

Microsoft FoxPro 2.6

di Raffaello De Masi

Personalmente ritengo Naomi Campbell la ragazza più bella del mondo, ma democraticamente riesco ad ammettere che qualche orbo non possa condividere la mia idea (ve la immaginate tutta l'umanità maschile innamorata della stessa donna?). E questo vale per tutte le cose della vita, ivi compreso mio fratello, che scarozza moglie e figliolanza plurigemellare in una Uno (ironia del nome) di quasi duecentomila chilometri, a suo dire ancora «fresca». Non parliamo poi dell'informatica, dove utenti scrivono ancora in WordStar e progettano edifici faraonici utilizzando ancora il loro gagliardo Apple II o qualche vetusto Z80 o 8088.

Un vecchio proverbio napoletano così recita: «Don Gennari», dicette Pulci-

nella, «voi col sigario e io col mozzone, cacciamo 'o stesso fumo». Partendo quindi dal principio che «pure i fegatelli di mosca fanno sostanza», e che non esiste categoria (tranne, forse, i cacciatori e pescatori) più legata alle proprie convinzioni, abitudini o manie, degli utenti informatici, non abbiamo mai preteso che quanto diciamo da queste pagine possa essere accolto da chi legge come vangelo, o anche solo come cortese suggerimento a cambiare.

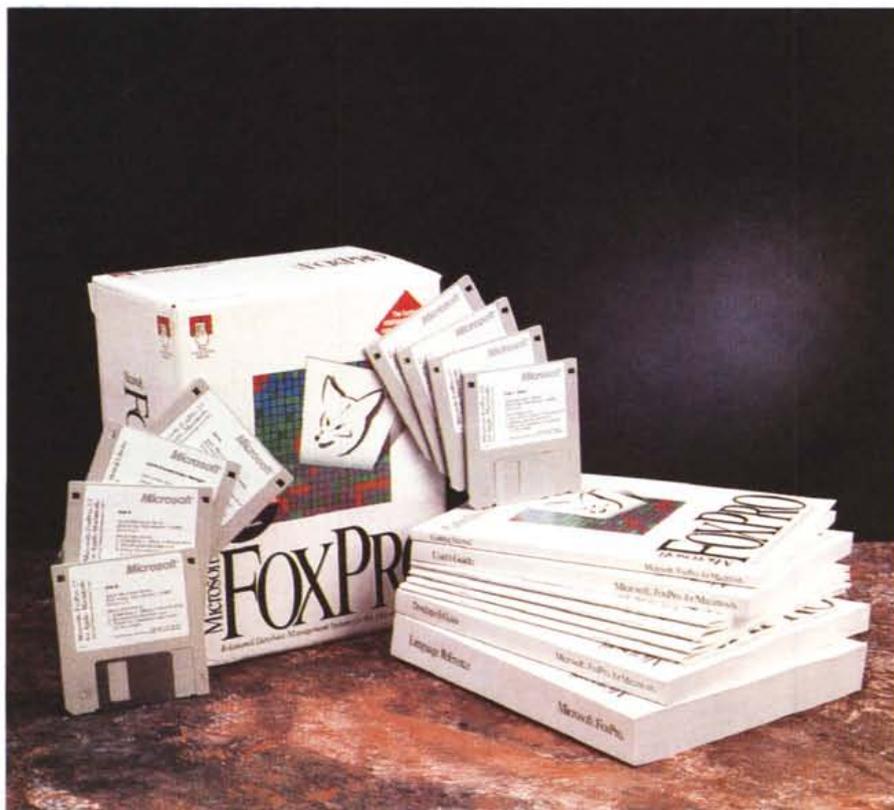
La cosa è tanto più vera nel caso dei database e questo per una serie di motivi, anche oggettivi; mentre per i wp o gli spreadsheet esiste una, potremmo dire, scuola di pensiero unica, per cui il passaggio da una parrocchia all'altra non genera soverchie preoccupazioni

né radicali cambiamenti di stile di utilizzo, i db sono spesso del tutto dissimili nell'impostazione, nell'uso, nella potenza, nella facilità di realizzazione di un progetto, per non dire nell'utilizzo. Chi ha avuto modo di utilizzare, anche non comparativamente, FileMaker e Omnis (anche se il paragone è improponibile come il confronto di una Punto con una Celica) non avrà certo potuto non notare l'estrema semplicità di gestione e d'uso del primo; e certamente a chi dovesse utilizzare un db con un file piatto, per esempio per gestire una biblioteca, non mi sentirei, in cuor mio, di proporli di comprare 4D o Omnis, appunto.

