIL: <NIL: PubScr <...>' !!

One good reason for another version !

Now PubScr is up !

1.5 PubScr on the run...

PubScr main window

Once started, PubScr opens a screen (your 'very own' public screen !!!!) and one window on it. By default, this is a little window in the upper left corner of the screen.

The window is equipped with some menus presenting these items:

---- Project ----

Settings:

See settings.

Information:

Opens up a requester which informs the user about some VERY VERY important things.

Exit:

just guess ! - allows to close the screen. By definition, you are not able to close a public screen when there are visitor windows left on it !!!!

Thus, close all alien windows before you close the screen. You need to close all windows, when you want to set up new preferences, too !

---- Utility ----

Open shell:

Opens a shell window and executes 'S:Shell-startup'. Note the chapter about the shell-window settings.

Run startup:

If the startup file is NOT defined as NONE, you can use this item to execute the startup file/command again.

---- Help ----

View doc:

See the ViewDoc chapter.

Moreover the window has several (system) gadgets (only if you use a not-backdropped window !):

CLOSE Gadget (upper left corner of the window):

works like the 'Exit'~item.

SIZING Gadget (lower right corner of the window):

allows you to size the window (see Amiga docs).

ZIP Gadget (to the left of the window title):

minimizes the window's size (see Amiga docs).

DEPTH Gadget (upper right corner of the window):

places the window in front/behind others (see Amiga docs, too).

DRAGBAR (the title field):

allows you to move the window within the screen (yes, see Amiga docs).

1.6 The Use of the settings window

The settings window looks like this (menu/Settings...):

Click at any gadget you are interested in:

```

+-[Menu]-----+-----+
+-----+-----+
|                                     | | |
| PubScrName: [.....] | [-Undo all--] |
|   ScrTitle: [.....]
| [Screen] | [---Reset---] |
|           [x] Make default pubscreen | |
|           [x] Exit when last window is been closed | |
|           [x] New window on pubscreen pops screen to front | |
|           [x] Open wb windows on def.pubscreen (SHANGHAI) | |
|           [x] Close workbench | |
|                                     | |
|   WinTitle: [.....] [.....] | |
|     WinX: [....] WinWid: [....] [....]
| [--Font---] | |
|     WinY: [....] WinHi: [....] | |
|           [x] Backdrop [x] BackFill [x] System default | |
|                                     | [Save deflt.] |
| ShellWin: [.....] | [----Use----] |
|   Startup: [O·None·][.....][Get] | [---Cancel--] |
|                                     | |
+-----+-----+

```

These types of gadgets are used: Click at one to get information on how

it works:

```
[----Txt----] Action gadget (boolean gadget)
[.....] Text gadget (string gadget)
[0...Txt....] Cycle gadget
[x] Checkbox gadget
```

1.7 Settings window menus

The settings window is been equipped with these menus:

---- Project ----

Load:

Opens a file-requester which you can use to load a previously saved set.

Save:

Save the current set (if you have already saved it before, i.e. it has already a 'name').
Otherwise see 'Save As'.

Save As:

Opens a file-requester which you can use to save the current set somewhere you like.
An icon will be created. 'default tool' will be set automatically.
If you change the location of 'PubScr', the tool-type will probably not match any further. Then, load all sets you can remember and saved them back again. Their tool-types will be set correctly, then.

Set default:

Opens a file-requester. Select the set you want to run by default. It will be stored.

Delete:

Opens a file-requester. Select the file set you want to delete. You better delete PubScr-sets (postfix '.pub') only, because these extension '.pub' will be added, automatically.

Cancel:

Abort the settings window without any changes.

---- Help ----

View Doc:

See the ViewDoc chapter.

1.8 Defining the screen values:

PubScrName:

Used to enter the name of the public screen. Other programs will need this name to refer to your screen. Enter this name, once given, in each application (which support public screens) you want to open its windows on the new screen.

ScrTitle:

The Title of the screen shown in its top bar (menu-bar), when the PubScr control window active.

