

MacsBug

di Raffaello De Masi

Realizzato specificamente per Apple, MacsBug è un debugger simbolico a basso livello. Dal 1995 è disponibile in due versioni, destinate a macchine 68 e PPC (si noti nella figura, la notevole differenza di dimensioni delle due applicazioni).

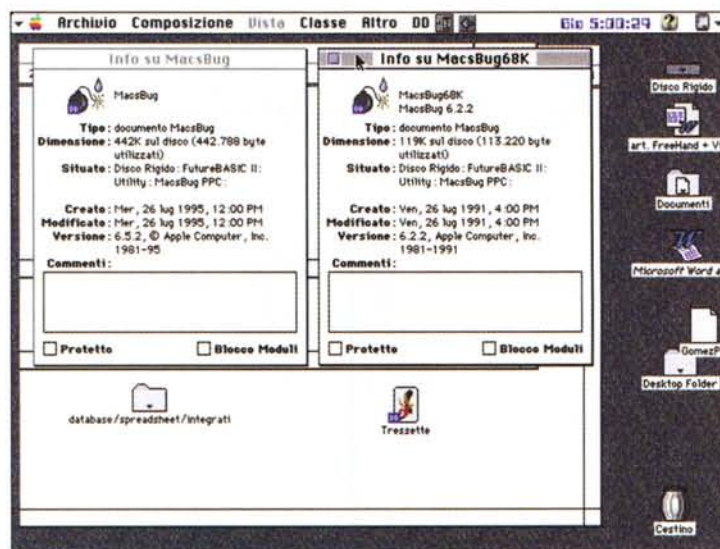
Sentir parlare di debugger low level, per un neofita, è cosa da far tremare le vene e i polsi. In effetti MacsBug è un tool che in mano a uno specialista permette di raggiungere informazioni, nel programma, quasi inaccessibili e rappresenta un potente aiuto nella fase di debugging, appunto, di un programma, qualunque sia il linguaggio che stiamo utilizzando. Ma, cosa senz'altro più interessante, usato con giudizio permette di radiografare applicazioni commerciali che pongono problemi alla nostra macchina, permettendo così di ottenere soluzioni a problemi che altrimenti sarebbero indecifrabili e insuperabili. Ovviamente, inutile nascondere, esiste il contraltare di lavorare con un tool piuttosto ermetico, talora criptico e stringatissimo nelle spiegazioni, e comunque bisognoso di una buona collaborazione da parte dell'utente-programmatore.

MacsBug non è in commercio direttamente, ma è presente su CompuServe, America OnLine ecc. Esso, inoltre è fornito di serie con numerosi linguaggi, come alcuni C, Future Basic, BP Basic e così via.

Molti sviluppatori considerano MacsBug un tool di sviluppo essenziale, anche se spesso non conoscono tutte le sue potenzialità. Occorre precisare che questa impostazione, anche se non pienamente corretta, è pur sempre valida, in quanto consente di trarsi d'impaccio da situazioni che richiederebbero fustiganti operazioni di recupero o, al limite, il reboot del sistema.

L'uso di MacsBug

In barba alla notevole differenza di dimensioni delle due implementazioni, i comandi sono identici per le due versioni. Installare MacsBug è abbastanza semplice; è sufficiente trascinare l'applicazione nella cartella sistema (in questa deve essere presente a livello di prima radice; in altri termini non va messo in sottocartella).



Le due MacsBug a confronto; si noti come la versione per PPC sia molto più grande di quella per il 68K.

I comandi principali di Mbug

Diamo di seguito l'elenco dei comandi utilizzabili nella finestra di MacsBug, in particolare quelli di più semplice uso, che non richiedono conoscenze di assembly e che hanno, immediatamente, un effetto in caso di blocco del sistema. Ricordiamo comunque che MacsBug è, più ampiamente e efficacemente, implementato e collegato con opportune chiamate e spesso col suo menu, in pressoché tutti i linguaggi disponibili sul mercato (eccellente, ad esempio, l'implementazione in Future Basic, oggi alla versione II, di cui forniamo una videata).

