

GadTools34

COLLABORATORS

	<i>TITLE :</i> GadTools34		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 17, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	GadTools34	1
1.1	Documentazione di gadtools.library 34.1 e GTE.lib	1
1.2	Introduzione	1
1.3	Utilizzo di gadtools.library 34.1	2
1.4	Utilizzo di GTE.lib	3
1.5	Motivo dell'esistenza di queste due librerie	3
1.6	Dettagli tecnici su GTE.lib	4
1.7	Esempio di utilizzo di GTE.lib	5
1.8	Nota importante circa GTE.lib	6
1.9	Nota importante circa gadtools.library 34.1 e GTE.lib	6
1.10	Differenze	7
1.11	Conclusione	7

Chapter 1

GadTools34

1.1 Documentazione di gadtools.library 34.1 e GTE.lib

GadTools Library 34.1 e GTE.lib 1.0

di Massimo Tantignone (27/01/93)

Una produzione MT Soft

1. **INTRODUZIONE**
2. **UTILIZZO DI GADTOOLS.LIBRARY 34.1**
3. **UTILIZZO DI GTE.LIB**
4. **MOTIVO DELL'ESISTENZA DI QUESTE DUE LIBRERIE**
5. **DETTAGLI TECNICI SU GTE.LIB**
6. **ESEMPIO DI UTILIZZO DI GTE.LIB**
7. **NOTA IMPORTANTE CIRCA GTE.LIB**
8. **NOTA IMPORTANTE CIRCA GADTOOLS.LIBRARY 34.1 E GTE.LIB**
9. **DIFFERENZE**
10. **CONCLUSIONE**

1.2 Introduzione

1. INTRODUZIONE

Gadtools.library 34.1 è una versione 1.3-compatibile dell'omonima libreria presente normalmente nel Kickstart 2.04 o superiori.

Per essere precisi, è una libreria di tipo shared che si propone di emulare la gadtools.library V37 e funziona anche sotto sistemi operativi anteriori al 2.0, come appunto l'OS 1.3.

GTE.lib è invece una libreria di moduli oggetto per il linker e svolge lo stesso compito.

Queste due librerie di pubblico dominio permettono di scrivere e/o usare programmi che usufruiscano anche sotto 1.3 dei molti vantaggi offerti dalla

gadtools.library nella costruzione e gestione della propria interfaccia utente grafica (purché ovviamente tali programmi non richiedano altre funzionalità specifiche dell'OS 2.0).

Entrambe le librerie sono dedicate principalmente ai programmatori, che potranno distribuire liberamente gadtools.library 34.1 insieme ai loro programmi, oppure collegare direttamente il codice di GTE.lib a questi ultimi (mediante il linker).

Se tuttavia fossero in circolazione programmi già fatti che usino la gadtools.library E NESSUN' ALTRA LIBRERIA O FUNZIONE DEL 2.0, anche questi dovrebbero funzionare dopo l'installazione di gadtools.library 34.1 nella propria directory LIBS: del Workbench 1.3 (ovviamente tali programmi possono essere individuati solo sperimentalmente).

In questo senso gadtools.library 34.1 può risultare utile anche all'utente finale (anche se dubito che al momento esistano già programmi del genere).

Nella directory di distribuzione di gadtools.library 34.1 e di GTE.lib ho incluso due programmi di test: uno (TestLinked) è stato linkato con GTE.lib e perciò contiene il codice di emulazione di gadtools, invece l'altro (TestShared) fa uso della gadtools.library esterna. Il primo può essere eseguito in qualsiasi momento, mentre per eseguire il secondo è necessario installare prima gadtools.library 34.1 come specificato nel prossimo paragrafo (oppure lanciarlo sotto 2.0 o superiori, ma in quel caso userebbe la VERA gadtools.library).

Nota: a parte il formato (shared library o linker library) le due librerie sono ASSOLUTAMENTE IDENTICHE nel funzionamento, infatti sono state ottenute dagli stessi sorgenti.

