

Convert to LBM

COLLABORATORS

	TITLE : ConvertiILBM		
ACTION	NAME	DATE	SIGNATURE
WRITTEN BY		August 15, 2024	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	ConvertiILBM	1
1.1	Documentation of ConvertiILBM 2.1	1
1.2	Presentation of the program	1
1.3	Purpose and motivations	2
1.4	General information	4
1.5	already_know	4
1.6	all_users	4
1.7	Usage from Shell	5
1.8	Usage from Shell/Template	6
1.9	tke	6
1.10	Usage from the Workbench	9
1.11	Usage from the Workbench/Conversion from Workbench	9
1.12	Usage from the Workbench/Display from Workbench	10
1.13	wb_errors	10
1.14	Localization	11
1.15	Conclusion	12

Chapter 1

ConvertiLBM

1.1 Documentation of ConvertiLBM 2.1

ConvertiLBM 2.1

by Massimo Tantignone

© 1989-1994 MT Soft

1. PRESENTATION OF THE PROGRAM
2. PURPOSE AND MOTIVATIONS
3. GENERAL INFORMATION
4. USAGE FROM SHELL
5. USAGE FROM THE WORKBENCH
6. LOCALIZATION
7. CONCLUSION

1.2 Presentation of the program

1. PRESENTATION OF THE PROGRAM

The program ConvertiLBM allows to convert any file that is in the IFF ILBM format into several other formats, among which icons, Sprites, BOBs, source code in various languages and raw binary files, and is also able to display it and examine its more significant chunks.

In addition it offers a number of options to vary the characteristics of some of the file types being produced.

The program handles IFF ILBM files of any resolution and graphic mode, including HAM, EXTRA-HALFBRITE and overscan.

It doesn't support the new AGA graphic modes yet, because I don't own an Amiga 4000 (but it is only a matter of time).

All conversion functions can be used both from Shell and from Workbench; displaying files from Workbench also supports extended selection and the indication of ConvertiLBM as Default Tool in an ILBM file's icon.

The program is compatible with any Amiga operating system, it works on PAL and NTSC machines, it can save the preferred settings into its icon and it can be adapted to any language (using the locale.library under 2.1 or higher and with another method under 2.04 or lower).

ConvertiLBM will come useful mainly to programmers wanting to include images and animations into their applications, but even other users could find useful, for instance, the possibility of converting their pictures into Workbench icons.

Because with version 2.0 ConvertiLBM has been almost completely rewritten and features noticeable differences compared to ConvertiLBM 1.4, I suggest you to read entirely the present documentation, even if you are already familiar with the previous version (non-italian users probably have never heard of ConvertiLBM before anyway).

NOTE: The italian version of this documentation is somewhat more detailed than the english one. Feel free to translate entirely the original doc file and to include it into the distribution directory of ConvertiLBM. I doubt I'll ever find the time to do this myself.

1.3 Purpose and motivations

2. PURPOSE AND MOTIVATIONS

Often Amiga programmers find themselves in need of inserting bitmapped images into their applications, but to do this these images must be in an appropriate format.

In fact a program could load from disk a IFF graphic file at run time, but usually it is preferred, for small applications, to code the image data directly in the executable, or, in the case of games or already very large programs, to load from disk raw binary files that are easier to handle at the input/output level.

Sometimes, then, the format is imposed by the programming language being used; for instance, in AmigaBASIC it is possible to read from disk the BOBs or the VSprites to be used in one's own animations, but only if the file containing them is structured as those produced by the ObjEdit program.

Finally, both the programmer and the user have often the desire or the need of customizing the icons of their own files, and to do that usually they need a drawing program that allows them to save their work as a

Workbench icon (such as IconEd or IconEdit).

Alas, not all drawing programs allow to save images into all of these different formats, which means that it is almost always necessary to use a different program for each one of them.

That may be problematic, as there are few really valid format-specific drawing programs, and in some cases it is available only one such program, and furthermore not a particularly reliable one (as, for instance, in the case of the already mentioned ObjEdit for the AmigaBASIC BOB or VSprite files).

The only solution in these cases is to use one drawing program, for instance DeluxePaint, to create all needed images, and then to "convert" them into the required format through the use of other programs, coming usually from the public domain. The problem, however, that way is merely postponed: we fall back into the necessity of having a different conversion program for each format we may need.