Screen:

Open a little requester to define the screen's display.

Make default pubscreen:

Some programs do not offer to open their windows on a specific public screen but on the 'default' public screen. Newer programs often do always open on the default pubscreen, which is used to be the workbench screen.

Set this flag to have all windows usually opening on the 'default' pubscreen on your own one.

Exit when last window is been closed:

Set this flag if you want your screen to be closed if the last visitor window is been closed.

Example: CygnusEd V2.12 can't be opened with a non-standard screen like 'productivity'. It always uses the HIRES mode.

But the editor can use the workbench screen.

Set flag 'Open windows on... (SHANGHAI)' (see below) and the flag we are talking about.

Run the editor.

It will open its window on this screen with your very own settings.

If you leave it now, the screen won't remain unused but will be closed, too.

New window on pubscreen pops screen to front:

By default, when a window opens on any public screen 'behind' the actual one, the screen will remain in the back.

Set this GLOBAL SYSTEM flag to pop a public screen to front, if a windows opens on it.

This flag is GLOBAL, i.e. it strikes ALL public screens around.

Open wb windows on def.pubscreen (SHANGHAI):

Some old programs do not open their windows on the 'default' public screen (see above), but on the workbench screen.

If you set this flag, those windows will also be put on the default public screen.

This is a GLOBAL SYSTEM flag !! I.e. when your screen is not the default public screen (see above), these windows will open on the default pubscreen, not on yours !

This flag had been called 'SHANGHAI'~by Commodore BM !

Close workbench:

Set this flag if you want to close the workbench during running PubScr 1.0. The workbench will automatically be opened again, when your public screen closes.

Might be useful to save some memory and for some tests. It would be wise to use the flags "SHANGHAI" and "Make default pubscreen", too, if you want to close the workbench.

1.9 Defining the control window

WinTitle:

The title of the PubScr control window opened on the public screen.
Unused when you want a backdrop-window.

WinX: WinY:

Define the upper left corner of the PubScr control window.
Unused when you want a backdrop-window.

WinWid: WinHi:

Dimensions of the control window.
Unused when you want a backdrop-window.

Backdrop:

Select this mode, if you want to put a backdropped window on your screen. That means it won't have any border graphics.

BackFill:

Fill the back of the window with a little pattern.

1.10 Setting a font

Upper text gadget (FontName):

Name of the font you want to have as screen/window border font.
Unused when you want the system default font.

Lower text number gadget (FontHeight):

Height of the font you want to use. If the system can find the font but not with the requested height, it will be generated.
Unused when you want the system default font.

Font:

Opens a font-requester using it you can select the font you want to use by mouse.
Unused when you want the system default font.

System default:

Select this option if you want to use the default system font you may set using the preferences 'Fonts' program.
In general, this would be topaz.font/8, I guess.

1.11 Defining the shell window and the startup-file

ShellWin:

Enter the window or output-file definition for the shell you are able to open by the control window's menuitem Open shell....

Moreover, this window/file definition is used as output/input stream when you are using a 'script' startup file (see below).

The shellwindow definition should always be a CON: file definition, like:

```
"CON:0/0/200/100/ Hello here I am /CLOSE/PUBSCREEN %s"
```

```

|                                     |
the first four numbers are the x,y,width and height values
of the window to be opened.      |

```

Impossible values will be changed to values that make the window fit on your screen (e.g. use a width of -1 to have a shell window that is always that wide like your screen).

```

|                                     |
Behind the coordinates, a window title is
expected.                             |

```

Additionally, some flags can be passed to the definition:

Flags for CON: definitions, seperated by '/':

CLOSE: Add a `closewindow-gadget` to your shell window. Thus, you can close the window by clicking at it, later.

ALT x,y,wid,hi: Can be used to define the alternate window position when clicking at the ZIP-gadget (second gadget from the right in the top border).

