

# **ClassAction**

Martin R. Gasmi

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> ClassAction		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Martin R. Gasmi	July 31, 2024	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>ClassAction</b>	<b>1</b>
1.1	ClassAction 4.0 Guide . . . . .	1
1.2	What is ClassAction? . . . . .	1
1.3	more about ClassAction . . . . .	2
1.4	System Requirements . . . . .	3
1.5	Configuration of ClassAction . . . . .	3
1.6	ClassAction Prefs . . . . .	8
1.7	Using the learn function . . . . .	8
1.8	What is a Class . . . . .	9
1.9	Creating a new Class . . . . .	10
1.10	Defire una nuova azione . . . . .	12
1.11	CLI mode . . . . .	13
1.12	WB mode . . . . .	14
1.13	NO CLI mode . . . . .	14
1.14	ARexx mode . . . . .	14
1.15	Arguments . . . . .	14
1.16	Commands . . . . .	15
1.17	Future Improvements . . . . .	17
1.18	About . . . . .	17
1.19	The Author . . . . .	18
1.20	Using ClassAction . . . . .	18
1.21	The Buttons . . . . .	19
1.22	The Menu . . . . .	21
1.23	Technicals Infos . . . . .	21
1.24	Installation . . . . .	22
1.25	Legal words . . . . .	22
1.26	History . . . . .	22
1.27	The Arexx Commands . . . . .	30
1.28	What should you do after install . . . . .	32
1.29	Message for Philippe THOMAS . . . . .	32
1.30	Known Bugs . . . . .	33
1.31	Rawkey codes . . . . .	33

---

# Chapter 1

## ClassAction

### 1.1 ClassAction 4.0 Guide

ClassAction  
Versione 4.0

Cos'è ClassAction	Leggi prima questo
Richieste di sistema	Ciò di cui hai bisogno
Installazione	Come installare ClassAction
Dopo l' Installazione	Creare classi e azioni
Usare ClassAction	Come usare questo strumento
I Bottoni	Cosa fanno e che significano
I Menu	Le voci dei menù
Configurazione	Cambiare i Parametri
Preferenze ClassAction	Configurazione classi e azioni
I comandi Arexx	La porta ARexx e i comandi
Aggiunte future	Cosa dovrebbe essere aggiunto nel futuro
Storia & Caratteristiche	Dall'inizio...
Licenza	
L'Autore	
Ringraziamenti	
Conoscere i Bugs	

Traduzione italiana a cura di Umberto SETOLA <umb.sirvi@libero.it>

Revisione di Arturo Franzin <arturfra@libero.it>

Copyright 2001 Tutti i diritti riservati

### 1.2 What is ClassAction?

ClassAction è un piccolo strumento che semplificherà la vita di tutti gli utenti di Hard Disk.

Chiunque possiede un Hard Disk, ha sempre un gruppo di file: eseguibili, moduli, immagini, sorgenti, animazioni, suoni...

ClassAction determina il tipo di un file selezionato e mostra una lista di azioni che può eseguire sul file stesso.

Per esempio, quando selezioni un'immagine GIF, essa sarà riconosciuta da ClassAction come un file di Classe GIF, verra' visualizzata una lista di azioni come 'Mostra' o 'Edita'.

ClassAction è anche un file manager completo con molte opzioni. Copia, sposta, rinomina, cancella, aggiungi icone, mostra informazioni, ... Semplicemente provalo e vedrai tutte le sue possibilità!

ClassAction è altamente configurabile:

Puoi aggiungere le tue classi e azioni.

Le azioni usano programmi esterni, così puoi usare il tuo visualizzatore preferito di immagini GIF per vedere i File GIF (o qualsiasi altra cosa). Specifica il drive le directory e le combinazioni dei tasti.

ClassAction ha un AppIcon, una porta ARexx ed è localizzato.

ClassAction usa xfdmaster.library per decomprimere file compressi. Con questa caratteristica, può essere rilevata anche una classe compressa.

Se impieghi un po di tempo a configurare ClassAction, dopo puoi usarlo per tutto!!! E' un modo molto semplice per manipolare file.

Prima di decidere di cancellarlo, provalo prima!!!!!!

Se vuoi saperne di più su ClassAction usalo ancora

## 1.3 more about ClassAction

ClassAction e ClassActionPrefs sono proprieta' © 1994-98 di Gasmi Salim e dal © 2001 da Martin R. Elsner.

Adesso ClassAction è MailWare, per cui spedisce una E-mail all'autore se usi ClassAction frequentemente.

I distributori PD possono includere ClassAction nella loro raccolta purchè mi informino che l'hanno fatto

Spero che troverai utile questo strumento.  
(almeno, io lo trovo utile:] )

Per informazioni tecniche di ClassAction usa Informazioni Tecniche

---

## 1.4 System Requirements

Requisiti:

- o Amiga OS 3.5 o superiore
- o un Hard Disk  
(ClassAction è abbastanza inutile senza un Hard Disk)

È tutto gente!!!

Raccomandato:

- o Maggiore velocità, come una scheda 68060 ...

## 1.5 Configuration of ClassAction

L'interfaccia e le caratteristiche di ClassAction sono configurabili attraverso i Parametri. ↩

Per cambiare un valore di un Parametro, devi semplicemente selezionare l'icona di ClassAction e scegliere la voce 'Informazione' nel menù 'Icône' del Workbench. ↩

Parte delle opzioni possono essere modificate mentre ClassAction è in esecuzione. Essi possono essere trovati nel menu Settaggi. ↩

Questa è la lista di tutti i Parametri:

=====

CAPREFS

Questo Parametro contiene il percorso di ClassActionPrefs.  
C'è bisogno per trovare ClassActionPrefs quando si sceglie la voce di menù preferenze.

Il valore di predefinito è: "SYS:prefs/ClassActionPrefs"

=====

CLISIZE

Rappresenta la dimensione della finestra di output della CLI, quando usa SystemTags(). ↩

La sintassi è: `DEVICE:TopX/TopY/Width/Height/Title`

Così puoi selezionare un altro dispositivo `CON:` per il CLI o definire una nuova posizione e dimensione della finestra.

ATTENZIONE:

Se assegni questo Parametro ad un valore strano, ClassAction non lancerà la CLI. ↵

Non modificarlo a meno che tu non sappia bene quello che stai facendo.

\*\* Mai \*\* aggiungere comandi dispositivo come `AUTO`, `CLOSE`, `WAIT` a questa stringa, ClassAction lo farà per te.

\*\* Mai \*\* mettere spazi nel titolo.

Nella versione 2.0, fu definito un Parametro simile adesso non è più voluto e nè più usato.