Besides, by converting the images outside the program with which they were created, one risks to get confused about the names of two different images, or to not remember the exact contents of a file, especially after some time; therefore it is often also needed a displayer to have a look at the images before converting them.

To resolve all these problems I created ConvertiILBM.

ConvertiILBM gathers into one program the functions of several others converters and displayers, from public domain and not, and therefore it eliminates the need to have many editing programs allowing to use only one (such as the DPaint) whatever be the file type you want to obtain in the end.

Thanks to ConvertiILBM you shall be able to eliminate from your disks or from your hard disk the programs:

IconEd

IconMerge

IE

LoadILBM-SaveACBM

ObjEdit

Gi

ZapIcon

and all their equivalents, as well as several simple displayers such as Show, SeeILBM, Display, and so on.

However, because the display ConvertiILBM does is very simple and its purpose is only to give a look at the files being converted, the program

can't replace more powerful displayers such as, for instance, ViewTek by Thomas Krehbiel or Mostra by Sebastiano Vigna, currently the best ones in circulation.

To sum it all up, now you can draw with the DPaint your icons, your BOBs or Sprites to use in your AmigaBASIC animations, and generally all images you may want to insert into your programs.

I think that using the DPaint is better than using IconEd or ObjEdit, as they in addition to being obsolete also are not much powerful and definitely uncomfortable for the user (not to mention their many bugs: I never managed to conclude a work session with ObjEdit without getting a visit from the Guru).

Even if with the release 2.0 of the operating system the IconEd program (now IconEdit) has been remarkably enhanced, certainly it is not yet in the DPaint's league. Furthermore, although IconEdit allows now to import IFF files, they get "cut" if they exceed the maximum accepted size (rather small). ConvertiILBM, instead, doesn't impose limits on the icon size.

1.4 General information

3. GENERAL INFORMATION

3.1 For those already knowing ConvertiILBM 1.x

3.2 For all users

1.5 already_know

3.1 For those already knowing ConvertiILBM 1.x (all the five of them)

ConvertiILBM 2.x is a new version of the program and not a simple revision. From version 1.4 were introduced several enhancements, changes, additions and corrections. So read on!

1.6 all_users

3.2 For all users

ConvertiILBM can currently convert an IFF ILBM file into the following formats: ACBM file, uncompressed ILBM file, source code (BASIC, C, Modula-2 or Assembler), raw binary file, C source code for a sprite, icon (a new one or an alternate image) and AmigaBASIC VSprite and BOB file.

Most conversion types present some specific options allowing you to configure the produced file; for instance you can decide whether to add or

not a colormap to a source code file or to a raw binary file, or whether the bitmap of the generated file must be contiguous or interleaved, or even the number of spaces to be used for the indentation of a source code's lines.

In order to make it easier to identify the desired ILBM files before converting them, the program allows you also to display an ILBM file and to examine the contents of its more significant chunks.

From CLI/Shell you can perform a single operation for each time, by running ConvertiILBM and specifying on the command line the names of the input and output files, the conversion type and/or the desired options, if any.

From the Workbench you can instead perform many conversions in one session, by using a comfortable control panel that appears when you double-click on the program's icon.

While working this way it is also possible to save the current settings of the panel into the ConvertiILBM's icon; this way every time you execute the program from Workbench these settings will get restored. The saved setting will instead be ignored if you use ConvertiILBM from CLI/Shell.

IMPORTANT NOTE: To be able to use the control panel under 1.3 (or lower) it is necessary that in your system's LIBS: directory be present a PD library emulating the gadtools.library, as the one I made and included with the program. If you are still using the 1.3 operating system, then, copy into your LIBS: directory the file called "gadtools_34.library" that comes with ConvertiILBM and rename it to "gadtools.library" (if you don't rename it the program won't find it, I do not distribute it directly with the name "gadtools.library" as I suspect that name to be copyrighted).

I realize all that may end up to be annoying, but this is only another reason you **SHOULD** update your Amiga and install at least the Release 2 of the operating system, if not an even higher one.