PUBSCREEN <title>: Opens the window on the pubscreen <title>. PubScr allows you to enter a '%s'~for the <title> and will replace it by your public screen's title.

Thus, it should always used the way I used it above.

For other flags, please refer to the Amiga documentations.

Startup:

Toggle between 'None' : Do nothing when starting up.

```
'Script' : The startupfile is meant to be an
          AmigaOS batch file and will be
          executed when starting up PubScr.
          PubScr will NOT close the window it
          opens for input/output itself.
          If you want the window to be closed,
          add a 'EndCLI' command at the end
          of the script file.
```

'Command': The startupfile is a command string like typed from the Shell. Will be run when starting up.

Startupfile string gadget (lowest string gadget in the window):

Enter the startup file's path and name here. If you're running a command from the startup code, add options for that command, if you want.

Get :

Open a file-requester used to locate the startupfile.

1.12 How to use the reqtools-file-requester

This file-requester is part of the reqtools.library by Nico François:
It should look like this:

```

Requester title
+-----+
|                                     | |
|                                     | |
|                                     | |
|                                     | |
|   FileList                         | |
|                                     | |
|                                     | |
|                                     |#|
|                                     |Scrollbar
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |-|
|                                     |^Arrow up
|                                     |-|
|                                     |vArrow down
+-----+

Pattern:      [(Patternstring)#?.....]
[(DirectoryName).....] [-Get-]
[(FileName).....] [.info]

[OKAYTEXT] [Volumes] [Parent] [Cancel]

```

Requester Title:

Gives information about what you are doing and what you are expected to do.

FileList:

The list of files and directories found. Move through by the scrollbar and the two arrows on its right hand side.

DirectoryName:

Text gadget for the name of the directory listened.

FileName:

Text gadget for the name of the selected file.

Patternstring:

Text gadget. Only files which matching names will be displayed in the FileList.

Get:

Button to read directory again. Useful when you change something

```

while the requester is up.

.info:
    If pressed, icons (<file>.info) will also be shown.

Volumes:
    Get a list of all mounted devices and assigns.

Parent:
    Goto parent directory (the directory underneath).

Cancel:
    Exit requester and cancel action.

[OKAYTEXT]:
    The text depends on the context from which the requester had been
    called. In general, this gadget accepts the selection of the
    current DirectoryName and FileName.

```

1.13 How to use the reqtools-font-requester

When you click at the 'Font'~gadget in the settings window, this requester will appear:

```

Select screen font...
+-----+-----+
|                                     | |
|                                     | |
|                                     | |
|                                     | |
|      FontList                      | |
|                                     | |
|                                     | |
|                                     |#|
|                                     |Scrollbar
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |-|
|                                     |^Arrow up
|                                     |-|
|                                     |vArrow down
+-----+-----+

[ (FontName) ..... ] [ (FontHi) ]

+-----+
| Font example      |
+-----+

Bold [x]      Italic [x]      Underline [x]

```

[-Use-]

[Cancel]

FontList:

List of all fonts found. Move through by the Scrollbar and the two arrows on the right hand side.

FontName:

Name of the font you selected.

FontHi:

Height of the font you selected. If the system is able to open the font but not that high, a new font will be generated.

Font example:

a little text written using the font you have selected.

Bold: Italic: Underline:

some style flags.

Use:

use the font displayed in 'Font example'.

Cancel:

leave the requester without any changes to the settings.

This font-requester is part of the reqtools.library by Nico François.

1.14 How to use the reqtools-screen mode-requester

When you click at the 'Screen' gadget in the settings window, this requester will appear:

```

Select screen mode...
+-----+-----+
|                                     | |
|                                     | |
|                                     | |
|                                     | |
|      Modes list                    | |
|                                     | |
|                                     | |
|                                     |#|
|                                     |Scrollbar
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |#|
|                                     |-|
|                                     |^Arrow up
|                                     |-|
|                                     |vArrow down

```

```

+-----+
| (Selected mode) |
+-----+

Overscan: [O.....OverscanMode.....]

Width:      [.....]  [x] Default_Wid
Height:     [.....]  [x] Default_Hi

Colors:    #? [:::#:::] Max: 256

AutoScroll: [x]

[-Use-]                                [Cancel]

```

Modes list:

List of all modes usable by the system. Scroll through by the Scrollbar and by the two arrows on the right hand side.

