

Auto C.L.I.

di Andrea de Prisco

Eccoci dunque alla seconda puntata di Amighevole riguardo il sistema operativo di Amiga, l'AmigaDOS. Lo scorso mese abbiamo mostrato un modo assai semplice per attivare il CLI (Command Line Interface, l'interfaccia utente per accedere ai vari comandi del sistema operativo) nonché l'organizzazione ad albero delle directory e i primi comandi per visitare una di queste o per navigare all'interno dell'albero di tutti i file. Come prima applicazione strettamente amighereccia abbiamo infine mostrato il comando RUN che implementa il multitasking al livello di CLI: manda in esecuzione un programma, parallelamente all'interfaccia utente che resta libera di ricevere nuovi ordini da tastiera.

L'argomento di questo mese, come già anticipato trenta giorni orsono, riguarda la preparazione di un dischetto che, inserito alla richiesta del workbench dopo il kickstart o dopo un reset (ctrl+Amigapiena+Amigavuota), permette di accedere immediatamente al CLI senza stare a cliccare sulle icone del workbench con conseguente perdita di tempo e di memoria per mantenere il workbench stesso. Con l'occasione una manciata di nuovi comandi AmigaDOS, giusto per gradire.

Chiedo venia

Seguendoci passo passo sullo scorso numero, qualcuno certamente si sarà chiesto come «uscire» da questo benedetto CLI o meglio, come tornare al workbench da cui eravamo partiti. Effettivamente, per una svista del sottoscritto non abbiamo parlato del comando ENDCLI che, come è facile intuire, serve appunto per disattivare il CLI di cui ci stiamo servendo. Né tantomeno possiamo scusarci tirando in ballo questioni di spazio editoriale, in quanto ENDCLI si esplica da solo, bastava solo dire che esiste. Insomma chiedo umilmente scusa e perdono.

Qualche altro comando

Detto questo, infiliamo la mano nel

sacchetto dei comandi, e dopo una brevissima rimestata estraiamo il primo comando di questo mese. Vediamo un po'... PROMPT.

Serve per cambiare il prompt del CLI: se non ci piace l> possiamo cambiarlo con:

PROMPT ciocheicipiace
dove ciocheicipiace è la stringa di caratteri che desideriamo come prompt. Ad esempio, se rimpiangiamo il nostro amato 64 potremmo (anche se è una buona offesa) digitare

PROMPT READY

se vogliamo qualcosa di più inusuale (vedi foto 1) potremmo scegliere un più garbato

PROMPT dica,dica! (senza spazi, ndr.)

resta inteso che per ripristinare il vecchio buon l> basterà digitare

PROMPT l>

e il gioco (perché di gioco si tratta) è fatto.

Due comandi di natura più classica, RENAME e DELETE permettono di cambiare nome o cancellare un file o una intera directory presente su disco. La loro sintassi è la seguente. Con:

DELETE NomeFile

cancelleremo dal disco il file indicato di seguito al comando delete. Con:

DELETE NomeDirectory ALL

cancelleremo dal disco tutta la directory di nome NomeDirectory. Con

RENAME NomeFile1 TO NomeFile2

avremo l'effetto di cambiare nome al file NomeFile1 in NomeFile2 (identicamente per cambiare nome a una directory). Si noti che (crepi l'avarizia) l'AmigaDOS accetta anche altre forme sintattiche di questo stesso comando e precisamente avremmo potuto scrivere anche:

RENAME FROM NomeFile1 TO NomeFile2
oppure:

RENAME NomeFile1 AS NomeFile2

Riassunto dei comandi AmigaDOS finora mostrati.

Comando	Sintassi	Descrizione
DIR	DIR DIR path	Legge la directory corrente Legge la directory indicata da path
CD	CD path CD / CD	Passa alla direct. indicata da path Torna alla directory padre Mostra il percorso finora compiuto
RUN	RUN NomeProgramma	Manda in esecuzione, parallelamente al CLI, il programma NomeProgramma
ENDCLI	ENDCLI	Termina la sessione CLI nella quale e' impartito tale comando
DELETE	DELETE NomeFile DELETE NomeDir ALL	Cancella dal disco il file NomeFile Cancella dal disco tutta la direct. di nome NomeDir
RENAME	RENAME Nome1 AS Nome2 RENAME Nome1 TO Nome2 RENAME FROM Nome1 TO Nome2	Cambia il nome al file Nome1 in Nome2. N.B.: 3 forme semanticamente uguali
PROMPT	PROMPT stringa	mostra "stringa" ad ogni prompt
SAY	SAY SAY stringa	Forma interattiva Recita la stringa indicata dopo SAY
WHY	WHY	Motiva una condizione d'errore
FAULT	FAULT n	Mostra l'errore corrispondente al numero n
N.B.: NomeFile, NomeProgramma, NomeDir comprendono un eventuale path		

