Visualizza

COLLABORATORS			
	TITLE : Visualizza		
ACTION	*NAME*	*DATE*	*SIGNATURE*
WRITTEN BY		August 15, 2024	

REVISION HISTORY			
NUMBER	DATE	DESCRIPTION	NAME

Contents

Chapter 1

Visualizza

1.1 Documentation of Visualizza 1.3

Visualizza 1.3

by Massimo Tantignone

1.2 Why Visualizza?

1. WHY VISUALIZZA?

After I finished programming ConvertiILBM, I found myself having a lot of generic functions for the handling of IFF files and bitmap graphic. Thus I decided to extract all these functions from ConvertiILBM and to collect them into an external library. In fact two libraries came out, IFF.lib and Max.lib. The latter was in the following months updated with many new general utility functions, and now most of its contents have NOTHING to do with graphics (hence the generic name Max.lib).

At this point I thought it would have been very simple to put together a displayer making use of these libraries; the quantity of new code to write would have been definitely limited (and indeed it was).

From all that was born Visualizza; this program is therefore to be

intended only as a didactic experiment, and it has nothing special (at least for now) compared to other popular displayers.
The program is here only because of its "relationship" with ConvertiILBM but it should not be seriously taken into consideration if you have other displayers (try it, however).
Because of its heavy usage of external linker libraries, it would be totally useless to supply the source code of the program without including also the required libraries; therefore the source code will be available in the future only if IFF.lib and Max.lib should also be distributed.
I know that with the 2.0 operating system probably IFF.lib is made useless by the new "iffparse.library", but you never know, someone could find it simpler to use.

1.3 Usage from Shell

2. USAGE FROM SHELL
If after all that you still intend to use Visualizza, here is its Shell template:
FROM/A/M,TEXTFILE/K,GUI/S
As you can imagine, the program can display several files sequentially (slideshow style) and this is the only detail by which its behavior differs from ConvertiILBM (as a displayer).
An interesting feature of the program is that a non-fatal error doesn’t interrupt a multiple display: if you, for example, enter the command
Visualizza Thing Thang Thung
and the file "Thang" isn’t in the IFF ILBM format, it will be reported an error message but the display of "Thung" shall be performed (or at least attempted) anyway.
As ConvertiILBM, Visualizza will attempt to "center" horizontally as much as possible the non-brush images (while many other displayers don’t do that, not even Commodore’s official Display, heh, heh).
Furthermore it can also handle the ILBM files containing a stencil (such as those produced by DPaint).
The TEXTFILE keyword allows you to specify the name of a text file in the ASCII format to be used to configure Visualizza into a language different than english (the default). For details on this see paragraph 4, "Localization".
The GUI switch, if specified on the command line, brings up a file requester to be used to select the files to display, as if Visualizza was started from the Workbench. Contrarily to the analogous WB option of Visualizza 1.2, this feature isn’t experimental and doesn’t imply any risk. See also the next paragraph for further information on the file requester.

1.4 Usage from the Workbench

3. USAGE FROM THE WORKBENCH
The usage of Visualizza from Workbench is totally identical to that of ConvertiILBM when it is used as a displayer.
For more detailed information on the usage of Visualizza from the Workbench, then, see the file "ConvertiILBM.guide" (if you are reading this document you should own that, too) and read specifically the paragraph 5.2, "Display from Workbench".
What is said in that paragraph about ConvertiILBM applies exactly also to Visualizza.
Anyway, you can use the extended selection (even a multiple one) and the specification of the Default Tool. Furthermore by double-clicking on its icon without specifying others it will appear a file requester with which you will be able to choose the file to display. To be able to obtain the file requester there must be in the LIBS: directory of your system at least one of the libraries asl.library, reqtools.library and arp.library (they are searched in this order of preference). If none of the three libraries should be found, it will appear a simple requester with an error message. In this case, you will be to display files only by the two methods mentioned above (extended selection or Default Tool).
Note: reqtools.library is copyright © Nico François.
You can specify the tool type TEXTFILE=<filename> in Visualizza's icon thus obtaining the same effect of the TEXTFILE keyword used from Shell.
See the paragraph 4, "Localization", for more information.

1.5 Localization

4. LOCALIZATION
Even in the matter of localization, Visualizza behaves exacty as ConvertiILBM, meaning that it supports both the specification of an ASCII configuration file (through the TEXTFILE keyword or tool type) and the direct usage of the locale.library and of a .catalog file.
Refer then to paragraph 6 of the file "ConvertiILBM.guide" for all details, bearing in mind that anything said there about ConvertiILBM also applies to Visualizza.
Here it is needed only to point out that the default name for the text file, if you don't want to specify an alternative path with TEXTFILE, is "S:v.txt", that is, the file should be named "v.txt" and should be in the S: directory of your system, while the name of the catalog file is "visualizza.catalog" and you find it together with "convertiilbm.catalog" (if it is supplied for your language).

1.6 Existing revisions

5. EXISTING REVISIONS

1.3 Enhanced the handling of the file requester, changed the syntax
from Shell, fixed some displaying problems under 2.x/3.x and made
smoother the display in sequence. Also added localization's support.

1.2 Now Visualizza is able to perform displays in sequence.

1.1 Added Workbench support.

1.0 The original version.

1.7 Known bugs

6. KNOWN BUGS

None. Alas the program also isn't too useful. If you really want to
receive the source code and/or ßeta versions of IFF.lib and Max.lib send
5$ to:

Massimo Tantignone
Via Campagnoli, 4
28100 Novara (NO)
ITALY

That's all; let's see us (figuratively speaking) by the next release!

Massimo Tantignone, 29 march 1993

//

"Thanks to the \X/ AMIGA for being the best computer ever!"