

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	Documentazione di ConvertiILBM 2.1	1
1.2	Presentazione del programma	1
1.3	Scopo e motivazioni	2
1.4	Informazioni generali	4
1.5	already_know	4
1.6	all_users	5
1.7	Uso da Shell	6
1.8	Uso da Shell/Template	6
1.9	tke	6
1.10	Uso da Workbench	12
1.11	Uso da Workbench/Conversione da Workbench	12
1.12	Uso da Workbench/Visualizzazione da Workbench	17
1.13	wb_errors	18
1.14	Precisazioni sui tipi di conversione	18
1.15	Messaggi di errore	20
1.16	Localizzazione	23
1.17	Localizzazione/Per chi usa il primo metodo	23
1.18	Localizzazione/Per chi usa il secondo metodo	25
1.19	La storia del programma	25
1.20	Revisioni esistenti	27
1.21	Conclusione	30

Chapter 1

ConvertiILBM

1.1 Documentazione di ConvertiILBM 2.1

ConvertiILBM 2.1

di Massimo Tantignone

© 1989-1994 MT Soft

1. PRESENTAZIONE DEL PROGRAMMA
2. SCOPO E MOTIVAZIONI
3. INFORMAZIONI GENERALI
4. USO DA SHELL
5. USO DA WORKBENCH
6. PRECISAZIONI SUI TIPI DI CONVERSIONE
7. MESSAGGI DI ERRORE
8. LOCALIZZAZIONE
9. LA STORIA DEL PROGRAMMA
10. REVISIONI ESISTENTI
11. CONCLUSIONE

1.2 Presentazione del programma

1. PRESENTAZIONE DEL PROGRAMMA

Il programma ConvertiILBM permette di convertire qualsiasi file che sia in formato IFF ILBM in diversi altri formati, tra cui icone, Sprite, BOB, sorgenti in vari linguaggi e file binari grezzi, ed è anche in grado di visualizzarlo ed esaminare i suoi chunk principali.

Inoltre mette a disposizione un certo numero di opzioni per variare le caratteristiche di alcuni dei tipi di file che vengono prodotti.

Il programma gestisce file IFF ILBM in qualsiasi risoluzione e modo grafico, compresi HAM, EXTRA-HALFBRITE e overscan.

Non supporta ancora i nuovi modi grafici AGA poichè non possiedo un Amiga 4000 (ma è solo questione di tempo).

Tutte le funzioni di conversione sono utilizzabili sia da Shell che da Workbench; la visualizzazione da Workbench supporta anche l'uso della selezione estesa e dell'indicazione di ConvertiILBM come Default Tool nell'icona di un file ILBM.

Il programma è compatibile con qualsiasi sistema operativo di Amiga, funziona su sistemi PAL e NTSC, può salvare la configurazione preferita nella sua icona e può essere adattato a qualsiasi lingua (tramite la locale.library sotto 2.1 o superiori e con un altro metodo sotto 2.04 o inferiori).

ConvertiILBM risulterà utile soprattutto ai programmatori che desiderano includere immagini ed animazioni nei loro applicativi, ma anche gli altri utenti potranno trovare utile, ad esempio, la possibilità di convertire i propri disegni in icone per il Workbench.

Poichè con la versione 2.0 ConvertiILBM è stato quasi completamente riscritto e presenta notevoli differenze rispetto a ConvertiILBM 1.4 consiglio la lettura completa di questa documentazione anche a chi avesse già familiarità con la versione precedente.

1.3 Scopo e motivazioni

2. SCOPO E MOTIVAZIONI

Spesso i programmatori di Amiga si trovano nella necessità di inserire immagini bitmapped nelle loro applicazioni, ma per fare questo è necessario che tali immagini siano in un formato adeguato.

Infatti un programma potrebbe caricare da disco un file grafico IFF durante il suo funzionamento, ma solitamente si preferisce, per le piccole applicazioni, codificare i dati dell'immagine direttamente all'interno del programma eseguibile, oppure, nel caso di giochi o di programmi già molto estesi, caricare da disco file binari grezzi che sono più facili da gestire a livello di input/output.

In certi casi poi il formato è imposto dal linguaggio di programmazione che si usa; ad esempio in AmigaBASIC si possono leggere da disco i BOB o gli VSprite, utilizzabili nelle proprie animazioni, solo se il file che li contiene è strutturato come quelli prodotti dal programma ObjEdit.

Infine sia il programmatore che l'utente spesso hanno il desiderio o

la necessità di personalizzare le icone dei propri files, e per fare ciò normalmente occorre un programma di disegno che permetta di salvare il proprio lavoro sotto forma di icona per il Workbench (come IconEd).

Purtroppo non tutti i programmi di disegno permettono di salvare le immagini in tutti questi formati diversi, il che significa che quasi sempre occorre utilizzare un programma diverso per ciascuno di essi.

Questo può rivelarsi problematico, poiché sono pochi i programmi di disegno "dedicati" ad un certo formato che siano davvero validi, e anzi in certi casi si ha a disposizione un unico programma per di più non particolarmente affidabile (come ad esempio il già citato ObjEdit per i file di BOB o VSprite AmigaBASIC).

L'unica soluzione in questi casi consiste nell'usare un unico programma di disegno, ad esempio DeluxePaint, per creare le proprie immagini, e poi "convertirle" nel formato richiesto mediante l'utilizzo di altri programmi, generalmente di pubblico dominio. Il problema però così è solo spostato: si ricade infatti comunque nella necessità di avere un programma di conversione diverso per ogni formato che può servire.

Senza contare che, convertendo le proprie immagini all'esterno del programma con il quale sono state create, si corre il rischio di confondere il nome di due immagini diverse o di non ricordare l'esatto contenuto di un file, specialmente a distanza di tempo, per cui spesso serve anche un visualizzatore per dare un'occhiata alle immagini prima di convertirle.

Per risolvere tutti questi problemi ho creato ConvertiILBM.

ConvertiILBM riunisce in un unico programma le funzionalità di diversi altri convertitori e visualizzatori, di pubblico dominio e non, e quindi elimina anche la necessità di avere molti programmi di editing permettendo di usarne uno solo (ad esempio il DPaint) qualunque sia il tipo di file che alla fine volete ottenere.

Grazie a ConvertiILBM potrete eliminare dai vostri dischi o dal vostro hard disk i programmi:

IconEd

IconMerge

IE

LoadILBM-SaveACBM

ObjEdit

Gi

ZapIcon

e tutti i loro equivalenti, oltre a diversi semplici visualizzatori come Show, SeeILBM, Display, ecc.

Poichè però la visualizzazione operata da ConvertiILBM è molto semplice e serve solo a visionare i file che si stanno convertendo, il programma non può sostituire visualizzatori più potenti come ad esempio Mostra di Sebastiano Vigna o ViewTek di Thomas Krehbiel, attualmente i migliori in circolazione.

A parte questo, ora potete disegnare con il DPaint le vostre icone, i vostri BOB o Sprite da usare per le animazioni in AmigaBASIC, e in generale tutte le immagini che vorrete inserire nei vostri programmi.

Credo che usare il DPaint sia meglio che usare IconEd o ObjEdit, che oltre ad essere obsoleti sono anche poco potenti e decisamente scomodi per l'utente (per non parlare dei loro numerosi bug: non sono mai riuscito a terminare una sessione di lavoro con ObjEdit senza ricevere una visita del Guru).

Anche se con la versione 2.0 del sistema operativo il programma IconEd (ora IconEdit) è stato notevolmente migliorato, di certo non è ancora all'altezza del DPaint. Inoltre nonostante IconEdit permetta di importare file IFF, questi vengono "tagliati" se superano le dimensioni massime consentite (piuttosto ridotte). ConvertiILBM invece non impone limiti sulle dimensioni.

1.4 Informazioni generali

3. INFORMAZIONI GENERALI

3.1 Per chi conosce già ConvertiILBM 1.x

3.2 Per tutti gli utenti

1.5 already_know

3.1 Per chi conosce già ConvertiILBM 1.x

ConvertiILBM 2.x è una nuova versione del programma e non una semplice revisione. Rispetto alla versione 1.4 sono stati introdotti diversi miglioramenti, modifiche, aggiunte e correzioni.