You can add new modes when you move new monitors from 'SYS:Storage/Monitors' to 'SYS:Devs/Monitors' using the workbench, e.g. the 'NTSC' monitor allowing you to have a NTSC HIRES/INTERLACED 640x400 screen which does not flicker as much as the PAL equivalent.

Be careful ! DO ONLY USE MONITOR MODES YOUR HARDWARE SUPPORTS !!
Your system could suffer damage from such experiments !!

Selected mode:

The mode you are currently watching at.

Overscan:

Click at the gadget and its content will change (Cycle gadget).
Select the overscan mode you want to work with. You can edit the Text Size / Graphics size by the preferences program 'Overscan'.
Select both Default checkboxes and cycle thorough the modes to see what happens to the screen dimensions.

Width: Height: Default_?:

Enter the width and height of the screen you want to open. Hit 'Default_?' set the default values.
You can use a (much) bigger screen, but in this case you're advised to select the 'AutoScroll'~button underneath, additionally (see below).

Colors:

Using the scroller you can specify how much colors the screen should have. The use of more colors will always mean bigger usage of CHIP memory ! Therefore be careful.
Impossible values (e.g 256 colors within a std mode) will be cut down to reliable values.

AutoScroll:

Select this gadget to have a screen with AutoScroll: If you are using a screen which is too big to be displayed on your monitor, you can move it when moving your mouse to its corners. Try it.

Use:

Use this screen mode.

Cancel:

Abort the requester without any change to the previous settings.

This requester is part of the reqtools.library by Nico François.

1.15 Settings gadgets that invoke basic functions:

The settings window contains three gadgets to end the request:

Undo all	: All changes will be rechanged to the status before you opened the settings window.
Reset	: All values will be set to the default PubScr values
Save default	: A file-requester will be opened. Enter the name of your current set. A postfix '.pub' will be added, automatically. If the set is been saved, it will be saved as default set file. That means, that PubScr will use this set when setting up without any parameters ! After all, the requester will be closed and PubScr will set up with the new settings (see Gadget 'Use'). Can also be used by menu.
Use	: PubScr will now verify all data, close window and screen (if possible). Then, the new screen will be set up with your new data.
Cancel	: Aborts the settings window; nothing will happen ! Can also be used by menu.

1.16 Looking at this file by menu: View doc

You can call this function both from main and the settings menu.

First, PubScr tries to open an AmigaDOS environment variable called 'ENVAR:pubscr/HelpFile' (GLOBAL) or 'ENV:pubscr/HelpFile' (LOCAL). In the case this was possible, PubScr reads the path and name of the document file from it and displays it using the amigaguide.library. Then, you will be faced to the standard amiga hothelp-system.

If the environment variable and/or the corresponding document file cannot be found, PubScr will open a filerequester begging you to locate the file within your system.

Just search the file (ought to be named 'pubscr.guide') and select it !

Then, the new path will be saved to the environment variable mentioned above.

You can also set the document's path by hand: Load the CLI or the

Shell, and type

```
'SetEnv pubscr/FileName <where_pubscr_can_find_it>'
```

Once set, the new variable will be present for all PubScr sessions in future.

1.17 PubScr icon handling:

PubScr tool icon:

There's only one tool-type defined: STDEF=<set.pub>. Enter the name and path of the set you want to load when starting without any parameters.

You can set this string by the settings menu Project/Set default and you may save a set as default by the gadget Save as default.

PubScr project icons (the saved sets, you know):

The default tool must contain the complete path name of the main program 'PubScr'. If you save a set, it will be set automatically. No tool-types defined, yet.