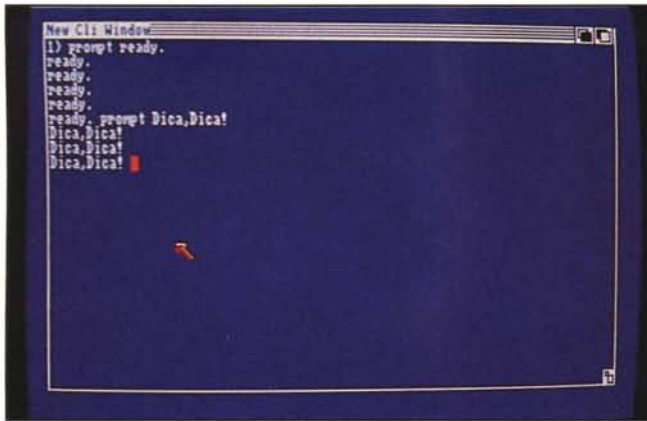


Foto 1 e 2



Foto 3

Tanto per RENAME quanto per DELETE, come per qualsiasi altra operazione che coinvolge file, dovremo sempre specificare il cammino da compiere nelle subdirectory per il raggiungimento del file in questione. Di questo ne abbiamo già ampiamente parlato lo scorso mese quindi vi rimandiamo alla puntata precedente per tutti i chiarimenti in merito. Lo abbiamo ricordato solo per mostrarvi un esempio da tenere in considerazione in futuro.

Immaginiamo di dover trasferire un file da una subdirectory ad un'altra. Di primo acchito si sarebbe tentati di fare un copy seguito da un delete per togliere la vecchia copia. In questo modo infatti otterremmo il voluto con poco sforzo, a condizione però che si sia abbastanza spazio sul disco per mantenere anche se per pochi secondi due copie dello stesso file. Se ciò non fosse possibile sarebbe un bel guaio: dovremmo fare spazio in qualche modo sul disco o salvare il file temporaneamente da qualche altra parte per procedere all'operazione. Nulla di più falso, naturalmente.

Il problema si risolve molto più facilmente col comando RENAME dato che il percorso per raggiungere un file è indicato nel nome stesso. Se ad esempio vogliamo trasferire il comando DIR dalla directory C (che contiene tutti i comandi dell'AmigaDOS) nella directory principale scriveremo:

```
RENAME C/DIR AS DIR
```

mentre per ripristinare DIR in C:

```
RENAME DIR AS C/DIR
```

Se volete sentir parlare il vostro Amiga è disponibile il comando SAY in forma di comando o in forma interattiva. Digitando:

```
SAY qualcosa
```

avremo l'effetto di sentire «qualcosa» uscire dall'altoparlante del monitor. Se invece digitiamo SAY senza indicare nulla al suo seguito, otterremo la versione interattiva (vedi foto 2) che presenta due finestre una per digitare le frasi da recitare mentre sull'altra appaiono i fonemi. È possibile cambiare tono e velocità della voce nonché

uscire dal SAY stesso battendo return senza aver indicato le parole o la parola da pronunciare.

Per concludere questa seconda cartellata di comandi AmigaDOS, mostreremo due utility presenti sul disco che permettono di avere chiarimenti (qui non si sono sprecati molto) circa i messaggi di errore che da CLI si possono presentare. Il primo di questi due comandi è WHY e serve appunto per chiedere perché una data operazione è fallita (quindi è da digitare dopo il verificarsi di un errore). Purtroppo spesso e volentieri sembra più una presa per i fondelli che un vero aiuto uscendo con delle giustificazioni alla Ferrini come mostrato in foto 3 dove Amiga non avendo spazio in memoria al comando say aveva giustamente risposto che non era in grado di caricarlo e al comando why (volevamo saperne di più) ci ha risposto che (vedi sempre foto 3) il precedente comando è fallito perché non c'era memoria.

Della serie: «non c'è memoria, non mi adegua e quindi ho ragione, lo dice il ragionamento stesso».