Fatto sta che, comunque, quando le cose si complicano e ci si trova a gestire grandi masse di dati relazionali, fortemente articolati e gerarchizzati, ci si scopre a vedere la trama di certe applicazioni. Una cosa è fare footing attorno all'isolato o giocare da terzino nella partita Avellino-resto del mondo, un'altra è partecipare alla maratona di New York o al Gran Premio di Montecarlo. E così ci si ritrova, nel mondo Mac, a guardarsi attorno e a ritrovarsi con pacchetti davvero poderosi, potenti, dai polmoni e dalle spalle possenti. E FoxPro, ve lo posso assicurare, è davvero un Mohammed Ali nel suo genere.

FoxPro; il pacchetto

Fornito in una monumentale e pesantissima confezione, FoxPro consta di ben otto dischetti che, decompressi, creano una cartella di ben 17 Mb (il programma nudo e crudo è ben 3 Mega,



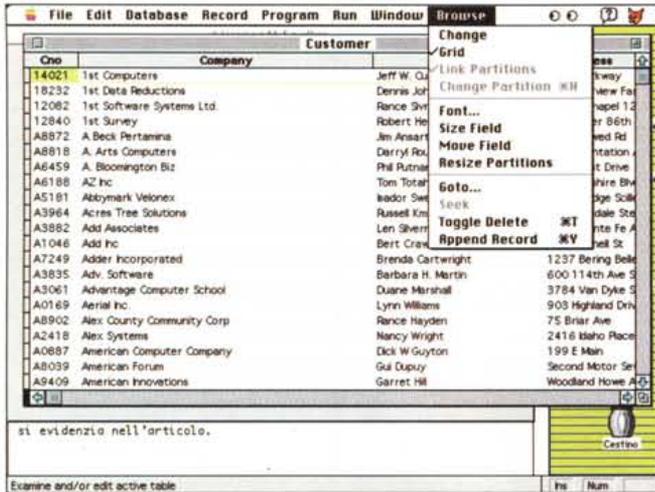
Microsoft FoxPro 2.6

Produttore e distributore:

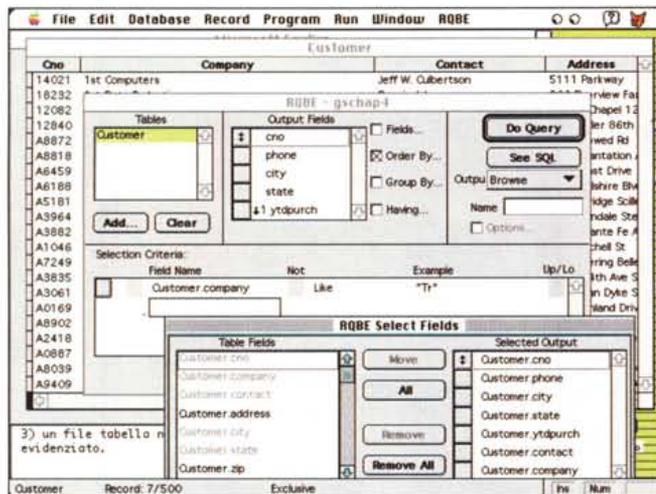
Microsoft S.p.A.
Centro Direz. S. Felice palazzo A
Via Rivoltana, 13 - 20090 Segrate (MI)
Tel.: 02/70392

Prezzo indicativo (IVA esclusa):

Microsoft FoxPro 2.6 Lit. 999.000



Un file tabella nel formato «browse», con il relativo menu evidenziato; nella figura qui di destra, la finestra di «query» con una fase di definizione delle modalità di campo.



«Open»; vero, ma qui le cose funzionano diversamente.