Il valore predefinito è:

`CON:0/0/640/100/ClassAction_Output_Window`

=====

#### CONFIRMREPLACE

Se clicchi su 'Copia' o un tasto attinente, esiste una probabilità che un vecchio file venga sostituito da quello nuovo. Se è assegnato `CONFIRMREPLACE`, in un caso del genere ti verrà richiesto se vuoi davvero cancellare il vecchio file.

=====

#### COPYICONS

Attivando questo Parametro, la 'copia' e i tasti attinenti copieranno anche l'icona quando copi un file. Questo è utile se `DISPINFO` non è assegnato.

=====

#### CREATEICONS

Il tasto 'Crea una nuova directory' crea una nuova directory. Se è settato ↵  
`CREATEICONS`,  
crea anche un'icona per questa directory.

=====

#### DECRUNCH

Se tu setti questo Parametro, ClassAction proverà a decomprimere file compressi usando la libreria `xfdmaster`.

C'è ne bisogno se vuoi riconoscere i file compressi.

---

Naturalmente se fai così, il file sarà pre-caricato in memoria e sarà allocato un buffer ai dati decompressi. Così, consumerà memoria (circa 2.5 x la dimensione del file). Così, se non hai molta memoria conviene disabilitare questa caratteristica.

---

#### DISPINFO

Se definisci questo Parametro, i files .info saranno mostrati nella lista.

---

#### DONOTWAIT

Non rimuovere questo Parametro, esso è indispensabile se ClassAction è caricato dal WBStartup.

---

#### DRAWERSFIRST

Usa questo Parametro per aggiustare il modo con cui i files sono ordinati. Se attivo, le directory sono posizionate in cima alla lista.

---

#### HELPPFILE

Se salvi questa documentazione in un'altra directory diversa da HELP:, allora tu devi indicare a ClassAction dove può essere trovata. Esempio: HELPPFILE=sys:docs/classaction.guide

---

#### ICONFILE

Questo Parametro è il file AppIcon. Così puoi selezionare il tuo AppIcon per ClassAction. Se assegni questo Parametro ad un file icona, ClassAction la userà come l'immagine AppIcon.

!Nota: Non fornire l'estensione .info per il nome del file.

Esempio: ICONFILE=Sys:icons/head  
userà l' icona.info dalla directiry Sys:Icane/  
come AppIcon.

Se ClassAction non riesce a caricare il file Icona, allora ne userà uno predefinito (ClassAction.info).

---

#### ICONSTART

---



Se definisci questo Parametro, ClassAction si avvierà come un AppIcon. Usalo per esempio se avvii ClassAction da WBStartup.

---

#### SHOWHELP

Se non sai quale scopo è per quell'icona, attiva questa opzione e verrà visualizzato un breve cenno in basso quando il mouse passa sopra ogni tasto.

---

#### X,Y

Specifica la distanza della finestra ClassAction dal margine sinistro e superiore dello schermo.

Se non è dato nessun X e nessun Y, la finestra è centrata sullo schermo.

#### HEIGHT, WIDTH

Setta l'altezza e la larghezza della finestra.

Se non è settato, la dimensione è aggiustata dai gadget della finestra.

#### POPUPHEIGHT, POPUPWIDTH

Puoi anche settare la dimensione della finestra popup.

---

Le seguenti azioni possono essere devolute ai tasti. Puoi specificare il tasto con il suo valore ascii (per esempio Key\_TOGGLE=":") o il suo codice (per esempio KEY\_ALL=66). Fai attenzione in quanto il Tasto HELP e i normali caratteri ('a'-'z','A'-'Z') sono già usati!

Se non li specifichi, sono usati i tasti in parentesi.

Io ho aggiunto una lista spesso desiderata rawkey codes .

KEY\_ALL:           seleziona tutti i file ("+")

---

Key\_DEVICES1: mostra la lista dei device nella lista sorgente (81,F2)

---

KEY\_DEVICES2:mostra la lista dei device nella lista destinazione (88,F9)

---

Key\_MKDIR1: crea una directory nel sorgente del browser (82,F3)

---

Key\_MKDIR2: crea una directory nella destinazione del browser (89,F10)

---

```
=====
```

```
KEY_NNE: deseleziona tutti i file ("-")
```

```
=====
```

```
KEY_PARENT1: va nella directory principale nella lista sorgente (65, BACKSPACE)
```

```
=====
```

```
KEY_PARENT2: va nella directory principale nella lista destinazione (87, F8)
```

```
=====
```

```
KEY_PATTERN: chiama il requester del filtro ("$\times$")
```

```
=====
```

```
KEY_SELECT:      auto-selezione file dello stesso tipo
                  come la selezione corrente ("/")
```

```
=====
```

```
KEY_SWAP:        scambia i contenuti della lista  directory (66, TAB)
```

```
=====
```

```
KEY_TOGGLE:      commuta i file selezionati (67, ENTER)
```

```
=====
```

```
DRIVE1 to DRIVE13
```

Assegna questi Parametri ad un percorso valido, così essi saranno mostrati nei pulsanti del Requester.  
È utile per andare velocemente in una directory.

La sintassi è:

```
DRIVEx=<Button Text>,<Percorso>
```

cioè Drive9=Jpeg,dh0:gfx/pictures/jpeg

In questo modo il pulsante #9 sarà 'Jpeg', ma il percorso è assegnato a dh0:gfx/pictures/jpeg.

Puoi anche fornire solo il <Tasto Testo> senza il <Percorso>  
cioè Drive3=dh0:libs

In questo caso, il pulsante testo sarà assegnato allo stesso percorso del ↔ testo.

Ma non usare testi troppo lunghi perchè la finestra non è molto grossa ...

Da default non è mostrato nessun tasto.

## 1.6 ClassAction Prefs

Definire nuove classi e azioni è il cuore del programma.  
Per fare questo, devi usare il programma ClassActionPrefs.

Usare ClassActionPrefs dovrebbe essere facile... così andiamo!!

La finestra è divisa in due parti: Classi & Azioni.

Prima selezionare una classe, le azioni associate saranno mostrate sulla parte delle azioni.

Per aggiungere una classe o cancellare una classe, devi semplicemente cliccare ←  
sui  
tasti associati.

E lo stesso concetto vale per le azioni...

1. Che cos' è una classe
2. La creazione di una nuova classe
3. La definizione di una nuova azione
4. La funzione apprendi

## 1.7 Using the learn function

La funzione apprendi è fornita per aiutarti a definire nuove classi.

Quando definisci una nuova classe, potresti dover definirgli lo scostamento, può essere lungo e scoccante editare file per vedere quali scostamenti potevano definire questa nuova classe.

Ecco perchè è stata fatta la meravigliosa funzione apprendi !