The default icon for PubScr's projects can be edited.

Therefore save it as 'ENVARC:PubScr/DefPubScr_project.info'.

1.18 Gadgets used within PubScr:

You will be faced to these kinds of gadgets:

Text gadget: Activate it and type in what you think will make you happy. you can hit 'TAB/Shift-TAB' to jump between the different text gadgets. Hit return to finish input.

Action gadget: Click at the gadget to invoke the relating action.

Checkbox: Toggles between activated/disabled.

Each gadget can also be activated by the underscored keyboard-equivalence. Gadgets which are shown shaddowed can't be selected.

One example: There is no reason to enter a window's dimension when you want to backdrop it.

1.19 Public screens - a new OS2.0 feature !

Public screens:

Under Kick1.2/1.3 a program could hardly open a window on other programs' screens (ie.displays). The programmer had to decide whether he wants his window(s) to be opened on the workbench screen or on a screen of his own.

One example: You're writing a program like 'DiskOpus', and you want to open a CLI window on your own ('custom' called) screen. The only way to figure that out was to patch the system ie. to take into account that you

are doing something that has been 'forbidden'~by the Amiga OS programmers.

With Kick2.0, public screens had come: A program can open a screen which it wants to be 'public' (commonly usable): Every program which supports this new feature 'public screen' is now able to open its window on this screen. These windows are called 'visitor windows', because the windows are visiting another program's screen !

Of course, now the 'public screen creating program'~must not close the screen if there're still visitor windows, because the other programs would not notice that their windows disappeared and will wait for input from a window which is already dead.

Thus, a public screen (or 'pubscreen', 'pubscr') must not close a screen as long as one alien visitor window is still active.

In this case,it means,that you can only close a 'PubScr public screen', when you have closed all other windows first.

When you change the settings, you have to close all windows first,too, because in the case you defined new settings,PubScr closes the old screen and opens a new one.

1.20 PubScr again and again...

History is just an idea but no comparison to the time you left behind:

- V0.01 Initial version. First planned as Shell argument version only.
 - V0.02 Command line parsing had problems using when detaching from the Shell.
Changed into icon based argument system.
No disk operation, no settings requester, etc...
 - V0.04 Some bugs fixed, several settings usable (by menu).
Still problems with the CLI command line parsing.
 - V0.05 General change: The settings will externally be saved.
Thus, PubScr is easier to use.
 - V0.05b Icons will also be created when saving a set.
 - V0.05c One tool-type remained for 'PubScr': The STDDEF startup definition.
 - V0.06 Very much bugs found.
 - V0.08 Bugs removed, settings window created using 'GadToolsBox' V2.0b (the bugs in its C-Source removed using an older tool of mine: GTP V1.06 'The GadToolsBox2.0b C SourceCode Patcher').
PubScr works very well.
 - V0.09 Possibility to select the default startup-set within the program added.
Option to fill up the window with a pattern added.
Function to delete sets added.
General bugs fixed and code optimized.
Thinking about an amigaguide help-system (just to train me; in fact, I think such a little program doesn't need such a function, but who cares ?).
 - V1.00a Amigaguide installed. Doesn't work. Why ?
 - V1.00b Amigaguide is buggy: My documents promise that one can pass the full path name to the function OpenAmigaGuide(), but this way it doesn't work.
-

Problems when supporting nag_Lock, too. Doesn't care for it...?

Now I change into the directory where this documentation is located in by myself before PubScr calls OpenAmigaGuide().

It's not nice this way, but it works !

V1.00c AmigaDOS environment variable used to store the path of this doc. Seems to work better than a new tool-type, because it's nothing specific for one copy of PubScr, but of common importance and truth !

Little bug in the filerequester-routine fixed.

V1.00d Everything works well !

V1.00e FIRST RELEASE

Workbench can be closed, now using an ENVARC: global variable (unfortunately PutEnv() doesn't offer this so I was needed to open a text-file myself...), little bug in the settings part fixed (the filename in the requester was always set to the standard settings' file...)