? [Help]; offre una concisa descrizione di tutti i comandi; può essere seguito da un comando, nel qual caso verrà spiegato solo quello.

ES [escape to shell]; dove lo shell è il Finder. Da qualunque situazione sia invocato, permette di uscire dall'applicazione disastrosa e di tornare al Finder, senza il reboot della macchina.

RS, RB [Restart e Reboot]; ambedue eseguono il rilancio del sistema; differiscono solo nel modo di maneggiare i drive di network. È l'ultima risorsa in caso di disastro grave.

HC [Heap Check]; comando diagnostico in senso stretto, verifica che l'Heap non sia danneggiato, nel qual caso è pensabile possano esserci più seri problemi nella gestione dell'applicazione (in caso di risposta positiva, quasi sempre MB non sarà capace di eseguire un recupero). Sovente, con un poco di

esperienza, l'analisi dell'Heap permette, a un buon programmatore, di valutare la gravità dell'errore.

HT [Heap Total]; verifica quanti blocchi sono presenti nell'Heap, e quanto spazio occupano; può essere utile per verificare lo stato dell'Heap.

S,T,G [Step, Trace, Go]; dall'ovvio significato; nell'ordine: esecuzione di una istruzione alla volta; di una istruzione o subroutine alla volta; la spiegazione della terza istruzione è ridondante.

tilde (~) [escape]; permette di passare alternativamente dalla videata di MB all'ultima dell'applicazione che ha generato l'errore.

le). All'avvio un messaggio nella finestra di startup avvisa del suo caricamento (esso segue, nell'ordine di lancio, solo RAM Doubler, ove mai fosse installato); dopo di che MB non dà più segnali della sua presenza fino a quando non ce n'è bisogno (per coloro che sono corti di memoria, specialmente sul loro PPC, nessuna paura; la maggior parte delle risorse dell'applicazione è del tipo «purgeable», vale a dire che il sistema operativo provvede a ridurre la presenza di esse dalla RAM ove ce ne fosse necessità).

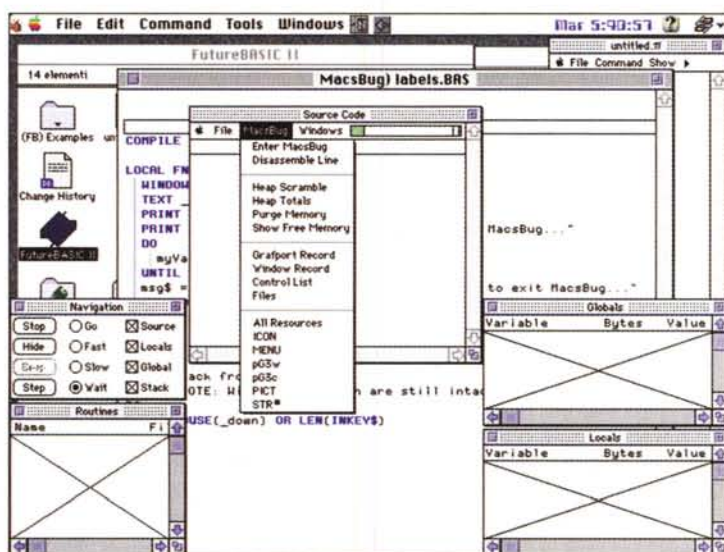
Ci sono due modi di invocare MacsBug; il primo, un po' brutale, è quello di premere il bottone di programmatore presente sul lato o sul davanti della

macchina (per i modelli più nuovi il bottone è sostituito da una combinazione di tasti). In questo modo si invoca la pagina di MacsBug, pagina principale e unica che, in assenza di altre necessità, mostra lo stack, lo stato dei registri di CPU e parte del codice.