Permettetemi una precisazione, necessaria a chi è (male) abituato all'ambiente Mac; e immaginiamo di aprire un file di FileMaker Pro, tanto per parlare di qualcosa nota a tutti e a cui tutti possono fare riferimento. Avremo subito a disposizione la scheda di layout, e sarà immediatamente possibile passare ad altri formati semplicemente selezionandone uno dal menu a discesa sopra il rolodex di sinistra; nessun problema neppure per la stampa di un report o per la creazione di un etichettario, sono tutti lì, a portata di mano. In altri termini vogliamo dire che il file contiene sia i dati che i formati.

4D va un poco più avanti; i dati sono immagazzinati in un file separato da un altro, che contiene tutti i formati e le procedure-macro. L'applicazione è certamente più snella e meglio organizzata; l'importante è legare bene le due

parti, costruendone efficacemente gli script di gestione.

FoxPro spinge ancora più a fondo questo concetto, vale a dire che tutto, ma proprio tutto, gode di vita propria e ha personalità e individualità separata. Esiste il file grezzo di dati, e su di questo si tagliano una serie di maschere di input, di output, di report, di visualizzazione, di progetto, di etichettario, di ricerca e ordinamento, addirittura di screen (vale a dire di formato di schermo). La filosofia è quella di avere, alla base, una massa grezza di dati, su cui poi applicare e adattare maschere e schemi diversi; queste maschere sono veri e propri filtri, destinati a nascondere o a visualizzare dati un poco come le grate di cartoncino utilizzate dalle damine e cortigiane per decifrare messaggi segreti in uno scritto apparentemente banale.

Potrebbe sorgere immediata la domanda: «Vale la pena adottare questa impostazione?» La risposta è senz'altro sì, a patto di impadronirsi presto e bene del linguaggio di scripting, in modo da gestire automaticamente abbinamenti di fogli e mappe diverse. Altrimenti, pensare di guidare il database da chiamate da menu (ancorché possibile) è pura follia. D'altro canto, una volta acquisita la necessaria padronanza del linguaggio, ci si ritrova in mano un mezzo estremamente articolato ed efficiente, veloce e piacevole, facile da usare proprio perché l'applicazione è fatta di tante parti concorrenti. Uno screen, un foglio di report, una tavola di ricerca, grandi non più di qualche k, si debuggano facilmente, si buttano via,

Perché FoxPro è così veloce!

La risposta è brevissima; perché usa la tecnologia RushMore. Si tratta di una tecnologia che permette di accedere in maniera estremamente efficiente a gruppi di record, neppure particolarmente ordinati e indicizzati. Una curiosità, il nome fu scelto da un componente del progetto dopo aver visto il film «North by NorthWest» di Hitchcock.

Con questa tecnologia, alcune operazioni riguardanti gli ordinamenti e le ricerche su tabelle riescono a «viaggiare» centinaia e anche migliaia di volte più veloci dei normali standard operativi. Con questa tecnologia FoxPro permette ai personal di maneggiare file giganteschi, contenenti anche milioni di record, a una velocità comparabile a quella dei supermini o dei mainframe.

RushMore usa gli indici standard B-tree tal quali e non richiede alcuna modifica di essi o dei file. Può essere quindi usato con

indici del tipo [.IDX] (delle prime versioni Fox), compattati (ancora [.IDX]), e composti [.CDX].

È interessante notare che RushMore non fa dipendere la sua velocità dal tipo di indice; il fatto che indici compattati siano più veloci, sotto RM, di quelli estesi dipende solo dalla loro minore grandezza, cosa che richiede, ovviamente, più limitati accessi alla memoria di massa. Sebbene sia comunque una precisazione inutile, RM funziona allo stesso modo su ambedue i tipi. Non è certo qui il luogo e il momento di parlare più ampiamente di RushMore (probabilmente ci sarebbe qualcuno che riuscirebbe ad essere più chiaro di me) tanto per non lasciare a bocca asciutta il lettore diremo, molto limitatamente, che RM usa clausole del tipo FOR per spezzettare il file in sottotabelle. E la spiegazione finisce qui; semmai ne ripareremo qualche altra volta.

se necessario, senza rimpianti, si riuniscono poi insieme in una grande famiglia, concorrenti tutti ad uno scopo. Non a caso, infatti, la costruzione di un'applicazione efficiente si basa, in FoxPro, sulla corretta realizzazione di un «Project», fatto appunto di tanti pezzi tutti concorrenti a uno scopo.