Per usarla, esegui semplicemente questa procedura:

- 1: Definire la tua classe come normale, riempire Nome e Nome classe.
  - 2: Cliccare sul pulsante 'Apprendi'.
  - 3: Selezionare con il requester fornito tanti file quanti ne vuoi chè appartengono alla classe che vuoi definire.  
Più file selezioni, migliore sarà il risultato.  
ClassAction proverà a scoprire la definizione di scostamento.  
(per selezionare file multipli, usare il tasto SHIFT.)
-

- 4: Dopo l'analisi, troverai una finestra di scostamento.
- 5: Modificali a mano se desideri.
- 6: Clicca su 'Accetta' per usare gli scostamenti per la tua classe.  
o su 'Cancel' per cancellare.

Avvertimento:

Se vuoi imparare la funzione lavora allora devi assicurarti che:

- \* TUTTI I FILE SCELTI APPARTENGANO ALLA STESSA CLASSE.
- \* TUTTI I FILE SCELTI NON SIANO COMPRESSI.

È tutto!!! Facile no?.

## 1.8 What is a Class

Una classe è un famiglia di file. Per esempio i file C possono essere considerati come una Classe, chiamiamoli Classe C.

Con ClassActionPrefs puoi definire tante classi quante ne vuoi, purchè tu spieghi come riconoscerle.

Per spiegare come riconoscere una Classe, ci sono due metodi:  
il nome del filtro (coincide con) e i contenuti dei file.

L'attributo Nome filtro è usato per riconoscere un file dal nome.

Gli attributi di scostamento sono usati per riconoscere un file dai contenuti.

Ci sono tre Classi inglobate che non puoi rimuovere, essi sono mostrati in bianco nella ListView delle Classi e non può essere rinominato.

Il primo è chiamato "Classe sconosciuta"  
Questa classe contiene tutti i file che ClassAction non può riconoscere.

Il secondo è chiamato "Azioni Generiche"  
questa classe contiene Azioni che saranno mostrate in Tutte le altre classi...

Interessante: se vuoi avere un'azione 'Copia' per tutte le classi, puoi crearlo per ogni classe definita, ma è una cosa molto lunga e scoccante.

Il modo migliore per fare ciò è creare questa azione nella classe "Azione Generica", così 'Copia' verrà mostrato per tutte le classi.

Le azioni generiche sono mostrate in bianco nella lista ClassAction delle azioni e  
sono visibili solo nella finestra ClassAction.

Essi non verranno mostrati quando ClassAction è un AppIcon,  
o attraverso comando ARexx Carica.

Il terzo è chiamato "Directory" e contiene azioni disponibili  
quando selezioni una directory.

## 1.9 Creating a new Class

Clicca semplicemente sul pulsante 'Aggiungi' sulla parte aggiungi una classe.

Una classe ha 3 proprietà:

- un Nome classe
- un Nome filtro
- uno Scostamento

o Il Nome è semplicemente il nome della classe, dipende da te sceglierla.  
AVVERTIMENTO: un nome classe deve essere unico.

o Il Nome filtro è qualsiasi espressione regolare AmigaDos, come:

`#?.c, mod.#?, #?.c|#?.h, #?b[a|c], #?toto? .....`

(leggi il manuale AmigaDOS per tutti i caratteri jolly)  
AVVERTIMENTO: Non usare la wildcard \* usa invece #?

Il Nome filtro non è dipendente dalle maiuscole, così toto.C coincide con #?.c

Se definisci una classe usando un nome filtro, devi essere sicuro che  
la definizione è sempre corretta.

Esempio: se definisci la classe GIF con il nome filtro #?.GIF  
tutti i file .GIF saranno riconosciuti come file GIF.

Ma sei sicuro che Tutti i .GIF file sono GIF, o  
che tutto i tuoi file GIF hanno l'estensione .GIF ?

Così, dovresti usare un nome filtro solo in due situazioni:

- il nome filtro è un bijection della classe  
(ex: #?.info è un buon nome filtro per classe di Icona)
- tu non hai la scelta  
(ex: come riconoscere un sorgente C, eccetto con #?.c)

o Altrimenti, dovresti usare Scostamento.

Uno scostamento è un posto dentro un file dove dovremmo trovare qualcosa

---

da riconoscerlo.

Per esempio, le immagini GIF iniziano sempre con la stringa 'GIF' allo scostamento 0. ↵

Ci sono tre sintassi per definire lo scostamento:

=====

Sintassi #1 : Scostamento,Stringa esadecimale

Scostamento è un numero Decimale che tiene lo scostamento.

Stringa esadecimale è una stringa esadecimale (HEX) che dovrebbe essere trovata a quello scostamento. ↵

Esempio 1: 0,4f4a significa che il file deve incominciare con byte \$4f e \$4a alla posizione 0

Esempio 2: 9,448b3c significa che al byte #9, dovremmo trovare \$44 \$8b \$3c

=====

Sintassi #2 : Scostamento,'Stringa'

Scostamento è un numero Decimale che tiene lo scostamento.

Stringa è una stringa ASCII che dovrebbe essere trovata a quello scostamento. ↵  
.

Esempio 1: 0,'GIF' significa che il file deve iniziare con la stringa 'GIF'

Esempio 2: 9,'FuBar' significa che al byte #9, dovremmo trovare la stringa 'FuBar'

=====

Sintassi #3 : Scostamento,"Stringa"

Scostamento è un numero Decimale che tiene lo scostamento.

Stringa è una stringa ASCII che dovrebbe essere trovata a quello scostamento. ↵  
.

Notare la differenza con la precedente sintassi, qui noi usiamo " definire la stringa, e l'abbiamo usata nella sintassi #2.

È lo stesso concetto della Sintassi #2 ma qui, la stringa comparata non è Dipendente dalle maiuscole.

Per esempio: un file AmigaGuide inizia sempre con la stringa: @database in lower or upper case.

Se usi il metodo #2 per riconoscere un file amigaguide, con 0,'@database', ClassAction non riconoscerà un file che inizia

con @DATABASE come un file AmigaGuide.

Funziona se lo scostamento è definito con la sintassi #3: 0,"@database"

=====

Puoi definire fino a 5 Scostamenti per definire una classe.  
Un file è riconosciuto come una classe se tutti gli Scostamenti coincidono.  
Se il file è adatto a più di una classe, esso sarà riconosciuto come la classe con più scostamenti.

Ex: se la classe X è definito come:

Scostamento #1: 0,4a8b6c  
Scostamento #2: 58,14

Tutti i file che iniziano con 4a8b6c e avendo \$14 al byte #58 saranno dichiarati come X.

Per definire molti Scostamenti devi semplicemente cliccare sul gadget ←  
ciclico 'Scostamento #'  
per attivare il prossimo scostamento.