1.3 Utilizzo di gadtools.library 34.1

2. UTILIZZO DI GADTOOLS.LIBRARY 34.1

Per utilizzare gadtools.library 34.1 è sufficiente copiarla nella directory LIBS: del proprio Workbench 1.3 (floppy o hard disk). Ogni programma che tenti di aprire la gadtools.library funzionerà da questo momento anche sotto 1.3 e aprirà questa libreria invece di quella "vera" del sistema operativo 2.0, senza accorgersi della differenza.

Ovviamente, sotto 1.3, il programma non deve fare uso di altre librerie riservate al 2.0 diverse dalla gadtools.library.

Questa libreria NON va invece copiata nella directory LIBS: di un Workbench 2.0, poichè il sistema operativo 2.0 dispone già di una propria

gadtools.library.

Nota: la libreria è distribuita con il nome di "gadtools_34.library" per motivi di copyright. Dopo averla copiata in LIBS: dovete rinominarla esattamente "gadtools.library" altrimenti non funzionerà.

Se non volete copiare manualmente la libreria potete lanciare lo script di installazione automatica chiamato "InstallGadTools34".

1.4 Utilizzo di GTE.lib

3. UTILIZZO DI GTE.LIB

GTE.lib, essendo una libreria per il linker, risulta utile solo ai programmatori.

Per sfruttare questa libreria è sufficiente scrivere il proprio sorgente come lo si scriverebbe se si usasse la vera gadtools.library, ma antepoendo il prefisso "NL_" (enne elle sottolineatura) al nome di ogni funzione di gadtools chiamata.

Naturalmente dopo la compilazione occorre eseguire il linking con GTE.lib, e si otterrà un programma funzionante sotto qualsiasi sistema operativo e avente un'interfaccia utente che emula la gadtools.library V37 sotto 1.x mentre fa uso direttamente di quest'ultima sotto 2.x.

Antepoendo il prefisso "EF_" (e effe sottolineatura) invece di "NL_", il programma risulterà di dimensioni leggermente inferiori ma funzionerà in emulazione ANCHE sotto 2.x.

In entrambi i casi occorre #includere nel proprio sorgente il file header "nl_gadtools.h" fornito insieme a GTE.lib.

1.5 Motivo dell'esistenza di queste due librerie

4. MOTIVO DELL'ESISTENZA DI QUESTE DUE LIBRERIE

La gadtools.library è certamente tra le più utili librerie fornite a partire dalla versione 2.0 del sistema operativo di Amiga, poiché permette a tutti i programmi di presentare delle interfacce utente grafiche che siano consistenti e simili tra loro nell'aspetto e nel funzionamento.

Purtroppo, per quanto utile, questa libreria non è compatibile con il sistema operativo 1.3 (o precedenti), che, sebbene obsoleto, rimarrà di certo in circolazione sui più vecchi Amiga ancora per qualche anno.

Ogni nuova applicazione che faccia uso della gadtools.library per la propria interfaccia utente risulta pertanto inutilizzabile sotto l'OS 1.3,

anche se non utilizza nessun'altra funzionalità specifica del 2.0.

Per risolvere questo problema ho scritto una libreria che emula la gadtools.library V37 e che può essere integrata nei propri applicativi.

In effetti ho realizzato due versioni di essa; una shared library (gadtools.library 34.1) e una libreria di moduli oggetto per il linker, chiamata GTE.lib (GadTools Emulation Library).

1.6 Dettagli tecnici su GTE.lib

5. DETTAGLI TECNICI SU GTE.LIB

Occorre qualche precisazione circa l'utilizzo di GTE.lib.

Al momento è disponibile solo la versione adatta al sistema SAS/C 6.0.

Non ritengo questa una limitazione in quanto prima o poi tutti i programmatori "seri" dovranno aggiornarsi a questa versione (ed ora hanno un motivo di più per farlo).

Usando il compilatore SAS 6.0 si può utilizzare GTE.lib con
SC LINK LIB=GTE.lib <Nomi dei propri sorgenti>.