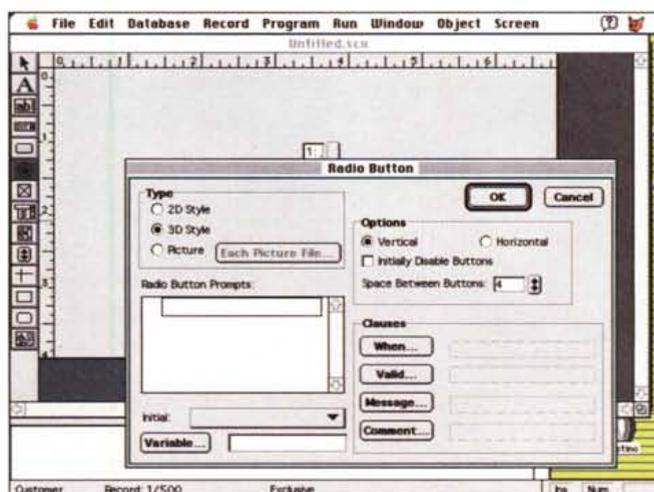
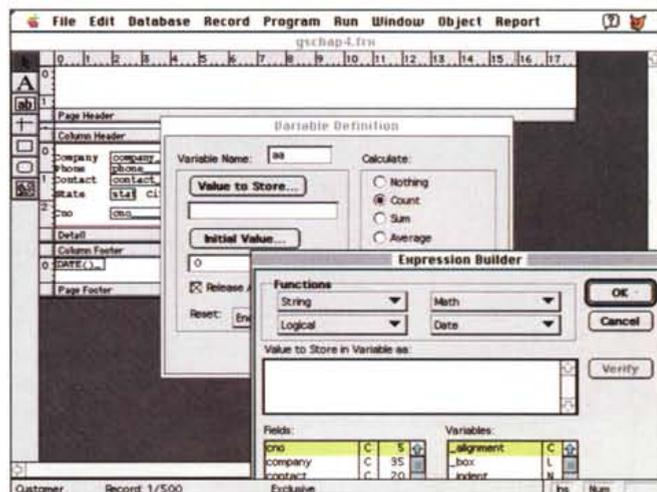
E così, quando chiamiamo «Open», FP pur proponendo in default l'apertura di una tabella (che poi altro non è che un file di dati, grossolanamente organizzato, appunto, in una tabella) offre la possibilità di aprire ben 11 tipi diversi di file. Sebbene, ovviamente la chiamata FILES\$ di Toolbox evidenzia solo i file apribili in base al tipo desiderato, il tipo di documento (e qui si sente la mano dell'«MS-DOS») si riconoscono dal suffisso, quasi sempre, ma non sempre, mnemonico. Avremo così DBF per le tabelle, PRG per i programmi, FRX per i report, SCX per gli screen, MNX per i menu e così via; ma non occorre spremere molto la memoria; ci pensa il programma a «filtrare» i tipi desiderati.

Bene, cominciamo; dobbiamo costruire il nostro bel database, ad esempio una rubrica-indirizzario delle top model che abitualmente frequentano casa nostra. La cosa migliore è creare immediatamente una tabella, forma un poco rudimentale e grezza, certo, ma pure Dio ci mise sei giorni a fare tutto. La tabella, inoltre, è molto pratica perché permette di cambiare, al tocco, la disposizione e l'ordine di accesso ai campi, ed ha il pregio di essere velocissima nelle manipolazioni.

Che fare, allora. Definiamo immediatamente i campi, in una finestrina che ha molto in comune (e non poteva essere che così) con quella di altri pacchetti. I campi vengono definiti per nome, tipo, lunghezza, e vengono organizzati gerarchicamente: i tipi possibili sono otto, tra cui quelli ben noti, come numerico e alfanumerico, data e picture, ma anche nuovi, come memo (campi di grandissima estensione, capaci di contenere documenti complessi prodotti anche con altri programmi), o general (capaci di accettare dati in forma binaria, come disegni, suono o addirittura risorse).

