

GUIFront

COLLABORATORS

	<i>TITLE :</i> GUIFront		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		July 22, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	GUIFront	1
1.1	The GUIFront Preferences Editor	1
1.2	Introduction	1
1.3	Installation	2
1.4	Usage	2
1.5	Gadget Preferences	3
1.6	Frame Preferences	4
1.7	Backfill Preferences	4
1.8	Miscellaneous Preferences	5
1.9	Application Information	5
1.10	Menus	5
1.11	Program history	7
1.12	Author	7

Chapter 1

GUIFront

1.1 The GUIFront Preferences Editor

Documentation for
GUIFront Preferences v1.1
An editor for the GUIFront
Dynamic GUI Layout Engine
Written by Michael Berg
Copyright (C) 1994 by Michael Berg
All rights reserved,

This document describes how to use the GUIFront preferences editor. Please select an item from the table of contents listed below:

Introduction	Introduction to the GUIFront Preferences editor
Installation	How to install it
Usage	How to use it
History	Program history
Author	How to reach the author

1.2 Introduction

Introduction

The GUIFront preferences editor is a utility which allows you to customize the visual appearance of GUIFront applications. For example, you can tell an application use a particular font, or change the color of frame headlines for another application. Or a combination of those, of course.

See the section on usage for a complete description of all features.

1.3 Installation

Installation

The GUIFront preferences editor installs very easily. Simply copy it and its icon into the directory or drawer where you want it. If you want the editor to run localized, you must also copy the necessary catalog files to your LOCALE: directory.

Two versions of the editor have been provided. One runs on all Amigas, while the other requires an Amiga with at least a 68020 processor. To maximise efficiency you should install the version which most closely matches the CPU in your machine.

I shouldn't have to say this, but just in case anyone's wondering - yes, you do need the guifront.library to use the preferences editor :)

Like the GUIFront library itself, the GUIFront editor requires at least AmigaDOS 2.04.

In case you are still confused, simply let one of the installation scripts do the work for you.

1.4 Usage

Usage

The GUIFront preferences editor allows you to change almost all GUI related aspects of a GUIFront application. It also allows you to define a "default" setting, which new applications automatically inherit when they are first run.

To use the editor, either double-click on it's icon, or activate it from a shell. You should see the main editing window pop up.

The listview gadget contains a list of active applications. To edit the preferences of any particular application, first select the application from this list. To edit the default preferences (which are inherited by new applications when they are first launched), select the entry marked with "« Default »").

Pressing the Gadgets button brings up a new window which allows you to change various settings pertaining to gadgets (font, style, etc).

The Frames button opens a window which allows you to modify settings pertaining to frames and frame headlines.

The Backfill button opens a window which allows you to control whether or not an application is allowed to use backfilling, and if so, what color the backfill should have.

The Misc button opens up a window with miscellaneous other settings. At present the only setting you can control from here is what type of refreshing an application should use, but more controls are planned.

The Info button brings up some information on the currently selected application. Note not every application supply information about themselves, and so some of the fields in the Information window may be empty.

Note that all the above windows are asynchronous. This means you can open and use as many of them as you like at any one time.

The Delete button allows you to delete all settings related to the currently selected application. You will be asked to confirm the operation, and then the application is deleted. Note you cannot undo this by pressing the Cancel button!

The Test button allows you to temporarily test the effects of the changes made to the preferences. Of course, this requires that the application(s) are also running at the time.

Note: GUI tests are passive on the part of GUIFront. An application has to explicitly listen for messages from GUIFront telling them their preferences have been changed, or the Test gadget will have no effect on them. If nothing seems to happen when you press Test, the most likely explanations are that either the application was written before preferences notification was implemented in the library, or is not currently able to change its visuals.

To permanently save the preferences as they are just now (regardless of which windows may be open), press Save. To use the current settings, press Use. To cancel changes you've made to the preferences, press Cancel.

Note that to `guifront.library`, changes happen only when you press Save or Use. If you launch an application while the preferences window is open, this new application will use the preferences which were in effect before the preferences editor was launched.

The GUIFront preferences editor also has some menus. See the menu section for a more detailed description of these.

Also note that the preferences editor listens for new applications to start. As soon as this happens, a new entry pops up in the application listview, regardless of what you are doing with the editor at that time.

Closing the window is logically equivalent to pressing Use, so you won't lose any changes if you do this.

1.5 Gadget Preferences

Gadget Preferences

This window allows you to control which font an application should use for gadgets, plus what style you prefer for various gadget types.

Select Screen Font to have the application use whatever happens to be the current screen font at the time when the application is launched.

Deselect Screen Font to have the application use a particular font. Use

the GetFont gadget to select the font the application should use.