Remark: La Classe ASCII

C'è un comando di Scostamento interno chiamato: ASCII[]

Se metti questo Comando nello Scostamento #1 (i.e Scostamento#1=ASCII[]),  
esso coinciderà con con il file ASCII.  
Ma ClassAction proverà questo dopo che ogni alta casa fallisce.  
Grazie a questo, i file Amigaguide (che sono ascii) non saranno riconosciuti come ascii se hai già definito una Classe Amigaguide.

Normalmente non dovresti usarlo in quanto ne ho fornito uno standard  
nel file Prefs dove la Classe 'ASCII' è già definito a usare questo comando.

## 1.10 Defire una nuova azione

Una volta definita la classe dovresti definirne le azioni.  
Ogni Classe può avere tante Azioni quante ne vuoi.

Clicca semplicemente sul tasto 'Aggiungi' sulla sezione relativa all'aggiunta di un'azione.

Un'azione ha 6 proprietà:

- un Nome
- un Comando
- una dimensione dello Stack (solo se è eseguito da Cli)
- un Ritardo (solo se è eseguito da Cli)

- un comando exec
- una flag Riesamina Directory

o Nome è il nome dell'azione.

o Modo può essere: Cli , WB , No Cli

o ARexx .

o Comando è una linea di comando valida AmigDOS e può contenere parametri.

Dovresti usare sempre il percorso completo per gli eseguibili.

Esempio: usa C:Copy invece di Copy in exec line.

Puoi inserire nelle linee exec, Arguments e Commands all'interno di ClassAction.

o Vista 'Riesamina Directory' se vuoi che questa azione riesamini la directory corrente, per azioni che cambiano il contenuto delle directory come cancellare o rinominare un file.

Tasti

o I tasti freccia Su e giù permettono di ordinare le azioni.

o Il tasto 'Carica' chiede di selezionare un eseguibile nella linea exec.

o I 'Comandi' tasto salta su un requester che elenca tutti gli argomenti possibili e i comandi per una linea exec.

## 1.11 CLI mode

\*\*\*\* Modo CLI \*\*\*\*

Se viene scelto 'Cli' , allora nel momento in cui sta selezionando l'azione, l'azione verrà lanciata da una cli e la dimensione dello stack del cli sarà determinata dal valore di stack (default è 4096).

L'avvio del modo Cli aprirà una Cli solo se c'è bisogno (se l'eseguibile visualizza qualcosa).

Puoi definire la proprietà del ritardo per la CLI:

Se il Ritardo è negativo (cioè Ritardo = -1), la Cli aspetterà finché la chiudi a mano con il gadget di chiusura in alto a sinistra della finestra.

Se il Ritardo è zero (cioè Ritardo = 0), la Cli si chiuderà non appena il task è terminato.

Se il Ritardo è positivo (cioè Ritardo = n con n>0), la Cli aspetterà n secondi prima di chiudersi, ma puoi forzarlo



cliccando il gadget di chiusura.

La dimensione della Cli utilizzata può essere trovata nel Parametro CLISIZE.

## 1.12 WB mode

\*\*\* modo WB \*\*\*

Se viene scelto 'WB', allora non sarà aperto nessun cli, e ClassAction simulerà l'azione di lanciarlo da Workbench.

La prima stringa sarà considerata un'icona, e i prossimi argomenti vengono utilizzati come se fossero stati lasciati sull'icona. Inserendo '[f]' come comando, il nome del file verrà eseguito come se si fosse cliccato due volte su esso.

AVVERTIMENTO: questo modo è valido solo con file che hanno icone, e dovrebbero essere usati come argomenti solo nomi di file (nomi completi).

## 1.13 NO CLI mode

\*\*\* modo senza CLI \*\*\*

Se viene scelto 'No Cli', non sarà aperto nessun cli anche se il programma visualizza qualcosa, ma il task sta ancora girando da un CLI.

## 1.14 ARexx mode

\*\*\* modo ARexx \*\*\*

Se viene scelto 'ARexx', allora sarà lanciato rx con il dato comando exec. Il comando exec deve essere uno script ARexx.

Naturalmente REXXMaster deve essere attivo e Rx nella directory Sys:rexxc/ per funzionare.

## 1.15 Arguments

Attualmente, sono possibili 8 comandi di argomenti:

i primi 4 comandi sono in minuscolo: [f] [s] [b] [x]  
e includono il risultato tra virgolette

[f]: percorso completo del file selezionato con virgolette  
[s]: percorso completo del file selezionato senza suffisso con virgolette  
[b]: nomefile del file selezionato con virgolette  
[x]: nomefile del file selezionato senza suffisso con virgolette

---

gli ultimi 4 comandi fanno la stessa cosa di quello precedente ma senza virgolette

```
[F]: percorso completo del file selezionato
[S]: percorso completo del file selezionato senza suffisso
[B]: nomefile del file selezionato
[X]: nomefile del file selezionato senza suffisso
```

Esempio: lasciando say selezioni il file ram:env/sys.prefs

```
[f] = "ram:env/sys.prefs"
[s] = "ram:env/sys"
[b] = "sys.prefs"
[x] = "sys"

[F] = ram:env/sys.prefs
[S] = ram:env/sys
[B] = sys.prefs
[X] = sys
```

Esempio: lasciando imagine hai selezionato il file ram:main.c

```
* La linea exec
          c:copy [f] [F].bak
sarà sostituita da:
          c:copy "ram:main.c" ram:main.c.bak

* La linea exec
          c:copy [f] [S].bak
sarà sostituita da:
          c:copy "ram:main.c" ram:main.bak
```

## 1.16 Commands

Attualmente, sono possibili cinque comandi di richiesta:

```
REQD[testo]: Richiesta per una Directory.
REQF[testo]: Richiesta per un File.
REQV[testo]: Richiesta per un Volume.
REQT[testo]: Richiesta per un testo.
SURE[testo]: Chiede all'utente per confermare.
```

=====

Comandi REQ

---

I comandi REQ aprono un requester con il titolo [testo].  
Questo è utile quando si ha bisogno di linee di comando interattive.

Esempio:

```
bin:lha x [f] to REQD[Choose a Directory to unarchive]
```

Questo aprirà un requester di directory lasciando all'utente la scelta della directory di destinazione, e il file selezionato [f] sarà ←  
disarchiviato  
nella directory selezionata.

REQF[] è lo stesso eccetto che chiede un file.

Esempio:

```
c:dir [f] > REQF[Choose a file]
```

REQV[] chiede all'utente di scegliere un volume.

REQT[] chiede un testo, ed è utile per chiedere argomenti per esempio:

```
c:cpu REQT[Enter arguments for CPU]
```

=====

Comando SURE:

Il comando SURE[testo]aprirà un requester con testo [testo]  
e con 2 pulsanti: si / no.  
Se l'utente sceglie no, la linea exec è termina.  
Se l'utente sceglie si, ClassAction eseguirà la linea exec  
sulla parte giusta del comando Sure.