Sono state aggiunte nuove opzioni, l'interfaccia utente grafica è stata completamente ridisegnata nell'aspetto e nel funzionamento secondo lo stile 2.0, la sintassi da Shell è cambiata e a sua volta rispetta il formato standard della Release 2, il programma consente la localizzazione con o senza la locale.library, sono ora supportati anche i linguaggi Modula-2 e Assembler, è possibile salvare la configurazione preferita nell'icona del programma, sotto 2.0 o superiori è abilitato finalmente

l'autoscrolling durante la visualizzazione di un disegno più esteso dello schermo visibile, sono presenti alcuni controlli per la formattazione del codice sorgente generato, e soprattutto sono stati eliminati diversi bug rendendo nel complesso il programma molto più solido.

Tutto questo mantenendo la compatibilità con il sistema operativo 1.3, anche se sotto 2.0 o superiori il programma presenta qualche caratteristica in più.

1.6 all_users

3.2 Per tutti gli utenti

ConvertiILBM attualmente può convertire un file IFF ILBM nei seguenti formati: file ACBM, file ILBM non compresso, sorgente (BASIC, C, Modula-2 o Assembler), file binario grezzo, sorgente C per uno sprite, icona (nuova o immagine alternativa), file di VSprite e file di BOB per AmigaBASIC.

La maggior parte dei tipi di conversione presenta poi delle opzioni specifiche che permettono di configurare il file prodotto; ad esempio si può decidere se aggiungere una mappa dei colori ad un sorgente o ad un file binario grezzo, oppure se la bitmap del file prodotto debba essere contigua o interleaved (intercalata), oppure ancora di quanti spazi debba essere l'indentazione delle righe di un sorgente.

Allo scopo di facilitare l'identificazione dei file ILBM desiderati prima di convertirli, il programma permette anche di visualizzare un file ILBM e di esaminare il contenuto dei suoi chunk più importanti.

Da CLI/Shell si può eseguire una operazione alla volta richiamando ConvertiILBM e specificando sulla linea di comando i nomi dei file di input e di output, il tipo di conversione e/o le eventuali opzioni desiderate.

Da Workbench, invece, è possibile eseguire più conversioni in un'unica sessione, utilizzando un comodo pannello di controllo che compare facendo doppio click sull'icona del programma.

Lavorando da Workbench è anche possibile salvare la configurazione desiderata del pannello nell'icona di ConvertiILBM, in modo che tale configurazione sia ripristinata ad ogni successiva esecuzione del programma. La configurazione salvata verrà invece ignorata utilizzando ConvertiILBM da CLI/Shell.

NOTA IMPORTANTE: Per potere usare il pannello di controllo sotto 1.3 o inferiori è necessario che nella directory LIBS: del proprio sistema sia presente una libreria PD di emulazione della gadtools.library, come

quella da me realizzata e inclusa con il programma. Se state ancora usando il sistema operativo 1.3, quindi, copiate nella vostra directory LIBS: il file "gadtools_34.library" che viene distribuito insieme a ConvertiILBM e rinominatelo "gadtools.library" (se non lo rinominate non verrà trovato dal programma, io non lo distribuisco direttamente con il nome di "gadtools.library" poichè sospetto che questo nome sia protetto da copyright).

Mi rendo conto che tutto ciò può essere seccante, ma questo è solo un altro motivo per cui DOVRETE aggiornare il vostro Amiga e passare almeno al sistema operativo 2.0 se non ad uno superiore.

1.7 Uso da Shell

4. USO DA SHELL

4.1 Template

4.2 Spiegazione del template

1.8 Uso da Shell/Template

4.1 Template

Il template di ConvertiILBM è il seguente:

**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**

Una keyword (parola chiave) del tipo CMAP=COLORMAP significa che è possibile indicare indifferentemente CMAP o COLORMAP (le keyword sono cioè equivalenti).

1.9 tke

4.2 Spiegazione del template

FROM

indica il nome del file IFF ILBM di input, cioè il file che deve essere convertito. Questo argomento deve essere sempre specificato.

Non si deve scrivere per esteso FROM, ma solo il nome del file.

Indicando solo questo argomento sulla linea di comando si provoca la visualizzazione del file.

TO

indica il nome del file di output, cioè il file prodotto dalla

conversione. Questo argomento deve essere specificato se non si richiede semplicemente la visualizzazione o l'esame del file di input. Non si deve scrivere per esteso TO, ma solo il nome del file.

CONVERSIONTYPE

serve per specificare il tipo di conversione desiderata. Non si deve scrivere per esteso CONVERSIONTYPE, ma solo il nome della conversione richiesta. I nomi delle conversioni sono i seguenti, insieme al loro significato:

ACBM Formato IFF simile all'ILBM ma non interleaved

ILBM Formato IFF ILBM non compresso

SOURCE Dati dell'immagine in BASIC, C, Modula-2 o Assembler

BYTES File binario con i dati ed eventuale colormap

SPRITE Sorgente in C con la definizione di uno sprite

ICON Produce un file .info o ne modifica uno già esistente

VSPRITE File binario usato da AmigaBASIC per le animazioni

BOB File binario usato da AmigaBASIC per le animazioni

nel caso dei tipi di conversione SOURCE e ICON, occorrerà indicare poi un altro parametro per specificare il linguaggio o il tipo di icona che si desidera.

Il default per questo parametro è SOURCE.

LANGUAGE

serve per specificare il linguaggio desiderato in caso di conversione in sorgente. Non si deve scrivere per esteso LANGUAGE, ma solo il nome del linguaggio voluto. I nomi dei linguaggi supportati sono i seguenti:

BAS=BASIC Linguaggio AmigaBASIC

C Linguaggio C

M2=MODULA2 Linguaggio Modula-2

ASM=ASSEMBLER Linguaggio assembler

Se non si indica anche la keyword **SOURCE** sulla linea di comando, questo parametro sarà ignorato.

Il default per questo parametro è C.

ICONTYPE

serve per specificare il tipo di icona desiderato in caso di conversione in icona. Non si deve scrivere per esteso ICONTYPE, ma solo il nome del tipo di icona. I (nomi dei) tipi di icona sono i seguenti:

TOOL Icona di un programma eseguibile

PROJECT Icona di un file di dati

DRAWER Icona di una directory (cassetto)

DISK Icona di un disco o device

TRASHCAN Icona del Trashcan

indicando invece APPEND non si creerà un nuovo file di icona, ma si appenderà l'immagine convertita (come immagine alternativa per l'icona selezionata) ad un file .info già esistente; se questo non sarà trovato il programma segnalerà un errore.

Se non si indica anche la keyword **ICON** sulla linea di comando, questo parametro sarà ignorato.

Il default per questo parametro è PROJECT.

SHOW

questo switch indica che si desidera visualizzare il file ILBM di input. Il contenuto del file verrà mostrato in uno schermo del tipo e della dimensione più adatti e si potrà terminare la visualizzazione premendo un qualsiasi tasto del mouse o della tastiera.

Non verrà eseguita alcuna conversione.

EXAM=EXAMINE

questo switch indica che si desidera esaminare il contenuto dei chunk principali del file ILBM di input. I chunk riconosciuti attualmente dal programma sono BMHD, CMAP, CAMG, GRAB e BODY; di quest'ultimo sarà naturalmente mostrata solo la lunghezza.

Non verrà eseguita alcuna conversione.

CMAP=COLORMAP

questo switch indica che si desidera che nel file di output prodotto, se applicabile, sia inclusa una colormap ricavata dal file ILBM di input.

I tipi di conversione che supportano questo parametro sono **SOURCE**, **BYTES** e **SPRITE**. Il formato della colormap varierà in base al tipo di conversione; in particolare nel caso di **BYTES** la colormap sarà appesa alla fine del file come una serie di word (2 byte per ogni colore contenenti il livello RGB, nello stesso formato usato dalla funzione LoadRGB4() della graphics.library).

Non specificando questo parametro non sarà inclusa alcuna colormap.