1.7 Usage from Shell

4. USAGE FROM SHELL

4.1 **Template**

4.2 **Explanation of the template**

1.8 Usage from Shell/Template

4.1 Template

**FROM/A,TO,CONVERSIONTYPE,LANGUAGE,ICONTYPE,SHOW/S,EXAM=EXAMINE/S,
CMAP=COLORMAP/S,INTERL/S,BHEADER/S,NOTRANSBACK/S,PAINTBRUSH/S,
WIDTH/K/N,TAB/S,INDENT/K/N,JOINLINES/S,TEXTFILE/K,GUI/S**

A keyword like CMAP=COLORMAP means you can use without any distinction CMAP as well as COLORMAP (i. e. the two keywords are equivalent).

1.9 tke

4.2 Explanation of the template

FROM

Name of the input ILBM file. Do not type "FROM" literally but only the file name. Specifying only that argument causes the display of the file.

TO

Name of the output converted file. Do not type "TO" literally but only the file name.

CONVERSIONTYPE [default: SOURCE]

Type of conversion to be performed. Don't type "CONVERSIONTYPE" literally but only the name of the conversion type. The current available types are:

ACBM IFF ACBM (Amiga Contiguous BitMap) graphic file

ILBM IFF ILBM (InterLeaved BitMap) uncompressed file

SOURCE Source code file in ASCII format

BYTES Raw binary file with plain image data

SPRITE C source code with a sprite's definition and image data

ICON Workbench icon file (new or alternate image)

VSPRITE AmigaBASIC VSprite binary file

BOB AmigaBASIC BOB binary file

LANGUAGE [default: C]

Language of the source code to be produced (if **SOURCE** specified).

Don't type "LANGUAGE" but only the name of the language. The currently available languages are:

BAS=BASIC AmigaBASIC language

C C language

M2=MODULA2 Modula-2 language

ASM=ASSEMBLER Assembly language

ICONTYPE [default: PROJECT]

Type of the icon to be produced (if **ICON** specified). Don't type "ICONTYPE" but only the name of the type. The possible types are:

TOOL Icon of an executable program

PROJECT Icon of a data file

DRAWER Icon of a directory (drawer)

DISK Icon of a disk or device

TRASHCAN Icon of the Trashcan

By specifying APPEND it is not created a new icon, but the image of the ILBM file is appended to an already existing icon file to form an icon with an alternate image.

SHOW

Switch to have ConvertiILBM display the ILBM file. Press any key or mouse button to end the display.

EXAM=EXAMINE

Switch to have ConvertiILBM examine the ILBM file, printing a list of the more significant chunks along with their contents, if applicable. The program currently recognizes the BMHD, CMAP, CAMG, GRAB and BODY chunks; of course of the last one will be showed the length only.

CMAP=COLORMAP [default: no colormap]

Switch to have ConvertiILBM include a colormap in the output file. This is applicable only to the **SOURCE**, **SPRITE** and **BYTES** conversion types. The colormap for a raw binary file is appended at the end of the file and consists of a number of words (2 bytes), a word for each color, in the same format used by the LoadRGB4() function.

INTERL [default: contiguous bitmap]

Switch to have ConvertiILBM output a file with an interleaved bitmap, as opposed to a file with a contiguous bitmap. This is applicable only to the **SOURCE** and **BYTES** conversion types.

BHEADER [default: no BASIC header]

Switch to have ConvertiILBM add a 3 word (6 bytes) header to the raw binary files it produces, containing the values of width, height and depth of the image. Useful to load the binary file directly into an AmigaBASIC array to use with the GET/PUT instructions.

NOTRANSBACK [default: transparent background]

Switch to have ConvertiILBM generate a BOB file with a non-transparent background. The background will be solid and of color 0.

PAINTBRUSH [default: BOB preserves background]

Switch to have ConvertiILBM generate a BOB file that doesn't preserve the background which it moves on, leaving a trail like a paintbrush.

WIDTH=<columns> [default: 76, minimum: 16]

Formatting option to specify the maximum line width of the source code produced (if **SOURCE** specified). The hexadecimal image data won't exceed that width but will go instead onto the next line.

By specifying WIDTH=UNLIMITED (or anything beginning with an U) the program will insert a newline only after a full line of image data.