Esempio:

```
SURE[Davvero cancello questo file?]C:delete [f]
```

Questo aprirà un requester chiedendo all'utente di rispondere si o no  
alla domanda "Vuoi veramente cancellare questo file?".  
Se l'utente risponde no, non succede nulla; se, invece risponde si,  
allora viene eseguito C:delete [f].

=====

Naturalmente puoi combinare qualsiasi numero di argomenti / comandi in una  
linea exec.

Esempio:

```
SURE[Vuoi veramente rinominare questo file]c:rename [f] REQF[Dai un nome ←  
nuovo ]
```

## 1.17 Future Improvements

Nessuno per il momento ....

## 1.18 About

Mi piacerebbe ringraziare le seguenti persone per il loro contributo al rinascimento di ClassAction:

- o tutti i beta-tester che hanno trovato difetti o avevano grandi proposte:
  - Cristiano Aichinger
  - Jean-Pierre Riviere
  - Jimmy Sulter
  - Jocke Sjöblom
  - Patric Hofmann
  - Rolf Max Rotvel
  - Sinan Gurkan
- o Mick Saile per il consenso ad usare le sue GlowIcons (opusgibar, disponibile in Aminet)
- o Dietmar Eilert per il grande GoldED che ho usato per la programmazione
- o e Salim Gasmi per lo sviluppo dalla versione 1.0 alla 3.6 di questa ottima parte di software!

Martin

Mi piacerebbe ringraziare le seguenti persone :

- o Philippe Thomas (per i suggerimenti, gli aiuti, tests, La guida Francese e UTT usata per l'install)  
  
Hey Phil, se leggi questo, clicca [qui](#)
  - o Bruno Durremberger (come Beta tester)
  - o Jean Michel Dessolas (come beta tester su un A4000/40)
  - o Joern Krueger come beta tester e per le buone idee.
  - o Ovvie Implementazioni Corp (per Dice C Pro)
  - o Nico Francois per la libreria ReqTools
  - o Georg Hörmann per la \*Grande\* libreria xfdmaster
  - o Stefan Stuntz per la meravigliosa interfaccia utente MUI.
-

- o Tutti gli utenti hanno che mi hanno contattato per segnalare difetti o per suggerimenti.

Salim

## 1.19 The Author

Puoi contattarmi al seguente indirizzo :

Martin R. Elsner  
Müsener Straße 46

57223 Kreuztal  
Germania

Web: <http://www.martin-elsner.de>

E-Mail: [classaction@martin-elsner.de](mailto:classaction@martin-elsner.de)

- o il precedente sviluppatore Salim Gasmi :

Gasmi Salim  
4b rue des petits champs

67300 Shiltigheim  
Francia

Irc: Dr\_Unix (#amigafr, #amiga)

Web: <http://www.gasmi.rete>

E-Mail: [salim@gasmi.net](mailto:salim@gasmi.net) oppure [salim@sdv.fr](mailto:salim@sdv.fr)

## 1.20 Using ClassAction

Usare ClassAction è molto semplice. Se tu non sai un'icona che scopo ha, devi semplicemente muovere il mouse su di essa e leggere il testo in basso (devi aver attivato l'opzione Mostra aiuto).

I pulsanti sulla sinistra sono relativi ai dati nei Parametri. Cliccandoci sopra ne mostreranno il percorso nella lista Sorgente.

Questa lista Sorgente puo' essere mossa come in un normale file-requester. Sono disponibili azioni speciali attraverso le icone in alto e in basso . Alcuni di questi metodi possono essere visitati da tastiera, vedi parametri .

Dopo aver selezionato un file nella Sorgente, vedrai nella lista visibile a destra, la classe dei file e le corrispondenti azioni ↔ definite.

---

Devi semplicemente selezionare l'azione che vuoi...

Premendo un tasto verra' mostrato il primo file che inizia con questa lettera, mentre Shift+Tasto fara' lo stesso per le directory.

Se clicchi due volte su un file, sarà lanciata la prima azione definita.

La finestra ClassAction è un AppWindow, così puoi lanciare le icone sulla finestra principale.

Aprire il programma di preferenze selezionando 'Preferenze' dal menu . Per uscire, devi semplicemente cliccare sull'icona piccola o selezionare 'Esci' dal menu .

Per trasformare la finestra in un AppIcon devi semplicemente scegliere l'icona iconify o scegliere 'iconify' dal menu .  
Clicca due volte sull' AppIcon per far apparire la finestra.

Quando ClassAction è AppIconified, puoi lanciarci sopra le icone così il tipo di file sarà riconosciuto.  
Se la classe ha solo un'azione definita, essa sarà eseguito.  
Altrimenti, si apre una finestra con la lista di tutte le azioni possibili, ←  
dove  
devi semplicemente scegliere una.

Usare il file manager è davvero semplice, la lista a sinistra è sempre la directory sorgente mentre quello sul gruppo "Manager" è sempre la directory di destinazione.

Con i bottoni sotto la lista destinazione puoi muovere in questa lista e trasferire file dalla sorgente alla destinazione.

Vedere anche

Comandi ARexx  
Bottoni  
Configurazione  
Menu

## 1.21 The Buttons

Tasti Selezione

Sopra alla lista sorgente ci sono 6 pulsanti che permettono selezioni più facili:

Seleziona tutto: nella lista, anche le directory.	Seleziona tutte le entrate ←
Deseleziona tutti i files: entrate.	Deseleziona tutte le ←
Seleziona i files adattando un filtro: coincidono con il filtro che hai inserito.	Seleziona tutti i file che ←

	Le directory non saranno ← selezionate.
Seleziona tutti i files con la stessa classe: della stessa classe di quello selezionato.	Seleziona tutti i file ←
	La directory non sara' ← selezionata.
Commuta selezione: deseleziona.	Switch seleziona/ ←
	Interessa anche directory.
Scambia i contenuti della lista: sorgente e della destinazione nella lista.	Scambia i contenuti del ←

### Tasti Directory

Per entrambe le liste ci sono tasti per muoversi dentro e creare nuovi cassette:

Vai alla directory madre:	Va alla directory principale. Se sei nella directory di base, viene visualizzata ← la lista dei dispositivi.
Mostra tutti i dispositivi:	Mostra volumi~e le assegnazioni.
Crea una nuova directory:	Creare una nuova directory; verrà richiesto il ← nome.  Se l'opzione 'Crea Icone' è attiva, sara' creata anche un'icona.

### Tasti Azione

Alcune azioni possono essere eseguite sui file selezionati direttamente da questi Pulsanti:

WbInfo:	Mostra le informazione delle icone Workbench dei file ← selezionati.  Lì si possono modificare i parametri, bit di protezione, ← ...
Aggiungi le icone:	Crea un'icona per i file selezionati. Se esiste già un'icona sarai avvisato, e non succederà nulla.
Rinomina:	Rinomina i file selezionati.
Elimina:	Cancella i file selezionati. Ti verrà richiesta una conferma per cio'.