INTERL

questo switch indica che la bitmap del file prodotto da ConvertiILBM deve essere interleaved, come quella del file ILBM di partenza, piuttosto che contigua. Non tutti i tipi di conversione supportano questa opzione: attualmente il tipo di bitmap può essere scelto solo per i sorgenti e per i file binari grezzi. In tutti gli altri casi

la bitmap sarà forzatamente contigua o interleaved, in base al tipo di conversione, e questo parametro verrà ignorato.

Non specificando questo parametro la bitmap, se selezionabile, sarà contigua.

BHEADER

questo switch, riconosciuto dal tipo di conversione **BYTES**, indica che il file binario grezzo prodotto deve avere all'inizio tre word supplementari contenenti rispettivamente larghezza, altezza e profondità dell'immagine. Questo è utile se il file binario dovrà essere in seguito caricato in un array AmigaBASIC utilizzabile con l'istruzione PUT, che richiede appunto questo formato.

Se non si indica anche la keyword **BYTES** sulla linea di comando, questo parametro sarà ignorato.

Non specificando questo parametro non sarà aggiunto alcun header.

NOTRANSBACK

questo switch, riconosciuto dal tipo di conversione **BOB**, indica che lo sfondo del BOB, quando quest'ultimo è visualizzato ed animato dall'AmigaBASIC, deve essere opaco (di colore 0) e non trasparente.

Se non si indica anche la keyword **BOB** sulla linea di comando, questo parametro sarà ignorato.

Non specificando questo parametro lo sfondo del BOB sarà trasparente.

PAINTBRUSH

questo switch, riconosciuto dal tipo di conversione **BOB**, indica che il BOB, quando quest'ultimo è visualizzato ed animato dall'AmigaBASIC, non deve preservare lo sfondo su cui si sposta ma deve copiare la sua immagine in ogni nuova posizione, creando così una specie di "effetto pennello".

Se non si indica anche la keyword **BOB** sulla linea di comando, questo parametro sarà ignorato.

Non specificando questo parametro il BOB rispetterà lo sfondo su cui si muove senza modificarlo.

WIDTH=<numero di colonne>

con questa keyword si indica a ConvertiILBM la larghezza massima di una riga di testo nel listato sorgente che verrà creato (se il tipo di conversione è **SOURCE** o **SPRITE**).

ConvertiILBM andrà a capo ogni volta che scrivendo il file sorgente raggiungerà questa larghezza, senza però spezzare i singoli dati.

È consigliato impostare WIDTH con il numero massimo di colonne

visualizzate dal proprio text editor, così che esso possa mostrare il file sorgente nella sua interezza, senza dati "fuori campo".

I commenti non rispetteranno questa larghezza massima; questa indicazione serve soltanto per incolonnare in modo soddisfacente i dati in esadecimale che definiscono l'immagine ed eventualmente la colormap.

Specificando UNLIMITED (o qualunque cosa che cominci con "U", anche minuscolo) al posto del numero di colonne la larghezza della riga sarà illimitata, cioè ConvertiILBM andrà a capo solo dopo avere scritto nel sorgente un'intera riga della bitmap dell'immagine convertita.

Se non si indica anche la keyword **SOURCE** o la keyword **SPRITE** sulla linea di comando, questo parametro sarà ignorato.

Il valore di default per questo parametro è 76, e il valore minimo è 16.

TAB

questo switch indica che nel file sorgente prodotto l'indentazione deve essere realizzata con una TAB piuttosto che con una serie di spazi. Il numero di colonne che verrà effettivamente visualizzato caricando il file nel proprio text editor dipenderà naturalmente dalle impostazioni di quest'ultimo.

Se non si indica anche la keyword **SOURCE** o la keyword **SPRITE** sulla linea di comando, questo parametro sarà ignorato.

Non specificando questo parametro l'indentazione sarà costituita da spazi.

INDENT=<numero di spazi>

questa keyword può essere interpretata in due modi. Se non si è specificato anche **TAB**, cioè se l'indentazione nel sorgente prodotto deve essere realizzata con spazi, indica semplicemente il numero di spazi desiderato per essa.

Se invece si è specificato **TAB**, dovrebbe indicare il numero di spazi che corrispondono ad una tabulazione nel proprio text editor, in modo che ConvertiILBM sappia regolarsi nella valutazione della larghezza di una riga (a meno che non si sia specificato anche **WIDTH=UNLIMITED**, nel qual caso, essendo la larghezza della riga illimitata, questo parametro è del tutto ininfluenza).

Se non si indica anche la keyword **SOURCE** o la keyword **SPRITE** sulla linea di comando, questo parametro sarà ignorato.

Il valore di default per questo parametro è 3, e il valore minimo

è 0.

JOINLINES

questo switch indica che le righe di testo del listato sorgente prodotto devono essere quanto più lunghe possibile, compatibilmente con la larghezza massima di una riga (specificata con **WIDTH**), ed eventualmente anche unite tra loro per ottenere questo scopo. Ciò significa che sulla stessa riga di testo potranno trovarsi anche i dati di più righe dell'immagine. Non è consigliabile specificare insieme questo parametro e **WIDTH=UNLIMITED**, perchè in questo caso il sorgente prodotto sarebbe costituito da un'unica lunga riga di testo.

Se non si indica anche la keyword **SOURCE** o la keyword **SPRITE** sulla linea di comando, questo parametro sarà ignorato.

Non specificando questo parametro una singola riga del sorgente conterrà al più i dati di una sola linea dell'immagine convertita (e anche meno se la larghezza massima indicata con **WIDTH** non è sufficiente per questo).

TEXTFILE=<nome del file di testo>

questa keyword serve a specificare il nome del file di testo contenente le stringhe che il programma dovrà usare durante la sua esecuzione. Questo serve a localizzare il programma (adattarlo ad una particolare lingua) anche senza possedere la locale.library.

Il file di testo deve essere scritto in un formato specifico.

Vedere il paragrafo 8, "**Localizzazione**", per informazioni più dettagliate su questo argomento.

Il valore di default per questo parametro è "S:ci.txt".

GUI

indicando GUI sulla linea di comando non verrà eseguita alcuna conversione in modo immediato, ma comparirà il pannello di controllo come se il programma fosse stato fatto partire da Workbench.

Qualsiasi altro argomento presente sulla linea di comando sarà preso in considerazione e usato per configurare inizialmente il pannello, compresi i nomi dei file. I parametri non specificati non saranno letti dall'icona del programma, come accadrebbe se ConvertiILBM fosse lanciato veramente da Workbench, ma verranno assunti uguali al loro valore di default.

A differenza dell'analogo parametro "WB" di ConvertiILBM 1.4, GUI non è sperimentale e non presenta alcun rischio nell'uso.

GUI è l'acronimo di "Graphic User Interface", interfaccia utente

grafica.

Inoltre dando "ConvertiILBM ?" si otterrà la visualizzazione del template, mentre raddoppiando il punto interrogativo, come ad esempio in "ConvertiILBM ??" verrà mostrata la sintassi nel formato più tradizionale, come segue:

ConvertiILBM 2.1 di Massimo Tantignone - © 1994 MT Soft

Uso: ConvertiILBM <File di input> [<File di output>]

[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]

il che può essere utile se non ci si ricorda l'esatto nome di alcuni parametri.

Potete specificare i vari argomenti in qualsiasi ordine, l'unico requisito è che il nome del file di input venga prima di quello del file di output.

1.10 Uso da Workbench

5. USO DA WORKBENCH

L'uso di ConvertiILBM da Workbench è estremamente intuitivo e non occorrerebbe in realtà illustrarlo in modo dettagliato, comunque lo farò ugualmente. Gli utenti esperti probabilmente non hanno davvero bisogno di leggere quanto segue.

ConvertiILBM può essere utilizzato da Workbench in due modi: come convertitore (uso principale) o come visualizzatore.

5.1 Conversione da Workbench

5.2 Visualizzazione da Workbench

5.3 Errori durante la conversione o la visualizzazione

1.11 Uso da Workbench/Conversione da Workbench

5.1 Conversione da Workbench

Per avviare una sessione di conversioni con ConvertiILBM è sufficiente fare doppio click sulla sua icona. Una volta lanciato il programma in questo modo apparirà il pannello di controllo delle operazioni.