TAB [default: use spaces]

Formatting switch to have ConvertiILBM use tabs instead of spaces to do the indent in the source code it produces.

INDENT=<characters> [default: 3, minimum: 0]

Formatting options to specify the size of the indent of the source code to be produced. If **TAB** is also used, make sure your editor's tab size to be the same as what you specify with this option.

JOINLINES [default: do not join lines]

Formatting switch to have ConvertiILBM put as much data as possible (with respect to what specified with **WIDTH**) onto a single source code line, that is it won't insert a newline between consecutive image data lines. Do not specify both JOINLINES and **WIDTH=UNLIMITED** or your code will end up being composed by one long line of text.

TEXTFILE=<filename> [default: S:ci.txt]

Option to select an ASCII text file containing the strings used by ConvertiILBM, useful to configure it to any language. Do not create these files by yourself, but use or modify (translate) one of those included with the program. However, if locale.library is available, use Amiga's standard localization system instead of this method.

See also paragraph 6, "**Localization**" for more information.

GUI [default: no GUI]

Switch to have ConvertiILBM open its control panel as if launched from the Workbench. All other parameters are taken into consideration and used to configure the panel at its opening.

GUI is the acronym of "Graphic User Interface".

Lastly, by typing "ConvertiILBM ?" you'll get a reminder of the standard template, while typing "ConvertiILBM ??" (with two question marks) the more traditional syntax will be displayed, as follows:

ConvertiILBM 2.1 by Massimo Tantignone - © 1994 MT Soft

Usage: WORK:ConvertiILBM <Input file> [<Output file>]

[ACBM | ILBM | SOURCE | BYTES | SPRITE | VSPRITE | BOB | ICON]

[BAS=BASIC | C | M2=MODULA2 | ASM=ASSEMBLER]
[TOOL | PROJECT | DRAWER | DISK | TRASHCAN | APPEND]
[SHOW] [EXAM=EXAMINE] [CMAP=COLORMAP] [INTERL] [BHEADER]
[NOTRANSBACK] [PAINTBRUSH] [WIDTH=<n>] [INDENT=<n>] [TAB]
[JOINLINES] [TEXTFILE=<file>] [GUI]

The parameters can be specified in any order and case, but the input file name must always come before the output file name.

1.10 Usage from the Workbench

5. USAGE FROM THE WORKBENCH

5.1 Conversion from Workbench

5.2 Display from Workbench

5.3 Errors while displaying

1.11 Usage from the Workbench/Conversion from Workbench

5.1 Conversion from the Workbench

ConvertiILBM can be used from the Workbench as a converter or as a displayer. To use it as a converter (main usage) simply start it by double-clicking on its icon. A control panel will pop up, with which you'll be able to do your conversions in an highly intuitive way.

The control panel features several gadgets. In the upper part of the window you'll find two string gadgets to enter the names of the input and output files the program must operate on. Besides them there are also two button gadgets whose purpose is to display a file requester to quickly select the desired file names.

To create the file requester ConvertiILBM will try to locate and use, in this order, asl.library, reqtools.library or arp.library. If none is found, the button gadgets will be disabled at startup.

Note: reqtools.library is copyright © Nico François.

In the left half of the panel there is a series of mutual exclude gadgets to choose the main conversion type you want the program to perform. Depending on the chosen type of conversion, the right half of the window shall contain some other gadgets to fine-tune the desired operation. Some conversion types, however, don't offer any additional gadgets.

The purpose and use of these gadgets should be intuitive as they have a direct correspondence to the already described Shell parameters.

The rest of the panel's gadgets will trigger a particular action, for

instance to display the input file or to actually execute the conversion.

The gadget labeled "Advanced options..." will bring up a second panel with which you'll be able to choose some formatting options for the source code the program will generate (only if you selected the conversion into source code or sprite).

With the Project menu you can save the current settings of the control panel into the program's icon (with the "Save settings" item). The settings will be used to configure the panel at the startup the next time you'll execute ConvertiILBM. Under 2.0 (or higher) the program can create an icon for itself if one doesn't already exist, while under 1.x it cannot.

Do not add, delete or modify by hand the tool types of the program's icon, except for the TEXTFILE tool type; if you add this tool type it will be preserved by ConvertiILBM at each new saving of the settings.