### Tasti Trasferimento

Puoi trasferire file con l'aiuto dei seguenti pulsanti:

Copia:	Copia i file selezionati nella directory di destinazione.
Copia e rinomina:	Copia i file selezionati nella directory di destinazione, e da ad essi il nome che viene inserito.
Sposta:	Muove i file selezionati dalla directory di destinazione.
Sposta e rinomina:	Muove i file selezionati dalla directory di destinazione, e da ad essi il nome che viene inserito.

Se i file che verranno sostituiti hanno l'opzione 'Confermi Sostituzione' ←  
attivata, ti verrà

richiesta una conferma.

Alcuni di questi metodi possono essere visitati da tastiera, vedere parametri ↵  
.

## 1.22 The Menu

Progetto

Aiuto: Mostra questo manuale.  
Come alternativa puoi anche premere il tasto 'Aiuto'.  
Preferenze: Apre ClassActionPrefs per modificare le classi e le azioni.  
Per cambiare settaggi del programma per ClassAction, usa il menù 'settaggi' o cambiare i Parametri.  
Autore: Mostra informazioni sul programma.  
Iconifica: Chiude la finestra e crea un appicon.  
Esci: Esce dal programma.

Settaggi

Fissa Finestra: Salva la posizione della finestra corrente e le ↵  
dimensioni.  
Al prossimo avvio di ClassAction userà queste dimensioni.  
Vengono memorizzati come Parametri.

Le seguenti opzioni sono identiche ai parametri con lo stesso nome.

Così se vuoi cambiare una delle opzioni permanentemente, devi cambiare il Parametro.

## 1.23 Technicals Infos

ClassAction in un primo momento fu codificato con Dice C 3.0 mentre adesso ↵  
viene  
sviluppato in C++ con lo StormC 3.0, usando ReAction (bel nome;-).

Informazioni varie:

Il file prefs è un file ASCII chiamato: ENVARC:ClassAction/ClassAction.prefs  
Le azioni generiche sono salvate in ENVARC:ClassAction/ClassAction\_Gen.prefs  
Le azioni delle Directory sono salvate in ENVARC:ClassAction/ClassAction\_Dir. ↵  
prefs

Le librerie DISK sono usate se trovato:

xfdmaster.library V30+

Come fa ClassAction a determinare una classe:



- 1- controlla se il nome del file coincide con i Nomi filtri delle classi ←  
definite.
  - 2- controlla se il file coincide con lo scostamento delle classi definite ←  
.
  - 3- decompri il file usando xfdmaster.library.
  - 4- controlla se il buffer scompattato coincide con lo scostamento delle ←  
classi definite.
  - 5- controlla se il file è ASCII
- Se tutto questo fallisce, allora il file viene identificato come 'Classe ←  
Ignota'.

## 1.24 Installation

Per Installare:

Usare l'installer script dato con l'archivio.  
Per usarlo, devi semplicemente cliccare sull'icona install.

## 1.25 Legal words

Copyright

ClassAction e ClassActionPrefs sono Copyright © di Gasmi dal 1994-2000  
e di Martin R. Elsner nel © 2001

ClassAction è un programma mailware. La confezione non può essere modificata  
in nessun modo e non può essere usata per scopi commerciali senza il  
precedente consenso scritto dell'autore. Il messaggio di copyright dovrebbe  
essere preservato.

Garanzia

Non sarà accettata nessuna responsabilita' per qualsiasi danno che  
può essere risultato dall'uso di questo programma. Tutto l'uso è a tuo  
rischio. Il software è fornito "così come è" senza garanzia  
implicita. La documentazione dovrebbe essere corretta ma l'autore si  
riserva il diritto di aggiornare il software e/o la documentazione  
senza preavviso.

## 1.26 History

Storia di ClassAction V 4.0 © Martin R. Elsner & Salim Gasmi

31/03/01: V4.0

---

- "Riscritto" il programma per usare ReAction e le nuove caratteristiche dell'OS3 ←  
.5+ .
- Adesso possono essere configurati solo 13 tasti(dovrebbero essere abbastanza!).
- Le azioni Manager sono ora delle immagini!
- I device adesso sono mostrati nella normale lista dei file.
- Nuovi pulsanti: MakeDir anche per lista di file di destra, WbInfo, AddIcon, MoveAs, tutti i metodi di selezione.
- Aggiunto un menu(!) con aiuto, preferenze, settaggi, ...
- Aggiunto aiuto asincrono!
- Aggiunta una barra di progressione per copiare/spostare/...
- Ora la classe di un file è quella con i maggiori aggiustamenti di offset, non il primo che si accorda.
- ottimizzazione del commento e rimozione del Parametro NOCOMMENT.
- Nuovi Parametri: COPYICONS,CREATEICONS,DRAWERSFIRST,HELPPFILE,SHOWHELP X,Y,WIDTH,HEIGHT,POPUPWIDTH,POPUPHEIGHT,piu' tasti
- Adesso la porta ARexx e'CLASSACTION.1 per avere una trattazione più facile

28/03/98: V3.6

- Aggiunto un nuovo eseguibile chiamato CStart che può essere assegnato come progetto per le icone.
  - Adesso ClassAction assegna il filetype nei commenti dei file per velocizzarne riconoscimento.
  - Aggiunto il Parametro NOCOMMENT per disabilitare i commenti di file.
  - La DirList principale di ClassAction adesso è usabile da tastiera  
Con i tasti su/giù/spazio/Enter"
  - Aggiunti i tasti filestart, se premi su un tasto, viene selezionato il primo file che inizia con questa lettera (Shift+Tasto per le Directory)
  - Aggiunto un tasto per permettere la selezione dei pattern (configurabile ←  
attraverso  
i Parametri)
  - Aggiunto un tasto per selezionare tutti i File (configurabile attraverso  
i Parametri)
  - Aggiunto un tasto per deselezionare tutti i file (configurabile attraverso i ←  
Parametri)
  - Aggiunto un tasto per commutare i file (configurabile attraverso i Parametri)
-

- Aggiunto un tasto per la directory principale (configurabile attraverso i Parametri) ←
- Aggiunto un tasto per selezionare i file (configurabile attraverso i Parametri)
- Eliminato il difetto di MoveDir
- Eliminato il difetto del requester di Makedir
- Ottimizzate alcune routine diventate così più veloci.
- Fissati i comandi Crash/Hang/Hit in ClassActionPrefs
- Fissati alcuni difetti minori

27/06/97: V3.5

- Versione di Premio.
- Il registro CA non è più indispensabile, la registrazione avviene on-line sul mio sito <http://www.gasmi.net>
- Il file manager maneggia anche le icone
- Aggiunto il Parametro DISPINFO
- Modificato il Parametro GTLIKE
- Rimossi difetti minori

07/02/97: V3.4

- Questo fu ritenuto l'Ultimo rilascio di ClassAction
- Corretti alcuni difetti minori
- Aggiunto il Parametro ICONSTART per forzare ClassAction ad avviarsi iconificato per persone che non sanno come farlo usando le Prefs di MUI

23/08/96: V3.3

- ClassAction non è più ShareWare ma MailWare!!!!
- Incluso il Registro CA per la registrazione libera
- Rimossi alcuni difetti.