The listview below the font settings contain a list of gadgets which can be rendered in one of two styles: Xen or Normal. Xen differs from Normal in that an extra black frame is rendered around the gadget hitbox itself. With the right font and a high graphics resolution, this can look remarkably good, especially for Buttons and Checkboxes.

Note that Checkboxes currently cannot be rendered in Xen mode under V36. You'll need at least V39 to enjoy this.

The Xen Frame Color palette gadget allows you to select which color should be used for rendering Xen frames. Xen frames are the thin frames which GUIFront draws around the hitboxes of those gadgets which supports the xen mode.

1.6 Frame Preferences

Frame Preferences

This window allows you to control how frame headlines are rendered for the selected application.

Select Screen Font to have the application use whatever happens to be the current screen font at the time when the application is launched.

Deselect Screen Font to have the application use a particular font. Use the GetFont gadget to select the font the application should use.

The Color gadget allows you to select the color of frame headlines.

The Alignment gadgets allow you to control if frame headlines should be vertically centered on (Center gadget), or rendered above the frame it is written on (Above gadget).

The Attributes gadgets allow you to change the attributes of frame headlines. Select Bold to render headlines in boldface, Italics to render headlines in Italics, and 3D to give headlines a black shadow. Note that the Color gadget controls the color of the headline, not the shadow. Selecting black color and 3D is therefore not a very good idea.

The Centering gadgets allow you to control the horizontal alignment of frame headlines. Select Right to have headlines rendered rightmost on the frame, Centered to have it centered and Left to have it rendered leftmost on the frame.

1.7 Backfill Preferences

Backfill Preferences

This window allows you to control application backfilling.

Select `Enable backfilling` to allow the currently selected application to use backfilling if it wants to.

If you enable backfilling, you can also select the two color components of the backfilling pattern itself, using the `Color 1` and `Color 2` gadgets.

1.8 Miscellaneous Preferences

Miscellaneous Preferences

This window allows you to control miscellaneous GUI settings for the currently selected application.

The Refreshing gadgets allow you to control what type of window refreshing the application should be using. Select `Smart` to use Intuition's smart refreshing scheme, and `Simple` to use manual refreshing.

The difference between the two methods is the old memory-versus-speed compromise over again. Smart refreshing looks great, it is quick and results in a minimum of visual flickering when windows are moved around. The drawback is that this type of refreshing is quite expensive in terms of memory usage. Worst case it uses twice the amount of memory it normally would.

Simple refresh is very cheap in terms of memory usage, but uses considerably more CPU time to perform. On slower machines, the flickering refresh cycle will probably be annoyingly obvious.

Thus, the general rule is if you've got the memory, use smart refreshing. Otherwise, if you have the CPU power, use simple refreshing to save memory.

1.9 Application Information

Application Information

This window simply shows you some information about the current application. Note that not all applications supply information about themselves. For such applications the information fields will remain empty.

1.10 Menus

Menus

The GUIFront preferences editor currently has the following menu items:

Project/

 Open...

This menu item allows you to load a complete set of preferences into the editor. Note that the current preferences (plus any changes you've made) will be lost in the process.

Save As...

This menu item allows you to save the current preferences to a file. Note that doing this has the side effect of making the saved preferences the actual, current preferences, just like if you had pressed Use.

About...

This menu item shows you some incredibly important information.

Quit

Select this menu item to quit the GUIFront preferences editor. The current settings will be lost in the process.

Edit/

Reset to Defaults/

This entry

This menu item will reset the current application to some internal, standard values. Note that this takes effect immediately (it may be cancelled, however)

All entries

Select this menu item to reset all application preferences to internal default values. Again note that this takes effect immediately (it may be cancelled, however).

Last Saved

This menu item resets the current settings to whatever you last saved using the Save gadget.

Restore

Select this menu item to undo all changes you've made to the current preferences.

Options/

Create Icons?

Check this menu item if you want project icons to be saved along with the preferences, every time you select Save As... from the menu.

GUIFront uses the image of the first of the following icons it can locate, for the icon to be saved:

ENV:Sys/def_guifront

```
ENV:Sys/def_project
ENVARC:Sys/def_guifront
ENVARC:Sys/def_project
```

If you design your own icon and store it under one of these names, GUIFront automatically uses this icon.

1.11 Program history

Program history

7.10.94 - Release 1.1, library version 37.3

- Added italiano translation (Michele Giorato, 2:333/707.4)
- Corrected a few minor bugs
- If guifront.library could not be opened, the program would crash
- Added Test gadget to main GUI

20.6.94 - Release 1.0, library version 37.2

1.12 Author

Author

If you have any suggestions, comments or catalog translations for GUIFront, you can reach me at the following addresses:

E-mail: mberg@scala.ping.dk (preferred)

Fido: 2:238/ 24.28

AmigaNet: 39:140/101.28

S-mail: Michael Berg
Sjællandsgade 56, 4
8900 Randers
Denmark