The other items of the Project menu are rather self-explanatory.

1.12 Usage from the Workbench/Display from Workbench

5.2 Display from the Workbench

To display an IFF ILBM file from the Workbench using ConvertiILBM, all you have to do is to click on ConvertiILBM's icon and then, holding down the SHIFT key, click on the icon of any ILBM file you want to display, double-clicking only on the last one. It is however not necessary that ConvertiILBM be the first icon selected, it just has to be comprised in the extended selection.

Alternatively, you can set ConvertiILBM as the "Default Tool" in the Info (Information under 2.0+) window of the icon of an ILBM file, then simply double-click on that file. If ConvertiILBM isn't in the same drawer as the ILBM file, you must specify its full path in the "Default Tool" string gadget.

You can also mix the above methods; if you multi-select a series of ILBM files one of which has ConvertiILBM as the Default Tool, the program will be invoked for each one of them even if it wasn't directly part of the multiple selection.

1.13 wb_errors

5.3 Error while displaying

If any error occurs during a multiple display, a requester with an error message will pop up. If the error wasn't a fatal one the display shall continue after the requester has gone away.

1.14 Localization

6. LOCALIZATION

If you are lucky enough to own the 2.1 or 3.x operating system, you can take full advantage of the localization capability of ConvertiLBM in the Amiga's soon-to-be standard way, namely by using the locale.library.

All you have to do is to copy the "convertiilbm.catalog" file for your language, if supplied, to the directory LOCALE:Catalogs/<langname>, where <langname> is the name of your locale language. For example, if you are italian, you could copy the file "Catalogs/italiano/convertiilbm.catalog" (found in the ConvertiLBM's distribution directory) to the directory "LOCALE:Catalogs/italiano/" of your system, thus obtaining the localization file "LOCALE:Catalogs/italiano/convertiilbm.catalog".

Of course you must also, if you don't have already done that, set your preferred language with the Locale preferences editor.

Even if you don't own the locale.library (or if you don't find a catalog file suited for you), you still can configure ConvertiLBM to a language different from english by using an alternate method.

You should use either the **TEXTFILE** keyword (from Shell) or the icon tool type of the same name to let ConvertiLBM know the location of an ASCII text file containing the program's strings in the desired language.

For example, if the name of your file is "cilbm.txt" and it is in the "WORK:Config/" directory, you could run ConvertiLBM from Shell with a command line like this:

```
ConvertiLBM TEXTFILE=WORK:Config/cilbm.txt [other parameters here]
```

or set the tool type TEXTFILE=WORK:Config/cilbm.txt in the icon of the program.

Alternatively, you can simply put the text file in your S: directory and rename it to "ci.txt", as ConvertiLBM will search that by default.

Do not create such a text file from scratch by yourself, but use or simply translate into your language, if needed, one of those included with ConvertiLBM (typically the english one, supplied only for that).

Remember that the text file expedient, if used, overrides the "normal" localization method, even if locale.library is present. To avoid this, if you plan to use only the standard localization method, you must make sure that there is no "ci.txt" file in your system's S: directory (and obviously do not use the TEXTFILE keyword or tool type).

NOTE: If you make versions for other languages of the .catalog files, you are free (and welcome) to include them into the distribution directory of ConvertiLBM.

OTHER NOTE: When installing a new version of ConvertiLBM, make sure that you replace all localization files (the .catalog and/or ASCII file) with the new ones, otherwise the program could behave strangely.

1.15 Conclusion

7. CONCLUSION

After the debugging phase I did not discover any more errors; however if you should find any let me know. Write to:

Massimo Tantignone

Via Campagnoli, 4

28100 Novara (NO)

ITALY

E-Mail: Massimo_Tantignone@skylink.aare.net.ch

Write to this address also to give any suggestion about modifications and/or enhancements of the program, as well as to send to me any money contributions to support future versions (well, I tried that).

I don't have any more to say: make a good use of the program, make a bad use of it if you wish, modify it, DISTRIBUTE IT, do all you want with it (except selling it).

To have any information write to the above address.

Oh, and forgive me for my very poor english.

Massimo Tantignone, 15 february 1994

And remember... "Only AMIGA makes it possible!"