07/07/96: V3.2

- ClassAction è stato ricompilato con ultimo rilascio di Dice ed è più stabile.
  - il motore ARexx è adesso completo.
-

- Rimossi alcuni difetti.

09/04/96: V3.1

- ClassAction ha adesso un piccolo file manager all'interno.
- Rimossi alcuni difetti.

12/03/96: V3.0 (Maggiore Aggiornamento)

- ClassAction e ClassActionPrefs adesso usano MUI 3.0+
- ClassAction può avere fino a 50 tasti di percorso
- ClassAction riconosce directory.
- ClassAction utilizza la selezione multipla.
- ClassAction ha un appwindow.
- Adesso puoi assegnare l'Esattezza di Apprendimento.
- Aggiunti i Parametri CAPREFS e GTLIKE.
- I Parametri, HEIGHT, APPSTART, WINX, WINY, ICONX, ICONY, CX\_PRIORITY, CX\_HOTKEY, REQBUG, STARTDIR, WBFONT, PUBSCREEN sono stati rimossi.

25/09/95: V2.8

- ClassAction ha adesso un pieno supporto per le commodity e un HotKey per visualizzare/nascondere ClassAction.
- Il Parametro APPSTART può essere assegnato per nascondere (APPSTART=Nascondere) se vuoi che ClassAction parta nascosto.
- Aggiunto il ToolType CX\_HOTKEY
- Aggiunto il ToolType PUBSCREEN per permettere di usare Schermi Pubblici
- Il selezionatore di file interno di ClassAction non stava liberando tutta la memoria allocata, adesso non piu'.

11/09/95: V2.75

- Il comandi REQD, REQF, REQV in realta' non erano incompatibili con il modo appicon, ma lo erano con alcuni programmi come MagicMenu (Sfortuna che io uso MagicMenu...)

Molti utenti si sono lamentati circa lo swith automatico swith in un comando REQ\_T anche se non usano un programma incompatibile.

Io ho aggiunto un ToolType (REQBUG) per lasciare all'utente la scelta di

---

auto-switch nel REQT.

03/09/95: V2.7

- Comandi aggiunti [b],[x],[B],[X],[F],[S]
- Aggiunto l'Azione comando ARexx.
- rimossi alcuni difetti nelle routine di rendering
- rimosso un difetto con le icone multiple lanciate sull'AppIcon.
- I comandi REQD,REQF,REQV adesso sono scambiati in REQT nel momento in cui li stai usando da AppIcon o ARexx, essi sono solo compatibili con il modo finestra.

28/08/95: V2.6

- La finestra ClassAction finestra adesso è ridimensionabile.
- ClassAction e ClassActionPrefs adesso usano la Reqtools.library.
- Aggiunto il comando REQV per richiedere un volume.
- Aggiunto il comando REQT per richiedere un testo.
- Aggiunti i ToolType WINX e WINY.
- Il requester imparare adesso ha un pulsante tutti.
- i file selezionati hanno tutti i requester REQs.
- ClassAction adesso usa un file di Key per le versioni registrate, questo file di key è messo in S:.

17/07/95: V2.5

- ClassAction e ClassActionPrefs adesso sono localised. viene fornito con l'archivio un catalogo Francese.
  - Aggiunta la funzione impara a ClassActionPrefs.
  - Per selezionare la prima azione di un file, devi Cliccarci sopra due volte invece di riselezionare il file.
  - Le azioni generiche sono adesso sincrone e riesamina la directory corrente.
  - Il ToolType AUTOSELECT non è più usato.
  - Il requester REQ adesso si apre nella directory corrente.
  - Fatti dei miglioramenti minori.
-

12/06/95: V2.1

- Aggiunto le 'Azioni Generiche' Classe BUILT-IN.
- Aggiunto il comando exec SURE.
- Aggiunto il comando ASCII[] di Offset per riconoscere i file ASCII.
- Aggiunti i comandi ARexx: AppIconify, Mostrare, Stato, GetClass.
- Cambiata l'esecuzione del sistema dei task, ora uso Systemtags(). Ancora non abbiamo bisogno di file tmp.
- Adesso puoi definire un ritardo per eseguire il modo CLI.
- Aggiunto 'stringa' e "stringa" per la definizione di offset.
- L-output dei ToolType è obsoleto, noi usiamo il nuovo ToolType CLISIZE in sostituzione.
- Il requester azioni è ora bene dimensionato, appare sotto il puntatore del mouse e usa uno schermo pubblico frontmost.
- Invertendo i pulsanti 'Usa' e 'Salva' e aggiunto 'Cancellare' in ClassActionPrefs per seguire l'Amiga prefs sembrare, spostato il tasto 'circa' in angolo in alto a destra come '?'.  
- ClassActionPrefs routine di Gadget ciclici non furono 100 % sistema amichevole e alcuni rappezzi come Ciclo2Menù fa difetti con ClassActionPrefs; Esso è fissato ora.
- Sviluppato il codice recognizer e la routine di Informazione: essi sono fino a 400% più veloce.
- Listview hilight errore di colore rimosso.
- ClassAction non ancora blocca lo schermo Workbench quando AppIconified.
- Un difetto sgradevole trovato e rimosso in ClassActionPrefs.
- Tasto destro del mouse mostra Assegna solo se mouse è nel requester e Mouse di Destra di nuovo porta indietro alla Directory.
- ClassAction ricordare ora la posizione di finestra.

23/05/95: V2.00 ( Maggiore Aggiornamento )

- ClassAction ha ora un AppIcon.
  - ClassAction è ora una commodity.
  - ClassAction ha ora una porta ARexx.
  - ClassAction usare ora il default WB Font.
-

- Exec modo 'ARexx' aggiunto.
- Differenti colori per Directory/File.
- Su/giù gadget aggiunti a ClassActionPrefs.
- 'Usare' gadget aggiunto a ClassActionPrefs.
- Classi sono ora ordinate in ClassActionPrefs Listview.
- APPSTART, ICONNAME, ICONX, ICONY, CX\_Priorità, WBFONT, Output, ICONFILE ToolTypes aggiunti.
- Quando la finestra è iconnified esso ha ora la giusta altezza riguardo il font di default di schermo.
- Nuovo Formato di Salvataggio (CASF20).
- Suffisso/Prefisso Tasto rimosso. Sostituito da MatchName Gadget che accetta qualsiasi Wildcard.
- Config file spostato in ENVARC:
- un installer è ora fornito con l'archivio.
- qualche ottimizzazione di Codice fatta.