La GTE.lib supporta ed emula tutte le funzioni della gadtools.library versione 37. Tali funzioni vengono fornite in due versioni: le funzioni con prefisso "EF_" e quelle con prefisso "NL_".

Le prime sono le vere e proprie funzioni di emulazione e chiamandole sotto qualsiasi sistema operativo (1.x o 2.x) verrà eseguito il codice che emula la gadtools.library.

Le altre sono dette "funzioni di interfaccia": esse, se chiamate, decideranno in base al sistema operativo sotto cui il programma sta girando se eseguire il codice di emulazione (chiamare cioè le funzioni EF_) oppure saltare direttamente alle vere funzioni della gadtools.library.

In breve, aggiungendo il prefisso NL_ a tutte le chiamate a funzioni di gadtools nel proprio codice sorgente ed eseguendo poi il linking con GTE.lib, si otterrà un programma che usa la vera gadtools.library sotto il sistema operativo 2.0 (o superiori) mentre esegue il codice di emulazione della stessa sotto 1.3.

Questo naturalmente comporta l'aumento delle dimensioni del proprio eseguibile di circa 30-40K.

In alternativa si possono compilare due versioni dello stesso sorgente, una utilizzante solo autentiche chiamate a gadtools, l'altra chiamante solo le funzioni di emulazione, senza passare attraverso quelle di interfaccia (il cui codice oggetto non verrà quindi unito al programma) e realizzare due versioni dello stesso applicativo, una di dimensioni

inferiori funzionante solo sotto 2.x, e un'altra più lunga e funzionante sotto entrambi i sistemi operativi ma sempre e comunque in emulazione.

Sarà in questo caso dell'utente la scelta circa quale utilizzare.

Naturalmente tutto ciò è valido solo assumendo che il programma non faccia uso di nessun'altra nuova funzione o libreria dell'OS 2.0 all'infuori delle funzioni della gadtools.library.

Nota: se l'aumento di dimensioni dell'eseguibile non è accettabile l'unica alternativa è usare la versione shared della libreria di emulazione (gadtools.library 34.1).

1.7 Esempio di utilizzo di GTE.lib

6. ESEMPIO DI UTILIZZO DI GTE.LIB

Vediamo ora un esempio per chiarire le idee:

Un classico programma che utilizzi gadtools potrebbe ad esempio contenere questo frammento:

```
if (gad = CreateContext(&glist))
{
gad = CreateGadget(MX_KIND,gad,&ng,GTMX_Labels,labels,TAG_END);
}
```

e sarebbe utilizzabile solo sotto 2.0 o superiori.

Usando GTE.lib si può modificarlo in

```
if (gad = NL_CreateContext(&glist))
{
gad = NL_CreateGadget(MX_KIND,gad,&ng,GTMX_Labels,labels,TAG_END);
}
```

rendendolo funzionante sotto qualsiasi sistema operativo, in emulazione sotto 1.x e chiamando la vera gadtools.library sotto 2.x.

Alternativamente si può modificarlo così:

```
if (gad = EF_CreateContext(&glist))
{
gad = EF_CreateGadget(MX_KIND,gad,&ng,GTMX_Labels,labels,TAG_END);
}
```

Anche questa versione è funzionante con ogni sistema operativo, ma operando sempre in emulazione anche sotto 2.x, il che permette di diminuire leggermente le dimensioni dell'eseguibile, ma ovviamente non è il meglio in termini di efficienza e impedisce al programma di trarre vantaggio da eventuali futuri miglioramenti della vera gadtools.library.

Quale metodo usare è ovviamente una scelta del programmatore; ciò che

conta, se si usa GTE.lib, è rispettare sempre le seguenti cinque regole:

- #includere SEMPRE nel proprio sorgente il file "nl_gadtools.h", fornito insieme a GTE.lib;
- NON usare mai nel proprio programma altre funzioni o librerie compatibili solo con il 2.x oltre a quelle della gadtools.library;
- NON usare più di un metodo nello stesso programma: o usare tutte vere funzioni di gadtools, o tutte funzioni EF_, o tutte funzioni NL_;
- APRIRE sempre la gadtools.library se si è sotto 2.x e si usano le funzioni NL_ (o direttamente quelle vere, ma questo è ovvio);
- APRIRE sempre la intuition.library e la graphics.library.