V1.01 SECOND RELEASE 25.2.1994

Startupscript or -command now available. PubScr now executes a script/command when having built up the screen. The script/command can later be executed by hand, too. Function to open a shell is been added. Additionally, PubScr is now able to detach itself from the Shell.

Undo/Reset to defaults gadget added to settings window. Code now optimized.

The future: What will happen ? But who cares, on the other hand ?
Here is what I want to do next:

- Use the workbench window - pattern/picture. I may copy the bitmap. But then I'm possibly ought to scale it. Would this look nice ?
- Resident code.
- I may add an option to add alien tools to the menu. Could be nice at all.
- More comfortable use of the MENUHELP-Flag. Now it just shows the general topic in this guide (try it...:-).
- 'Sleep' tooltype for the project: open pubscreen if one's trying to lock it, only.
- 'Iconify' function.
- Make use of locale.library !

Contact me if you have further considerations !

1.21 Copyright (c)1994 by CodexDesign:

-- INSURANCE --

The author cannot be held liable for the suitability or accuracy of this manual and/or the program it describes. Any damage directly or indirectly caused by the use or misuse of this manual and/or the program it describes

is the sole responsibility of the user her-/himself.

-- ADDITIONAL USED PROGRAMS --

- * all requesters designed and programmed using the fabulous and very, very comfortable

'reqtools.library'~ (c)1991/1992 by Nico François
all rights reserved.

- * settings window hotkey handling done using the not even less fabulous and genius

'gadtoolsbox.library' (c)1991-1993 by Jaba Development
(Jan van den Baard)
all rights reserved.

- * settings window itself designed, changed and translated into C-Source by the

'GadToolsBox 2.0b' (c)1991-1993 by Jaba Development
(Jan van den Baard)
Giftware; all rights reserved.

- * GadToolsBox C-Source automatically debugged and patched(unfortunately GadToolsBox doesn't make use of its own 'gadtoolsbox.library') by

'GTP 1.06' (c)1993 by Codex Design Software
(Hans Bühler)
Freeware; source available from the author.

-- THANK --

Jaba Development: GadToolsBox is GREAT !!! You'll hear from me, soon !
I've just tried to call you, but I failed !
gadtoolsbox.library is GREAT, too, but still has some bugs (MX - Gadgets)

Nico François: Thank you for everything you did for this computer !

David Braben: For Frontier - Elite II (since I bought this game, all my
programmer projects had time to rest on my HD !!!!)

Cat Stevens: For the music he's making.

Genesis: For all you did until 1974 !

Benjamin Panier: For all he did and we did together.

Niels Linde & Lutz Petrick: For all the funny nights.

Christian Würdemann: Once you persuaded me to learn programming.
I did.

-- PUBSCR ITSELF --

This documentation, the program it describes and all additionally supplied files except those mentioned above are

(c)Copyright 1993/1994 by Codex Design Software
all rights reserved.

You are allowed and advised to distribute this program as long as you
a) don't purchase money for that except a LITTLE amount for copying the
data and the disk (if sold),
b) don't change anything in the program, the documentation,
c) distribute all files provided together.

To distribute this program commercially get a written permission from
me first !

Codex Design Software (section Amiga)
Hans Bühler,
Kirchstr.22, 10557 Berlin 21
FRGermany
call: +49 (0)30 39333814

Contact me for whatever you want.

1.22 Amiga on its way to paradise

\ /
-- * --
/ \

A little star, once born and often misunderstood,
is still shining down on us
- on those who believe in this little star,
knowing the strength behind its warm light,
feeling the endless universe
inside its slight corona.
A star among others, at the dark sky at night.

But we believe !

For us, the way it shines and glows
is bright and wonderful,
and nobody else could ever understand !

A light among others
but a hidden conqueror !

Amiga is the sun we need to live !

Be his victim ! And be lucky, then !
I swear I am.

\ /
-- * --

/ \