05/05/95: V1.43

- Ripulito il codice di requester, ora 5% più veloce.
- ClassAction ora cercare il suo nome usando WBstartup struttura e poi può essere rinominato.

02/05/95: V1.42

- 'Nessun Cli' Exec modo aggiunto a ClassActionPrefs.
- "" sono sempre aggiunti ai nomi dei file anche se non avuto bisogno di esso è più facile per script ARexx.

06/04/95: V 1.4

- Altezza di ToolType aggiunta.
- Qualche ottimizzazione di codice aggiunta.

22/02/95: V 1.31

- Se tu clicchi due volte sullo stesso file la prima azione sarà lanché (esso è più veloce di selezionato la prima azione a mano).
-

- questa versione è ora ShareWare quindi devi registrarti per avere la versione registrata.

15/01/95: V 1.3

- Aggiunto REQD[] e REQF[] interattivi comandi.
- Copiare gadget aggiunto a ClassActionPrefs.
- minore improvements fatto.
- Beta-tester riportati questa versione è davvero stabile.

25/11/94: V 1.22

- Prima Versione Pubblica.
- Tasto 'Informazione' Aggiunta.
- ClassAction era Locking() le directory che esso legge senza UnLocking() essi .... \*Fissati\*
- Ottimizzazione di Codice.
- Minori altri Difetti rimossi.

08/11/94: V 1.21

- Programma stava schiantando con unità di dischetto vuote .. \*Fissate\*
- Volume/Nome Difetto eliminato
- <..> Voce rimossa quando radice di un volume.

07/11/94: V 1.2

- Nuova interfaccia, io ho incluso il mio proprio veloce file-requester.
- Aggiunti i ToolType STARTDIR & Drive1 Drive11.

01/11/94: V 1.1

- Adesso viene usata la libreria xfdmaster per riconoscere e decomprimere file.
- Configurazione con ToolTypes aggiunti (Decomprime,AUTOSELECT).
- 'Classe Ignota' è ora una classe BUILT-IN con azioni illimitate.
- Nuovo formato di salvataggio (CASF11).

16/10/94: V 1.0

---



- Nuovo Formato di Salvataggio (CASf10).
- Finestra ha ora un gadget zoom.
- molte definizioni di classi aggiunte.

10/10/94: Beta Versione

- tmp difetto di file rimosso.
- Offset incrementare errore rimosso.
- usando asl libreria per il file-requester.

01/10/94: Alpha Versione

## 1.27 The Arexx Commands

ClassAction ha una porta ARExx chiamata: CLASSACTION.1

Questa è la lista di tutti i comandi Arexx:

=====

Quit

Semplicemente abbandona ClassAction...

=====

Use

Forza ClassAction a ricaricare il file prefs.

=====

Ver

Mostra la versione di ClassAction.

=====

Status

Ritorna lo stato corrente di ClassAction.

Ritorna 0: ClassAction è AppIconnified.

Ritorna 1: ClassAction è in modo di Finestra.

=====

---

AppIconify

Forza ClassAction a AppIconify.

Se ClassAction è già un AppIcon, questo comando non farà niente.

=====

Show

Forza ClassAction a mostrare la finestra principale.

Se ClassAction è già una finestra, questo comando non farà niente.

=====

Load <nome del file>

ClassAction prova a caricare il file <nome del file>, e apre il requester delle azioni, lasciando all'utente la scelta di uno di essi.

Se il file non esiste, questo comando non farà niente.

=====

GetClass <nome del file>

ClassAction prova a caricare il file <nome del file> e ritorna alla classe del file caricato.

Se il file non esiste, questo comando non farà niente.

=====

Action <nome del file> <Azione Pattern>

ClassAction esegue la prima azione che trova associata al file <nome del file>.

esempio : Action ram:toto.lha extr

esegue la prima azione che contiene la stringa 'extr' sul file in RAM:toto.lha

Il pattern è utile, non devi dare il nome esatto dell'azione, ma ne basta solo una parte.

Nota: Guarda sulla directory Rexx/ dell'archivio per trovare gli script ARexx inclusi.

## 1.28 What should you do after install

Dopo l'installazione, tu dovresti avere un ClassAction operativo, con molte definizioni di classi, ma con pochissime Azioni definite per essi (la maggior parte delle classi definite non ha azioni definite).

Dipende da te definire azioni riguardo alla tua configurazione di sistema e i tuoi programmi preferiti da usare.

Allora devi semplicemente caricare ClassActionPrefs e configurarlo per le tue esigenze...

Se non sai come configurare le azioni e le classi, allora leggi questa guida :)).

Dopo aver configurato le classi e le azioni, prova a configurare vedendo il comportamento di ClassAction con i suoi ToolTypes.

Potresti perdere parecchio tempo per eseguire una 'bella' configurazione. Ma una volta finito, è davvero Grande!!!

Un piccolo cenno: non usare troppe classi: più tu definisci, più sarà lungo il tempo impiegato da ClassAction per trovare la classe a cui il file appartiene.

Okay, ora leggi la guida e Buona fortuna!

## 1.29 Message for Philippe THOMAS

Salut Philippe!!!

Je voulais simplement te remercier affluer toute l'aide que tu m'as apportée à la création de ce programme.

Quasiment toutes les améliorations de la version 2.0, c'est toi qui me les as proposées, parfois même avec insistance, stilo le redimensionnement de la fenêtre que je n'ai toujours pas fait.. :(

Encore merci affluer tous les appels téléphoniques; parfois plus d'une heure à discuter des Uncini, SystemTagList, de difetti etc... et ce, même en période d'examen.

Franchement si ClassAction commence à être fresco, c'est beaucoup grâce à toi, je crois que j'aurais eu la flemme de le peaufiner autant si tu n'étais pas là.

Bref, ce programme est aussi un peu le tien.

Okay Phil, à la prochaine.

Salim.

---

PS : Non, non ton processeur il est bien, il est pas buggé... :)).

## 1.30 Known Bugs

Attualmente sembra che non ci sia nulla da riportare.

## 1.31 Rawkey codes

Codice di Key

-----

BACKSPACE 65

Cursore Giù 77

Cursore a sinistra 79

Cursore a destra 78

Cursore Su 76

DEL 70

Insert 67

ESC 69

F1 80

... ..

F10 89

Help 95 (fare attenzione, questo key è già definito!)

Return 68

TAB 66