Leggermente più interessante, il comando FAULT permette di convertire il numero di un errore in una frase più esplicativa. Ad esempio, in seguito a qualche operazione il sistema potrebbe segnalare error 221. Se non abbiamo sottomano il manuale della macchina e proviamo a digitare WHY (ovvero: perché c'è stato l'errore 221?) Ferrini da dentro all'Amiga risponderà che l'ultimo comando è fallito in seguito a... errore 221 (stessa «serie» di sopra). Digitando di contro FAULT 221 otterremo la frase Disk Full che vale molto di più, ed è effettivamente quanto ci è successo; il nostro disco è pieno zeppo e non si può procedere a qualche operazione di scrittura.

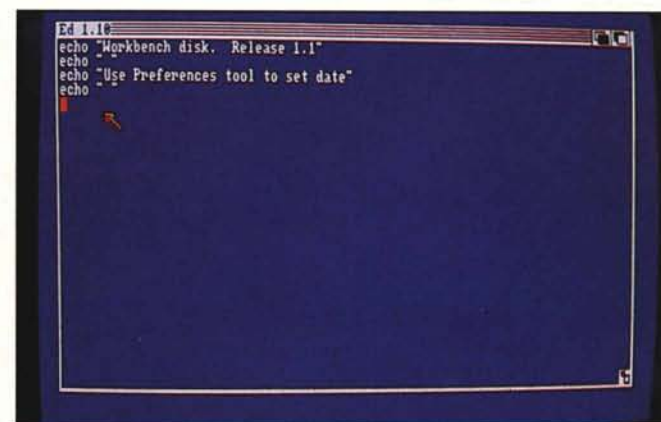
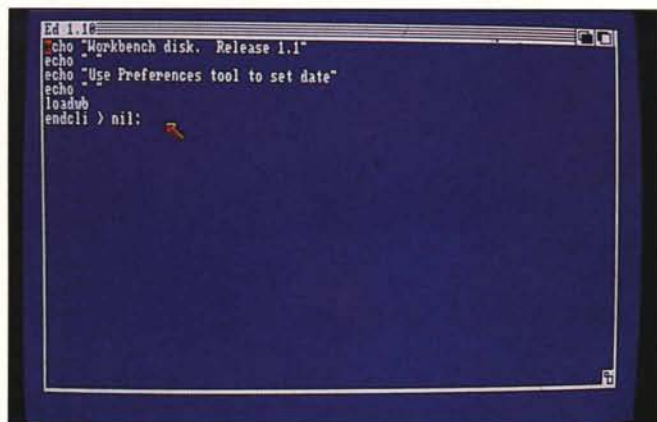


Foto 4 e 5

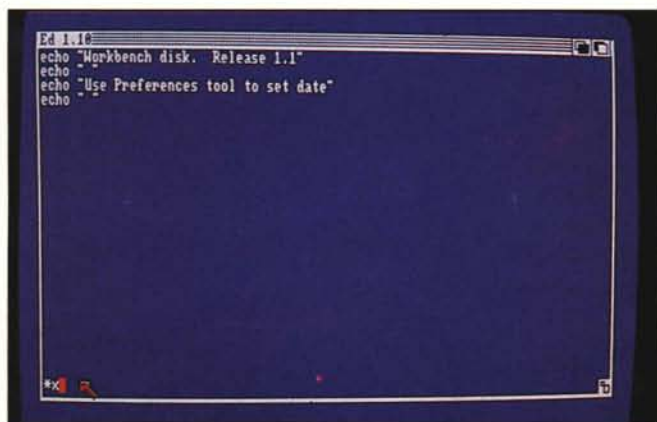


Foto 6

L'editor ED

All'interno del sistema operativo di Amiga esiste (come in qualsiasi S.O. che si rispetti) un buon editor per creare e/o modificare file di testo. Una sorta di semplice word processor fatto apposta per le «piccole cose di tutti i giorni» e che certamente non guasta. Tantopiù che il fatto che è locato su disco e quindi occupa spazio in memoria (approx. 18K) solo quando è in funzione.

Per attivarlo, essendo anch'esso un comando, basta digitare ED seguito dal nome del file da editare. Se tale file è presente sul disco, sarà caricato in memoria per essere editato, altrimenti l'editor partirà col buffer vuoto e quando si salverà il «digitato» questo avrà il nome indicato all'inizio. Semprechè lo desideriamo: possiamo anche salvare con un nome diverso, ovviamente.

Ma non finisce qui: con l'editor ED possiamo fare anche un altro bel mucchio di cose: settare i margini destro e sinistro, ricercare parole nel testo in avanti e all'indietro, cambiare automa-

ticamente ogni occorrenza di una parola con un'altra, spostare blocchi di testo e altre cose.