Nella descrizione del pannello userò, per riferirmi ai gadget e ai

menu, il loro nome in italiano; se intendete usare il programma in un'altra lingua (che conoscete) non dovrete comunque incontrare difficoltà; eventualmente provate ad usarlo prima almeno una volta in italiano, per acquisire familiarità con i suoi vari elementi.

Il pannello è suddiviso in quattro sezioni. In alto ci sono due gadget di tipo button e due gadget stringa che servono a specificare i nomi dei file di input e di output.

A sinistra nella zona centrale c'è una serie di otto gadget a mutua esclusione usati per selezionare il tipo di conversione desiderato.

A destra nella zona centrale c'è un sottopannello variabile il cui contenuto dipende dal tipo di conversione correntemente selezionato; questo sottopannello conterrà vari gadget utilizzabili per impostare opzioni specifiche di ogni determinato tipo di conversione. Per alcuni tipi di conversione questo sottopannello sarà completamente vuoto, il che significa semplicemente che non esistono (ancora) opzioni specifiche per essi.

In basso ci sono cinque gadget di tipo button, usati per varie operazioni, come avviare la conversione, visualizzare il file ILBM corrente oppure terminare il programma. Uno di essi permette di richiamare un ulteriore pannello, contenuto in una finestra separata, che serve a specificare alcune opzioni avanzate per la formattazione del sorgente.

Il pannello principale presenta anche un menu, "Progetto", contenente tre voci, che esamineremo tra breve.

Vediamo ora la funzione e l'uso di ogni gadget.

I due gadget stringa, come già detto, consentono di immettere il nome dei file di input e di output. Premendo <RETURN> mentre si sta scrivendo del testo in uno dei due, il cursore si sposta immediatamente nell'altro. Inoltre premendo <RETURN> quando nessuno dei due è selezionato il cursore apparirà nel primo, evitando di dover muovere il mouse per fare click su di esso (se volete il cursore nel secondo basterà premere <RETURN> una seconda volta).

Se non volete usare la tastiera, potete azionare uno dei due gadget di tipo button (pulsanti) che si trovano a fianco dei due gadget stringa, e cioè "File di input..." e "File di output...". Questo provocherà la comparsa di un file requester mediante il quale potrete selezionare il nome di file desiderato (anche se dovrete usare comunque la tastiera per inserire inizialmente il nome del file di output, a meno che non esista già un file con il nome desiderato che vogliate sovrascrivere).

Il file requester presentato sarà quello fornito dalla prima ad essere trovata delle seguenti librerie (cercate in questo ordine di preferenza):

asl.library, reqtools.library e arp.library.

Ovviamente sotto 1.3 o inferiori non sarà disponibile la asl.library e la scelta si ridurrà alle altre due librerie.

Se nessuna di queste librerie verrà trovata, i due pulsanti per i file requester saranno disabilitati.

Nota: reqtools.library è copyright © di Nico François.

Gli otto gadget a mutua esclusione del sottopannello fisso di sinistra (da questo momento li chiamerò "gadget MX" per semplicità) permettono di specificare il tipo fondamentale di conversione desiderata.

Essi sono:

File ACBM

File ILBM

Sorgente

Bytes

Sprite

Icona

VSprite

BOB

Solo uno di questi otto gadget potrà essere selezionato in un dato momento.

La selezione di uno di essi provocherà la modifica del sottopannello variabile di destra, dove compariranno, se disponibili, degli ulteriori gadget di vario tipo utilizzabili per indicare opzioni aggiuntive (relative al tipo di conversione corrente). Se il tipo di conversione selezionato non offre ulteriori opzioni la zona del sottopannello di destra apparirà vuota e rientrante (indicando così che non contiene niente di manipolabile da parte dell'utente).

Passiamo in rassegna i diversi gruppi di gadget che possono comparire nel sottopannello variabile.

Per il tipo di conversione "File ACBM":

Nessuno.

Per il tipo di conversione "File ILBM":

Nessuno.

Per il tipo di conversione "Sorgente":

· Una serie di quattro gadget MX utilizzati per selezionare il linguaggio del listato sorgente da prodursi. Essi sono:

BASIC

C

Modula-2

Assembler

- Un gadget di tipo cycle, "Bitmap:", tramite il quale indicare il tipo di bitmap desiderato, "Contigua" oppure "Interleaved".
- Un gadget di tipo checkbox, "Aggiungi colormap". Se selezionato nel sorgente prodotto sarà inclusa una mappa dei colori.

Per il tipo di conversione "Bytes":

- Un gadget di tipo cycle, "Bitmap:", tramite il quale indicare il tipo di bitmap desiderato, "Contigua" oppure "Interleaved".
- Un gadget di tipo checkbox, "Aggiungi colormap". Se selezionato nel file binario prodotto sarà inclusa una mappa dei colori.
- Un gadget di tipo checkbox, "Header BASIC". Se selezionato verrà aggiunto un header di 3 word in testa al file binario prodotto, contenente informazioni su larghezza, altezza e profondità dell'immagine.

Per il tipo di conversione "Sprite":

- Un gadget di tipo checkbox, "Aggiungi colormap". Se selezionato nel sorgente C prodotto sarà inclusa una mappa dei colori.

Per il tipo di conversione "Icona":

- Un gadget di tipo cycle tramite il quale indicare il tipo di operazione da svolgersi, cioè "Crea nuova icona" oppure "Appendi seconda immagine" (ad un'icona già esistente).
- Una serie di quattro gadget MX utilizzati per selezionare il tipo dell'icona da prodursi nel caso l'operazione indicata sia "Crea nuova icona". Essi sono:

Tool

Project

Drawer

Disk

Trashcan

Per l'operazione "Appendi seconda immagine" questi gadget MX saranno invece disabilitati.

Per il tipo di conversione "VSprite":

Nessuno.

Per il tipo di conversione "BOB":

- Un gadget di tipo checkbox, "Sfondo trasparente". Se selezionato il BOB prodotto avrà lo sfondo trasparente, altrimenti lo sfondo sarà solido e di colore 0.
 - Un gadget di tipo checkbox, "Effetto pennello". Se selezionato il BOB prodotto non preserverà lo sfondo durante i suoi movimenti
-

ma lascerà una traccia come un pennello del DPaint.

Passando alla parte inferiore del pannello, troviamo il pulsante "Mostra l'immagine", la cui selezione provocherà l'immediata visualizzazione del file di input indicato nell'apposito gadget stringa. Nel caso non sia indicato alcun file di input comparirà un messaggio di errore.

Questa possibilità è utile se si vuole controllare che il file da convertire sia esattamente quello desiderato; l'impiego di ConvertiILBM come visualizzatore, comunque, è possibile anche in altri modi (vedi [paragrafo 5.2](#)).

Il pulsante "Opzioni avanzate...", selezionabile solo se il tipo di conversione corrente è "Sorgente" oppure "Sprite", provoca la comparsa di un ulteriore pannello in una finestra separata, dotato di alcuni gadget utili a specificare particolari opzioni di formattazione del sorgente prodotto, come vedremo tra poco.

Il pulsante "Esamina il file..." fa visualizzare un requester con i dati più importanti relativi al file ILBM di input. Anche in questo caso è ovviamente necessario che ne sia indicato uno nell'apposito gadget.

Il formato in cui i dati sono presentati è assolutamente identico a quello usato in ambiente Shell.

Infine ci sono i due pulsanti "Converti" ed "Esci". Una volta che avrete indicato i nomi dei file di input e di output e selezionato i gadget opportuni per la conversione desiderata, quest'ultima verrà effettivamente eseguita premendo "Converti". Eventuali problemi o errori saranno presentati tramite requester. In ogni caso la fine del processo di conversione verrà sempre segnalata da un requester, che in certi casi fornirà anche alcune informazioni utili circa l'operazione appena eseguita.

Quando avrete terminato le vostre conversioni, premendo "Esci" farete terminare il programma. Potete ottenere lo stesso effetto anche azionando il gadget di chiusura della finestra o selezionando la voce "Quit" del menu "Progetto".

Nella finestra delle opzioni avanzate per il sorgente si potrà indicare se la larghezza massima in colonne di una riga di testo debba essere limitata o illimitata (con un gadget di tipo cycle) e nel primo caso anche la sua dimensione (con il gadget di tipo integer "Colonne").