1.8 Nota importante circa GTE.lib

7. NOTA IMPORTANTE CIRCA GTE.LIB

Esaminando il file GTE.lib con "OML GTE.lib I" si noterà che contiene anche molte altre funzioni oltre a quelle dell'emulazione di gadtools.

Queste sono utilizzate dalle funzioni di emulazione stesse e per il momento NON dovete tentare di chiamarle direttamente.

In futuro distribuirò queste funzioni (e molte altre ancora) in una libreria linked separata completa di documentazione e forse anche in una shared library.

Le uniche funzioni (presenti sia in versione NL_ che in versione EF_) che potete già ora utilizzare sono le seguenti:

NL_FindTagItem(), NL_GetTagData(), NL_NextTagItem();

EF_FindTagItem(), EF_GetTagData(), EF_NextTagItem();

che emulano tre funzioni della utility.library (altra nuova libreria 2.0).

Per l'utilizzo di queste valgono le stesse regole espone per le funzioni di emulazione di gadtools; inoltre occorre #includere nel proprio sorgente il file "nl_utility.h", fornito con GTE.lib.

In futuro queste tre funzioni saranno distribuite in una libreria di emulazione parziale o completa della utility.library.

1.9 Nota importante circa gadtools.library 34.1 e GTE.lib

8. NOTA IMPORTANTE CIRCA GADTOOLS.LIBRARY 34.1 E GTE.LIB

La funzione GetVisualInfoA() di gadtools.library 34.1, come anche le sue corrispondenti funzioni EF_ e NL_ di GTE.lib, e a differenza dell'omonima funzione della vera gadtools.library, accetta anche un NULL come primo argomento, cioè come puntatore allo schermo; in questo caso infatti

assume che si intenda lo schermo del Workbench.

Lo scopo di ciò è evitare di dovere rintracciare lo schermo del Workbench: infatti la funzione di intuition LockPubScreen(), che sotto 2.0 si usa proprio a questo scopo, non esiste sotto 1.3 (e non deve quindi essere chiamata!!!!).

Occorrerà pertanto usare qui del codice condizionale, poichè purtroppo la funzione GetVisualInfoA() della VERA gadtools.library V37 NON accetta un NULL come argomento.

Esempio:

```
if (SysBase->lib_Version >= 36) /* Se siamo sotto 2.0 o superiori */
{
screen = LockPubScreen(NULL);
if (!screen) Errore("Niente Workbench");
}
else /* Se siamo sotto 1.3 o inferiori */
{
screen = NULL;
}
visualinfo = GetVisualInfoA(screen,NULL);
... /* Resto del programma qui */
FreeVisualInfo(visualinfo);
if (screen) UnLockPubScreen(NULL,screen);
```

1.10 Differenze

9. DIFFERENZE

Ci sono naturalmente delle lievi differenze tra il funzionamento di GTE.lib (o gadtools.library 34.1) e quello della vera gadtools.library, e sono tutte elencate nel file "Differenze.doc". Si tratta comunque di dettagli minimi e di secondaria importanza (almeno secondo il mio parere).

1.11 Conclusione

10. CONCLUSIONE

Questa è la prima versione di gadtools.library 34.x e di GTE.lib resa pubblica, pertanto è possibilissimo che vengano riscontrati dei bug nel funzionamento. Se dovesse accadere, prego chiunque ne fosse vittima di segnalarmi il problema che sarà (se possibile) eliminato nelle prossime versioni.

Scrivete a:

Massimo Tantignone

via Campagnoli, 4

28100 Novara (NO)

ITALIA

Per concludere, vorrei segnalare che sto lavorando all'emulazione di altre librerie 2.x e distribuirò anche queste quando saranno ultimate.

Massimo Tantignone, 27 gennaio 1993