Così tante che dedicheremo in futuro una intera puntata a questo potente strumento. In questa sede lo abbiamo nominato dovendocene servire per preparare il dischetto che va automaticamente in CLI al momento del boot o del reboot. Ma prima di procedere all'operazione, che come vedremo ci impegnerà per pochi secondi, occorre anticiparvi qualcos'altro in merito a quanto succede...

Al momento del Boot

Quando inseriamo il disco del workbench dopo il kickstart o dopo un reset CTRL+Amigapiena+Amigavuota il sistema carica prima qualcosa da disco, poi scrive alcuni messaggi su video (tra cui «use prefereces tool to set date») e poi carica qualcos'altro prima di mostrare le icone del workbench.

Per la cronaca (sono tutte anticipazioni) la prima «caricata» riguarda alcune librerie di sistema che devono necessariamente stare in memoria per

un corretto funzionamento del tutto. La seconda «caricata» è invece relativa al workbench vero e proprio ed è questa che dobbiamo eliminare. Infatti, dopo la prima «caricata» siamo già in CLI ed è proprio da CLI che vengono dati i messaggi e l'ordine di caricare la seconda parte.

Curiosando un po' nelle directory del disco workbench, nella subdirectory S troviamo un file di nome Startup—Sequence che guardacaso è proprio un insieme di comandi AmigaDOS (separati da [RETURN]) che, costituendo di fatto un programmino, descrivono il da farsi al momento del boot o del reboot.

Se ad esempio desideriamo l'auto-start di un programma contenuto nel disco non abbiamo che da inserire il suo nome nella Startup—Sequence per ottenere il voluto. Oltre a questo, in AmigaDOS è possibile preparare a volontà questi file di comandi da mandare in esecuzione col comando EXECUTE. In questo modo è possibile creare una propria personalizzazione del CLI con l'uso di queste MACRO, che come vedremo in seguito possono anche accettare parametri in ingresso ed essere strutturate con tanto di IF—THEN—ELSE.

AutoCLI

Siamo pronti per preparare il nostro disco AutoCLI ovvero un disco che al momento del boot o del reboot non va in workbench, ma mostra il prompt dell'AmigaDOS con tanto di cursore lampeggiante. La prima operazione da compiere sarà naturalmente quella di eseguire una copia del nostro workbench in modo da non manomettere il disco originale (non si sa mai). Per eseguire la copia attenersi a quanto descritto sul manuale fornito con la



Da 10 anni, in queste tre

macchina a pagina 4 — 31.

Come indicato sul numero scorso, sul disco copia rendete attivo il CLI selezionando preferences e da questo clickando su CLI ON. Salvate la preferenza modificata e inserite il disco copia nel drive eseguite il reboot tramite l'arcinota sequenza CTRL+Amigapiena+Amigavuota. Ciò serve per rendere il disco copia il nuovo disco sistema di questa sezione di lavoro. Sempre come indicato sul numero scorso, selezionate l'icona CLI e cliccate due volte su questa per ottenere il prompt dell'AmigaDOS. (Nota bene: fin qui non è stato detto niente di nuovo). Digitate ED S/STARTUP—SEQUENCE e, colpo di scena (foto 4), vedrete apparire la sequenza di comandi che al momento del boot sono eseguiti automaticamente dal sistema.


Le prime righe di queste sono i messaggi che vengono stampati su video (il comando echo è praticamente un PRINT) seguono il comando loadwb che carica il workbench e l'endcli che disattiva il cli. Forza e coraggio: posizionatevi col cursore su ognuna di queste due ultime linee (foto 5) e coi tasti DEL e BACKSPACE presenti sulla vostra tastiera cancellatele. Solo a questo punto, premete il tasto ESC seguito da X (foto 6) per uscire dall'editor salvando la nuova Startup—Sequence. Fatto!