Il gadget cycle "Tipo" permette di specificare il tipo di indentazione desiderato, cioè "Spazi" o "TAB", e il gadget integer "Caratteri" serve ad indicare la misura effettiva di tale indentazione.

Il gadget di tipo checkbox "Congiungi righe" corrisponde all'opzione **JOINLINES** relativa all'uso da CLI/Shell (vedi paragrafo 4.2 per dettagli su questa e su tutte le altre opzioni di formattazione).

Premendo "OK" le vostre modifiche saranno rese operative, mentre selezionando "Annulla" o chiudendo la finestra saranno ripristinati i valori precedenti delle opzioni.

Il menu "Progetto", per finire, contiene tre voci:

- "Salva configurazione": selezionando questa voce la configurazione corrente dei vari gadget (anche di quelli non visibili) verrà salvata sotto forma di tool type nell'icona del programma.

Questi tool type corrispondono alle parole chiave usate da Shell, anche se alcuni sono indicati in una forma alternativa più comoda da gestire per il programma. In ogni caso non dovrete mai modificare a mano questi tool types, ma solo usare questa voce di menu. L'unica eccezione è il tool type "TEXTFILE", che potete liberamente manipolare o aggiungere direttamente nella finestra di Info (Information) dell'icona se esso non fosse già presente (una volta aggiunto o modificato verrà lasciato inalterato da ConvertiILBM anche salvando altre volte la configurazione).

Sotto 2.0 o superiori il programma creerà un'icona per se stesso se non ne esistesse già una nella directory in cui esso si trova; sotto 1.3 o inferiori invece ciò non è possibile e nel caso non esista l'icona di ConvertiILBM questa voce di menu sarà disabilitata (questo può accadere solo se il pannello è stato richiamato da Shell con l'opzione **GUI**).

- "Autore...": fornisce informazioni sul programma e sull'autore.
- "Quit": vi lascio indovinare a cosa serve.

1.12 Uso da Workbench/Visualizzazione da Workbench

5.2 Visualizzazione da Workbench

Per visualizzare un file ILBM da Workbench occorre che esso e il programma ConvertiILBM siano entrambi dotati di icona. L'icona di ConvertiILBM dev'essere di tipo Tool e quella del file ILBM dev'essere di tipo Project.

La visualizzazione può venire effettuata in due modi:

a) Selezione estesa.

Facendo click una volta con il mouse sull'icona di ConvertiILBM e poi, tenendo premuto il tasto SHIFT, due volte sull'icona del

file ILBM, quest'ultimo verrà visualizzato. Si può anche fare click prima sul file e poi sul programma. Selezionando diversi file ILBM in questo modo (con un solo click su ognuno e doppio click sull'ultimo, sempre tenendo premuto SHIFT) i file verranno visualizzati in sequenza.

b) Default Tool.

Specificando "ConvertiILBM" (senza virgolette) come Default Tool nella finestra di Info (Information per il 2.0+) dell'icona del file ILBM, e selezionando poi il gadget "Save", ogni volta che si farà doppio click sull'icona stessa verrà richiamato il programma ConvertiILBM che visualizzerà il file. Se il programma non si trova nello stesso cassetto del file ILBM, è necessario ovviamente indicare il path (percorso) completo, ad esempio "DF1:Progs/ConvertiILBM" (sempre senza virgolette).

I due modi elencati possono anche essere combinati: se l'icona di un file ILBM contiene ConvertiILBM come Default Tool, si può effettuare una selezione estesa multipla anche senza fare click sull'icona di ConvertiILBM; è sufficiente che il file ILBM in questione sia compreso tra quelli selezionati.

1.13 wb_errors

5.3 Errori durante la conversione o la visualizzazione

Nel caso si verifichino dei problemi, ad esempio se il file non fosse in formato ILBM, oppure se mancasse la memoria necessaria per l'operazione, il programma visualizzerà un requester con un messaggio di errore. Se l'errore non è fatale non interromperà un'eventuale visualizzazione multipla; questa infatti riprenderà alla scomparsa del requester.

1.14 Precisazioni sui tipi di conversione

6. PRECISAZIONI SUI TIPI DI CONVERSIONE

È opportuna ora qualche precisazione su alcuni tipi di conversione:

La conversione in VSprite e in BOB produce un file binario compatibile con il formato usato dall'AmigaBASIC per questi oggetti d'animazione. Tale formato è lo stesso utilizzato dal programma ObjEdit, anche se ObjEdit non può creare BOB con lo sfondo opaco o dotati di "effetto pennello".

La conversione in file binario grezzo produce un file contenente semplicemente la bitmap dell'immagine convertita, senza nessuna altra informazione, a meno che non si specifichi anche BHEADER e/o CMAP (da CLI/Shell) o si selezionino i checkbox "Header BASIC" e/o "Aggiungi colormap" (da Workbench).

Nel primo caso viene generato un file corrispondente alla porzione di memoria contenuta in un array usato in AmigaBASIC con le istruzioni GET e PUT. Questo file dev'essere caricato in un array di interi, un elemento per volta (a meno di usare le funzioni della dos.library, ma questo per i programmatori avanzati).

Nel secondo caso viene appesa al file binario una mappa dei colori che verrà quasi sempre ignorata dai programmatori in BASIC, ma che da C (e anche da BASIC, usando la graphics.library) è molto comoda da usare con la funzione LoadRGB4() essendo già nel formato richiesto.

Specificando anche INTERL (da Shell) o impostando il cycle gadget "Bitmap:" a "Interleaved" (da Workbench) viene generata una bitmap interleaved e non contigua, per quanto questo solitamente non sia molto utile.

La conversione in sorgente produce un listato contenente i dati dell'immagine strutturati nel modo più comodo per essere gestiti nel particolare linguaggio che si è scelto. Anche in questo caso si possono specificare CMAP e/o INTERL ("Aggiungi colormap" e/o "Bitmap: interleaved") con risultati analoghi a quelli già descritti per i file binari.

La conversione in file ILBM non compresso non è molto utile, ma non mi costava niente inserirla e poi chissà, a qualcuno potrebbe servire...

Tenere presente che nel file generato NON saranno presenti tutti i chunk del file originale, ma solo i più importanti, e cioè BMHD, CMAP, CAMG, GRAB e naturalmente BODY. Lo stesso vale per la conversione in file ACBM (dove il chunk BODY è sostituito dal chunk ABIT).

La conversione in icona genera un file di icona del tipo specificato con il nome indicato PIÙ l'estensione .info: non è necessario usare tale estensione sulla linea di comando oppure nel gadget stringa per il nome del file di input, comunque potete farlo se preferite.

Se invece di un tipo di icona si indica APPEND (da Shell) o si imposta il cycle gadget apposito a "Appendi seconda immagine" (da Workbench), viene appeso un disegno ILBM ad una icona già esistente; ovviamente il disegno dovrebbe avere le stesse dimensioni dell'icona o potrebbero accadere spiacevoli inconvenienti. Vi consiglio di usare questa opzione solo con icone già create con ConvertiILBM, in modo che possiate disporre del file

ILBM di origine e quindi essere in grado di valutare le dimensioni in modo preciso. Potete creare un'icona con una doppia immagine (immagine alternativa) semplicemente usando ConvertiILBM in due tempi, prima creando un'icona con un'immagine e poi unendovi l'altra. Ricordate che ConvertiILBM NON è in grado di unire a un'icona un'altra icona, come il programma IconMerge, ma soltanto un disegno in formato ILBM (il risultato è comunque lo stesso).

La conversione in sprite produce un file C sorgente con la definizione di uno sprite (struttura SimpleSprite + dati dell'immagine).

Non tutti i file ILBM sono adatti a questo scopo; l'immagine deve avere non più di 2 bitplane e una larghezza uguale o inferiore a 16 pixel (tale requisito vale anche per la conversione in VSprite per AmigaBASIC).

Queste restrizioni saranno modificate nel modo opportuno se e quando rilascerò una versione di ConvertiILBM AGA-compatibile.