Detto senza mezze bugie, abbiamo davanti uno spaccato dello stato della memoria in formato assembly. Ma, per carità, non girate pagina! Vi possiamo assicurare che il linguaggio assembly è molto meno misterioso di quello che può sembrare, basta guardarlo come a un idioma che maneggia le informazioni a livello atomico. Con un poco di pratica e di esercizio, si possono ricavare da

esso molte più informazioni di quanto si possa pensare; ma anche per chi di assembly non ne vuole proprio sapere, è ancora interessante continuare a leggere; vi daremo molte altre di salvataggio per il vostro disastro informatico con bomba.

Perché è proprio in questi casi che MacsBug serve più spesso all'utente di tutti i giorni; il secondo modo, infatti, di invocare MB non dipende dalla nostra volontà. Immaginate (a chi non sarà successo) una di quelle applicazioni che, da un momento all'altro, decidono di andare in bomba; finora l'unica possibilità era quella di resettare il sistema. Oggi, con un poco di fortuna e facendo i passi giusti, con l'aiuto di MB non tutto è definitivamente perduto. Basta leggere e tenere a portata di mano l'elenco dei comandi che vedete nel riquadro.



La chiamata al debugger dall'interno di un linguaggio, in questo caso il Future Basic II.

Conclusioni

MacsBug è un vero e proprio mezzo marinaio in una tempesta; non può certo risolvere tutto, ma ve la immaginate una situazione in cui, col vostro linguaggio, siete entrati in un loop da cui non riuscite a uscire neppure con il tuffare [Cmd-].? Oppure una applicazione si presenta con la sua finestrina [error XXX; unrecoverable error trap]? O ancora schermo e mouse si congelano e vi viene voglia di dare un calcio alla macchina? MB non è tuffare, certo, ma è pur sempre un airbag, un salvagente e una saponetta di BlackSoap (il sapone repellente presente nei kit di sopravvivenza per difendersi dagli squali) nel momento della rovina totale. Provare per credere!

Questi sono i comandi principali; ne esiste poi una miriade d'altri, destinati a un utente più avanzato, alcuni dei quali molto potenti. Di questi diamo un breve accenno, rimandando per la completa descrizione degli altri al manuale che è fornito insieme al programma stesso e che è reperibile allo stesso modo.

IL, IP [Instruction List, Instruction List Page]. Si tratta di comandi specifici del disassembler, che agiscono differenzialmente su indirizzi, specificati o non.

DM, MS [Dump Memory, Set Memory]; per il display e il cambio del contenuto della memoria.

BR e BRD [BreakPoint e BreakPoint Display]; utilissimi, settano un BP o ne mostrano il set.

ATB (A-Trap Break); anch'esso di grande utilità, causa un break ogni volta che viene eseguita una specifica chiamata al ToolBox o al SO.

GT [Go To]; esegue un programma fino a uno specifico indirizzo.

SP [Step Spy]; ancora un mind-saver. Eseguendo un programma fino a che una specifica locazione di memoria cambia; sovente viene abbinato con un altro comando, DX [Toggle Debugger Break] che permette di accendere e spegnere gli user-break in MB.

LOG; scarica tutti gli output di MacsBug su un file o su una stampante.

MC [Macro Command]; consente di definire una macro di comandi MB, separati da un [:]; ha come compagno MCD [Macro

Command Display], che permette di visualizzare, oltre le macro definite dall'utente, anche quelle built-in di MacsBug, il cui numero, ben superiore a quello che vedete, è davvero impressionante.

Sempre a proposito di comandi, una delle possibilità più interessanti è quella che MB è facilmente espandibile.

Per inserirne qualcuno di nostra costruzione (vale a dire non realizzabile con una macro) la via migliore è aprire il programma con un editor di risorse (ad esempio Resedit o Resourcer) e inserire la risorsa da noi creata nella lista di quelle già esistenti, ovviamente rispettandone il tipo (ricorderemo che una risorsa va definita con una stringa di quattro lettere; anche quelle che ne mostrano solo tre, ad esempio «snd» possiedono, come ultima lettera, uno spazio bianco).