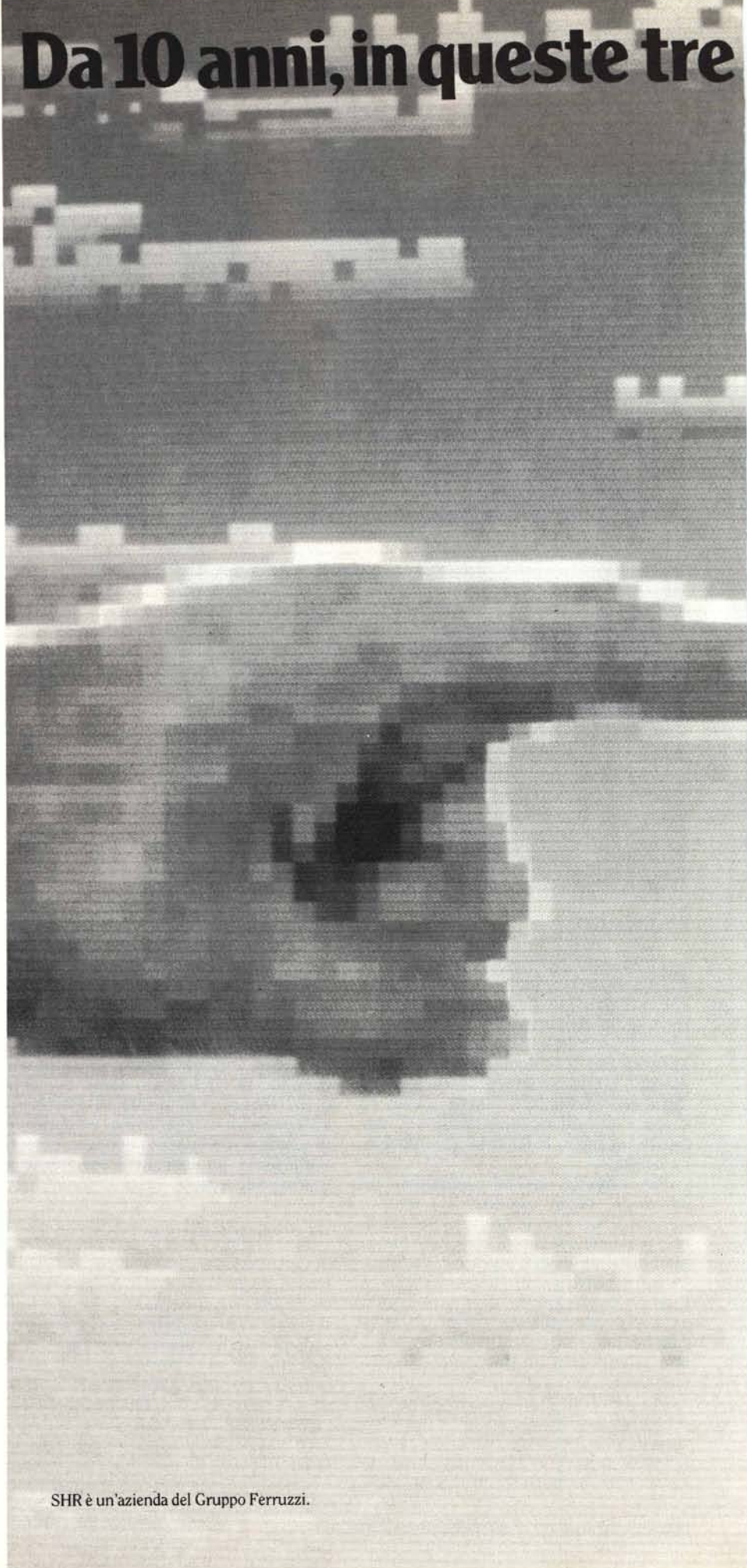
Già fatto? Sì. Incrociate pure le dita e effettuate il reboot: se tutto è andato bene otterrete direttamente il cursore lampeggiante che aspetta i comandi AmigaDOS che avete già imparato ad usare.

A proposito: resta inteso che se da CLI volete passare al workbench dovette prima caricarlo col loadwb che non è stato eseguito nella Startup—Sequence e poi digitare endcli.

Leviamo un po' di roba

Posto che questo nuovo disco lo useremo prevalentemente in CLI, possiamo togliere un po' di roba che occupa spazio, ma che certamente da CLI non useremo.

Possiamo certamente togliere le directory Trashcan, Demos, Empty, System e Utilities ricordandovi di usare l'opzione ALL del comando DELETE. Analogamente possiamo togliere Clock e tutti i «file.info», che contengono la forma delle rispettive icone, relativi agli oggetti che abbiamo appena scratch—ato, nella fattispecie: Clock.info, Trashcan.info, Demos.info, Empty.info, System.info, e Utilities.info. Punto e basta. 



SHR è un'azienda del Gruppo Ferruzzi.

lettere c'è una grande esperienza.

GruppoEthos

SHR

Software.

Questa "S" risolve ogni giorno centinaia di problemi come il tuo, perché significa soluzioni innovative garantite da migliaia di installazioni in Italia.

Hardware.

Questa "H" ti dà la soluzione che non trovi altrove, perché ti offre una gamma completa del miglior hardware con la massima garanzia e assistenza, oltre ogni tua aspettativa.

Ricerca.

Questa "R" significa ricerca: cioè un enorme impegno per essere avanti, con soluzioni sempre aggiornate e sicure nel tempo.