1.15 Messaggi di errore

7. MESSAGGI DI ERRORE

Nello spiacevole caso che si verifichi qualche inconveniente durante l'uso di ConvertiILBM, il programma tenterà, se possibile, di comunicare il problema tramite un messaggio di errore.

Esaminiamo ora i possibili messaggi di errore prodotti da ConvertiILBM insieme al loro significato.

"Errore: manca un parametro obbligatorio"

ConvertiILBM è stato lanciato da CLI/Shell senza argomenti, oppure non si è specificato il nome del file di input e/o del file di output. Se l'operazione richiesta è **SHOW** oppure **EXAMINE** è sufficiente indicare il nome del file di input, altrimenti occorrono entrambi.

"Errore: sconosciuto il tool type <tooltype>"

ConvertiILBM è stato lanciato con un argomento che non è in grado di interpretare. Esso può essere una keyword (da Shell) oppure un vero e proprio tool type contenuto nell'icona del programma.

Questo può accadere se, ad esempio, si specificano per errore tre nomi di file sulla linea di comando invece del massimo di due: il terzo nome di file è considerato una keyword che naturalmente non viene riconosciuta.

"Errore: non si apre la <nome>.library"

La libreria "<nome>.library" non è stata trovata dal programma.

Assicuratevi che tale libreria sia presente nella vostra directory

LIBS: di sistema (o in uno dei vari percorsi associati al nome logico LIBS: sotto 2.0 o superiori).

"Errore: non si apre la finestra"

Non è possibile aprire il pannello di controllo di ConvertiILBM, probabilmente per mancanza di memoria.

"Errore: non si apre lo schermo"

L'operazione di visualizzazione di un file ILBM non può essere eseguita poichè Intuition non riesce ad aprire lo schermo richiesto per l'immagine. Questo può dipendere a sua volta da un'insufficienza di memoria, ma è anche possibile che il sistema operativo non riconosca il modo grafico indicato nel chunk CAMG del file ILBM.

Sotto 2.0 o superiori a volte può essere necessario attivare un monitor di sistema appropriato per rimediare a questo errore.

"Errore: non si può aprire il file <nomefile> di input"

Il file IFF ILBM di input indicato non esiste oppure non può essere aperto perchè è già in uso da parte di un altro programma.

Questo errore si verifica anche se non è stato specificato alcun file di input (solo da Workbench con il pannello di controllo).

"Errore: questo programma non gestisce files IFF di tipo "LIST"

"Errore: questo programma non gestisce files IFF di tipo "CAT"

ConvertiILBM attualmente non è in grado di gestire concatenazioni di file ILBM come quelle raccolte in un file IFF LIST o IFF CAT.

Non è escluso che tale supporto venga aggiunto in futuro.

"Errore: <nomefile> non è un file IFF"

Il file di input indicato esiste ma non è in formato IFF.

Può anche darsi che il file sia stato compresso con PowerPacker o qualcosa di simile. La versione 2.1 di ConvertiILBM non supporta ancora questo tipo di file, ma state certi che una versione futura lo farà.

"Errore: <nomefile> non è un file ILBM"

Il file di input indicato esiste, è un file IFF ma non è di tipo ILBM. ConvertiILBM, come dice lo stesso nome, gestisce solo file di tipo ILBM.

"Errore: non si trova il chunk <nomechunk>"

Leggendo il file ILBM di input ConvertiILBM non ha trovato il chunk <nomechunk>, indispensabile per una corretta gestione del file.

"Errore: il chunk <nomechunk> è vuoto"

"Errore: il chunk <nomechunk> è corrotto"

Il chunk <nomechunk> del file ILBM di input ha lunghezza nulla o comunque diversa da quella indicata dopo il suo ID.

"Errore: memoria insufficiente per i dati del chunk "BODY""

"Errore: memoria insufficiente per la bitmap contigua"

"Errore: memoria insufficiente per la bitmap interleaved"

Non è stato possibile allocare abbastanza memoria per i dati su cui ConvertiILBM deve lavorare. Per garantire la massima velocità di esecuzione, il programma carica sempre l'intera bitmap di un file ILBM in memoria, e può anche dovere allocare una seconda zona di memoria per i dati in formato non compresso oppure contiguo.

Questo rende difficoltoso lavorare su file molto lunghi in condizioni di scarsa memoria.

"Errore: buffer insufficiente per l'immagine"

Questo è un errore interno che non dovrebbe MAI verificarsi.

Se capitasse, vi prego di contattarmi e di descrivermi in modo dettagliato le circostanze dell'accaduto.

"Errore: memoria insufficiente"

La memoria del vostro sistema è insufficiente per allocare anche le più semplici strutture dati su cui ConvertiILBM lavora.

Chiudete tutti gli altri programmi che stanno girando, eventualmente disconnettete tutti i drive esterni, e poi ritentate.

Se il problema persiste, fareste meglio ad acquistare un'espansione di memoria.

"Errore: non si trova l'icona del programma"

Questo errore si può verificare se, sotto 1.3, avete lanciato ConvertiILBM da Workbench e poi, mentre il programma girava, avete cancellato la sua icona. Sotto 1.3 ConvertiILBM non è in grado di creare una nuova icona per se stesso.

"Errore: il file <nomefile> non è adatto per uno sprite"

State tentando di convertire in sprite o in VSprite un file ILBM contenente un'immagine larga più di 16 pixel, o dotata di più di due bitplane.

"Errore: non si può aprire il file <nomefile> di output"

ConvertiILBM non riesce ad aprire il file di output su cui scrivere il risultato della conversione. Il nome che avete indicato potrebbe essere nullo o non valido, oppure riferirsi ad un dischetto protetto da scrittura.

"Attenzione: il file <nomefile> è già in formato non compresso"

State convertendo in file ILBM non compresso un file ILBM che è già non compresso. Questo è solo un avvertimento, non un errore: l'operazione viene svolta ugualmente. Non è detto che il file ILBM prodotto sia del tutto identico a quello di partenza (infatti di esso conterrà soltanto i chunk BMHD, CMAP, CAMG, GRAB e BODY).

1.16 Localizzazione

8. LOCALIZZAZIONE

A partire dalla versione 2.0 ConvertiILBM supporta la localizzazione.

In termini meno tecnici ciò significa che il programma può essere adattato a qualsiasi lingua si desideri, purchè si possenga un opportuno file di configurazione contenente le stringhe del programma nella lingua in questione.

Esistono due metodi per localizzare ConvertiILBM. Il primo, compatibile con qualsiasi sistema operativo, consiste nell'indicare al programma il nome di un semplice file in formato ASCII contenente le stringhe del programma organizzate in un certo modo, che tra poco vedremo.

Il secondo metodo, utilizzabile solo dal sistema operativo 2.1 in poi, è quello standard introdotto dalla Commodore e si basa sull'uso della locale.library e di un opportuno file in formato IFF detto "catalogo".

Chi possenga il sistema operativo 2.1 o uno superiore è fortemente consigliato di usare il metodo standard Commodore, in quanto certamente in futuro diventerà l'unico sistema usato (anche dai miei programmi).

Nel caso nessuno dei due metodi venga usato il programma presenterà delle stringhe di default in inglese, il che dovrebbe facilitare una più ampia diffusione di ConvertiILBM in tutto il mondo.

NOTA: Quando installate una nuova versione di ConvertiILBM accertatevi di sostituire tutti i file necessari per la localizzazione (file .catalog e/o file ASCII) con quelli nuovi, altrimenti il programma potrebbe presentare dei malfunzionamenti.

8.1 **Per chi usa il primo metodo**

8.2 **Per chi usa il secondo metodo**

1.17 Localizzazione/Per chi usa il primo metodo

8.1 Per chi usa il primo metodo (file di testo in formato ASCII)

Il formato del file di testo usato per configurare le stringhe di ConvertiILBM è molto semplice: per ogni stringa si trova un numero di identificazione su una riga e poi la stringa stessa sulla riga successiva. Non sono permesse righe vuote e la prima riga del file deve contenere un numero di identificazione. I commenti possono apparire solo dopo il numero di identificazione e fino al primo newline.

Esempio di un frammento del file:

(...)

14

Immagine

15 Questo è un commento

Dimensioni

(...)

Non dovrete comunque crearvi da soli questo file: infatti ho incluso con il programma alcuni file di questo tipo già pronti, e anche uno con le stringhe in inglese che dovrebbe essere facile da tradurre qualunque sia la vostra lingua di origine.

Nel caso non sia già incluso il file nella vostra lingua, dunque, usate quello in inglese come base per crearvelo. Naturalmente anche se FOSSE già incluso il file che vi serve potete sempre personalizzarlo come preferite. Ogni file già fornito riporta, sotto forma di commento, anche il testo originale in inglese per ogni stringa, per assicurare la correttezza della traduzione.

Con la versione 2.1 di ConvertiILBM sono forniti i file per le seguenti lingue:

inglese

italiano

Ulteriori file saranno inclusi in versioni future.

Se siete in possesso del file nella vostra lingua, potete fare in modo che ConvertiILBM usi il suo contenuto al posto delle sue stringhe di default semplicemente rinominandolo "ci.txt" e ponendolo nella vostra directory "S:" di sistema. In questo caso ConvertiILBM lo leggerà automaticamente ad ogni esecuzione. Se preferite invece dare un altro nome al file, potete comunicare al programma il path (percorso) alternativo tramite la keyword (o il tool type) TEXTFILE.

Ad esempio se il vostro file di testo si chiama "cilbm.str" e si trova nella directory WORK:Config, potete chiamare ConvertiILBM nel modo seguente:

ConvertiILBM TEXTFILE=WORK:Config/cilbm.str (qui il resto delle opzioni) oppure inserire il tool type "TEXTFILE=WORK:Config/cilbm.str" nell'icona del programma mediante la finestra di Info (Information) relativa all'icona stessa.

1.18 Localizzazione/Per chi usa il secondo metodo

8.2 Per chi usa il secondo metodo (locale.library e file di catalogo)

Se siete uno dei fortunati possessori di un sistema operativo uguale o superiore al 2.1, potete (e dovrete) usare la locale.library per localizzare ConvertiILBM.

In questo caso non dovete fare altro che copiare il file .catalog relativo alla vostra lingua nell'appropriata sottodirectory della vostra directory LOCALE:Catalogs.

Ad esempio, se la vostra lingua è l'italiano, potete copiare il file "italiano/convertiilbm.catalog" (che si trova nella directory di distribuzione di ConvertiILBM) in "LOCALE:Catalogs/italiano/" ottenendo così il file "LOCALE:Catalogs/italiano/convertiilbm.catalog".

Se il file che vi serve non è fornito purtroppo dovete ricorrere al primo metodo (paragrafo 8.1) fino a quando non riuscirete a procurarvelo, a meno che possediate il programma per sviluppatori "CatComp", nel qual caso potete crearvelo da soli operando sul file standard di descrizione "convertiilbm.cd" (fornito anch'esso insieme a ConvertiILBM).

Se siete sviluppatori dovrete conoscere già i dettagli relativi a questa operazione, pertanto non li tratterò in questa sede.

Per finire, ricordate che dovete usare l'editor "Locale" delle preferenze per impostare la lingua globale del sistema, altrimenti il programma funzionerà comunque in inglese.

Nota: il primo metodo ha la precedenza sul secondo, cioè se avete un file "ci.txt" in S: oppure usate la keyword TEXTFILE il programma userà il file ASCII e ignorerà il vostro file .catalog anche se presente.

1.19 La storia del programma

9. LA STORIA DEL PROGRAMMA

Ho scritto ConvertiILBM per due motivi: il primo è che da tempo desideravo farlo ma non ne avevo mai trovato l'occasione; il secondo è che finalmente mi è capitata l'occasione.

Mi spiego meglio: qualche anno fa (nel 1989) sono riuscito a procurarmi una copia con tanto di sorgente dell'allora famoso programma di pubblico dominio Gi, di Michael J. Farren, che, secondo quanto si diceva, era in grado di convertire un file ILBM di tipo Brush (DPaint) in un frammento di codice in C contenente i dati dell'immagine descritta nel file ILBM.

La mia prima prova è stata catastrofica: ho convertito un semplice

Brush a quattro colori e di piccole dimensioni e ho scritto un piccolo programma in C che visualizzasse l'immagine con la funzione DrawImage() della intuition.library.

Non mi aspettavo che il programma funzionasse subito; tuttavia questa volta il programma HA funzionato subito... il problema era che l'immagine visualizzata non era esattamente la stessa che io avevo disegnato con il DPaint (per usare un eufemismo; in realtà non si poteva neanche definire un'immagine). Poi ho scoperto che mi ero dimenticato di mettere i dati grafici nella RAM CHIP, e così ho tirato un sospiro di sollievo, mi sono detto "ah, è per questo", e ho ricompilato il programma (questo mi ha anche portato a fare interessanti scoperte su alcuni strani effetti collaterali dell'opzione -a del compilatore Lattice 4.0 che allora usavo... ma questa è un'altra storia).

Ci credereste? Neanche stavolta veniva visualizzata un'immagine anche solo lontanamente riconoscibile! C'era una sola cosa da fare: con infinita pazienza ho trasformato l'array C generato da Gi in una serie di DATA in AmigaBASIC, e poi ho scritto un programma (sempre in AmigaBASIC) che visualizzasse l'immagine usando l'istruzione PUT.

Niente da fare neanche stavolta: questo significava che i dati erano davvero scorretti e non era colpa dei miei programmi di visualizzazione.

Allora ho fatto un listato del sorgente Gi.c che mi sono portato al mare, e lì esaminandolo ho trovato l'errore: l'autore, scrivendo il programma nel 1985, all'alba dell'Amiga e dello standard IFF, non aveva rispettato la fondamentale regola che dice "niente rimarrà sempre uguale".

In altre parole, aveva fatto alcune assunzioni sul formato dei file ILBM di Brush che potevano essere valide allora, ma adesso non lo sono più.

Il formato dei file di Brush è cambiato nel corso degli anni e delle successive revisioni di DPaint, e se il programma funzionava con i Brush prodotti da DPaint I ora non funziona più con quelli prodotti da DPaint II, DPaint III o DPaint IV.

Così, in un primo tempo mi sono limitato a correggere il programma rendendolo universalmente valido (non è che ci volesse molto) e l'ho chiamato NewGi.c, ma poi mi sono chiesto: già che ci sono, perchè non scrivere io stesso un programmino simile, magari aggiungendo qualche opzione in più?

Detto fatto: così nacque ConvertiILBM, in grado di trattare i file ILBM in 15 modi diversi (in seguito aumentati), compresa la visualizzazione.

Inoltre il mio programma non era più limitato ai brush, ma funzionava anche con schermate intere.

Recentemente, dopo l'avvento delle release 2.1 e 3.0 del sistema operativo di Amiga, mi sono reso conto che l'intero programma andava rivisto e aggiornato sia nell'aspetto che nel funzionamento per restare al passo con i tempi.

Da quest'opera di restauro è nato ConvertiILBM 2.0, che condivide pochissimo codice con le versioni precedenti, essendo una riscrittura quasi completa del programma.

La versione 2.1 è solo un aggiornamento minore che corregge qualche bug e migliora leggermente l'estetica del programma.

L'evoluzione di ConvertiILBM non è certo finita: ho già numerose idee circa ulteriori modifiche e miglioramenti che intendo apportare in futuro al programma, come supporto dei chip AGA, conversione pilotata da file script, uso dell'ARexx, conversione solo di parti dell'immagine, gestione dei file IFF LIST e CAT, nuovi tipi di conversione, help in linea, uso di AppIcon e AppMenu, funzionamento in background, supporto dei file compressi con PowerPacker, uso dei datatypes, GUI font-sensitive, ecc.

Non aspettatevi comunque di vedere tanto presto nuove versioni del programma poiché in questo periodo devo dedicare la maggior parte del mio tempo agli studi universitari, e inoltre ho anche molti altri progetti in sospeso che intendo realizzare prima.

Ma ConvertiILBM 2.2 (o 3.0!) ci sarà prima o poi, questo è certo.

1.20 Revisioni esistenti

10. REVISIONI ESISTENTI (in ordine cronologico dal basso verso l'alto)

2.1 Corretto un insidioso bug nell'operazione di esame dei file con più di 32 colori (grazie a Luigi Callegari). Inoltre ora sotto 3.0 o superiori il programma fa uso dei menu NewLook. L'estetica dei requester per i messaggi è migliorata (spero!) e l'eseguibile è più corto di 796 bytes.

2.0 Quasi completamente riscritto. Apportato un numero considerevole di modifiche. Seguono i dettagli.

- Ora l'interfaccia utente grafica è completamente conforme allo stile 2.0 nell'aspetto e nel funzionamento, infatti usa la gadtools.library per la generazione e la gestione dei gadget e dei menu.

- La sintassi da Shell è cambiata ed è conforme allo stile 2.0 utilizzando keyword al posto delle meno comprensibili opzioni inizianti con il trattino (-). Inoltre ora con il comando

"ConvertiILBM ?" viene mostrato il template secondo lo standard dei comandi AmigaDOS.

- Ora si può decidere se includere o no la colormap nei sorgenti e nei file binari, mentre prima questo veniva fatto in ogni caso.

- La conversione in BOB presenta due opzioni supplementari, per creare BOB con lo sfondo opaco e BOB che lasciano una traccia come un pennello.

- Il pannello di controllo presenta ora anche un menu.

- Ora è possibile salvare la configurazione corrente del pannello di controllo nell'icona del programma (come tool types). Ad ogni successiva esecuzione da Workbench la configurazione salvata verrà ripristinata. Inoltre il programma può creare un'icona per se stesso nel caso questa non esistesse nella sua directory (questo solo sotto 2.0 o superiori).

- Sistemati alcuni (e spero finalmente ULTIMI) problemi di visualizzazione. Pare che sotto 1.3 o inferiori ci siano problemi ad aprire schermi più larghi di 672 pixel, così chi ancora usa questo obsoleto sistema operativo purtroppo dovrà accontentarsi di questo massimo overscan orizzontale.

Inoltre il massimo overscan verticale in modo NTSC (in interlace) è di 482 pixel, ma questo non dovrebbe essere un problema. In questi casi le immagini più larghe o più alte verranno tagliate, ma come ho già precisato, ConvertiILBM non è un visualizzatore: l'opzione "Mostra immagine" serve solo per farsi un'idea del file che si sta convertendo, e per questo scopo funziona adeguatamente. Se poi volete anche un buon visualizzatore, usate "Mostra" di Sebastiano Vigna.

- Sistemato un ulteriore problema di visualizzazione nato con il sistema operativo 3.0: con i precedenti ConvertiILBM sotto 3.0 era necessario attivare il monitor non di default (NTSC sotto PAL, PAL sotto NTSC) per visualizzare immagini in quel formato.

Ora non è più necessario in quanto sotto 2.0 o superiori ConvertiILBM passa attraverso il monitor di default (usando un DisplayID invece che NewScreen.ViewModes) per aprire lo schermo voluto. Nota: probabilmente una delle prossime versioni di ConvertiILBM supporterà anche i modi DoublePAL e DoubleNTSC, ma non prendete comunque questo per garantito.

- Sotto 2.0 (o superiori) è ora supportato l'autoscrolling

degli schermi. Ciò permette di esaminare l'intera immagine visualizzata nel caso questa fosse più estesa della zona visibile.

- Ora i file ACBM e ILBM non compressi generati da ConvertiILBM contengono, se derivano da un brush, anche il chunk GRAB.

- ConvertiILBM non ha più problemi di stack! Non è più necessario indicare uno stack enorme nell'icona di ConvertiILBM. Ora il programma è in grado di allocare automaticamente lo stack che gli occorre durante l'esecuzione, e non provocherà più crash dovuti a Stack Overflow (grazie al compilatore SAS!).

- Ora sotto 1.3 viene usato il file requester fornito dalla reqtools.library, se quest'ultima è presente. Molto meglio del requester ARP, che comunque continua ad essere supportato.

1.4 Ora il supporto del Workbench è completo. Tutte le funzioni di conversione possono essere utilizzate da Workbench, e per l'utente medio questo è certamente più comodo. Inoltre ora è consentita la visualizzazione multipla mediante la selezione estesa (la versione 1.35 permetteva di visualizzare un solo file per volta). Infine, adesso ConvertiILBM può gestire anche i file ILBM contenenti uno stencil (stile DPaint).

Questa è la prima versione distribuita nel Pubblico Dominio.

1.35 Finalmente ConvertiILBM è in grado di supportare il Workbench, sia pure (per ora) solo come visualizzatore. In una futura versione verrà implementata un'interfaccia grafica allo scopo di dirigere da Workbench anche tutte le altre operazioni di conversione.

1.31 Corretto un bug che causava una visita del Guru ogni volta che si tentava di convertire in file di Sprite o VSprite un'immagine ILBM di formato non adatto (solita storia, "freeing memory that is already free").

1.3 Aggiunto il supporto dell'overscan (fino a 360x282 in bassa risoluzione o a 720x564 in alta risoluzione, oltre ovviamente a tutte le risoluzioni intermedie). In generale il programma tenta di visualizzare quanto più possibile delle bitmap più grandi dello schermo.

1.22 Ora l'immagine visualizzata (se è uno schermo e non un brush) viene posizionata orizzontalmente il più possibile al centro dello schermo (utile nel caso di WB overscan).

1.21 Incredibilmente, fino a questo momento la visualizzazione dei

file HAM e EHB (teoricamente possibile fino dalla versione 1.01) NON veniva ancora eseguita correttamente. Ora finalmente sembra che tutto funzioni.

1.2 Ora l'opzione -e prende in considerazione anche il chunk GRAB e finalmente viene segnalata nella sintassi l'esistenza dell'opzione -itc tra quelle disponibili (l'opzione era presente fino dalla versione 1.0). Forse avrei dovuto chiamare questa versione "1.12", ma non è il caso di sottilizzare.

1.11 Ottimizzata la gestione della memoria.

1.1 Totalmente riscritto ponendo quanto più codice possibile in funzioni generiche che in una futura versione verranno estratte ed inserite in una libreria. Aggiunta la nuova opzione -e che permette di esaminare il contenuto di un file ILBM. Inoltre migliorati il parsing della linea di comando e il trattamento degli errori.

1.02 Ora anche il tasto destro del mouse permette di uscire dalla visualizzazione (in precedenza causava la cancellazione della immagine sotto la title bar dello schermo).

1.01 Aggiunta la possibilità di gestire file di tipo HAM o EHB. Corretto un errore di visualizzazione delle bitmap più estese dello schermo visibile.

1.0 La versione originale. In grado di trattare i file ILBM in 15 modi diversi, compresa la visualizzazione. L'opzione -dc permette di ottenere gli stessi risultati del programma Gi di Michael J. Farren. Inoltre funziona con ogni tipo di file ILBM (anche non brush) prodotto da qualsiasi versione del DPaint o di tutti gli altri programmi di disegno.

1.21 Conclusione

11. CONCLUSIONE

Al termine della fase di debugging non ho più rilevato alcun errore degno di nota; comunque se ne trovaste fatemelo sapere. Scrivete a:

Massimo Tantignone

Via Campagnoli, 4

28100 Novara (NO)

ITALIA

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

Scrivete a questo indirizzo anche per qualsiasi suggerimento circa

modifiche e/o miglioramenti del programma, e per inviare eventuali contributi in denaro per finanziare future versioni (qui mi sto rivolgendo agli utenti che non vivono in Italia, SO BENE che di norma gli Italiani non sprecano il loro denaro in questo modo). Tale contributo è assolutamente volontario, infatti il programma è freeware e non shareware.

Non rimane altro da dire: fate buon uso del programma, fatene cattivo uso se volete, modificatelo, DISTRIBUITELO, fatene quello che vi pare (tranne venderlo).

NOTA: Chiunque realizzasse versioni per altre lingue della documentazione o dei file .catalog è libero di includerle nella directory di distribuzione, anche se sarebbe preferibile che prima li inviasse a me per un controllo.

Per informazioni scrivete all'indirizzo di cui sopra.

Massimo Tantignone, 15 febbraio 1994

E ricordate... "Solo AMIGA lo rende possibile!"