Una volta eseguito il tutto vengono proposti due semplici formati lista e tabella; i formati sono i più semplici possibile (in uno i campi sono l'uno accanto all'altro, nell'altro uno sull'altro) ma non c'è da spaventarsi; si tratta di una timida proposta del pacchetto, avente comunque scarsa importanza. Già qui, comunque, è possibile fare delle modifiche, come spostamento e ridimensionamento dei campi; ma è inutile perdersi più di un poco di tempo.

Un file di report, anche qui con le diverse finestre di definizione.



Una fase di definizione di uno screen e in particolare quella di uno spinner.

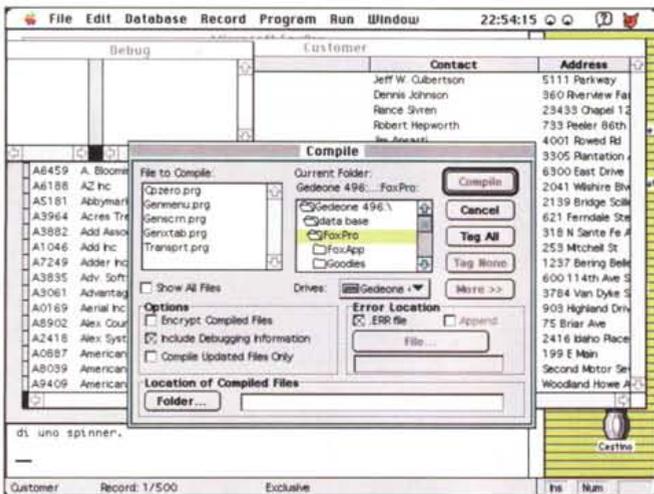
È arrivato il momento di inserire i dati; possiamo battere a mano i dati o importarli da un file esterno. Sempre per rimanere nel campo delle top model, una telefonata alla nostra amica Pat Cleveland ci fa avere, portato direttamente da lei, un file che importiamo direttamente nella nostra lista. Siamo così in possesso di qualche centinaio di indirizzi e numeri telefonici (non telefonatemi, per carità, sono incorruttibile) ancora sotto forma di lista, né più né meno di come lo si vedrebbe in un foglio Excel. Possiamo cominciare a lavorare sul serio.

Al lavoro con i formati

Sviluppato tutto il viaggio del Guided tour, si ha già bene in mente la struttura d'impostazione del pacchetto. All'inizio, come abbiamo già diverse volte detto ci si ritrova un po' in difficoltà con la tecnica di caricare un file e poi di

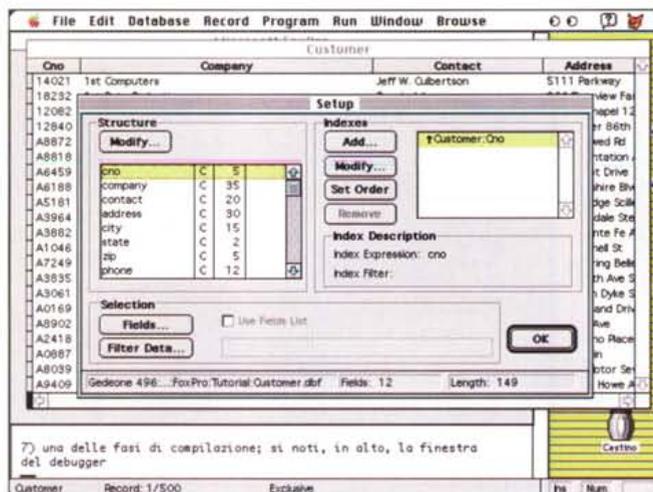
chiamare la modalità d'uso che gli si vuole applicare; ma è roba da niente di fronte al vantaggio di poter scegliere e tagliare di persona la tecnica d'uso del db; in poche parole è come avere a disposizione un unico motore a cui applicare carrozzeria, gommatura, cambio diversi a seconda dell'uso che dobbiamo farne.

FoxPro parte dal principio, adottato anche nei più nuovi linguaggi di programmazione, come Future Basic, PG-Pro, ProGraph o CPX che un'applicazione non è una struttura monolitica, ma è rappresentata, almeno durante la sua fase di costruzione, da un «progetto»; i pezzi relativi (database, report, maschere di input, ecc.) sono costruiti singolarmente e solo successivamente assemblati, secondo i più recenti dettami della programmazione OO. La disponibilità di lavoro su tool separati permette di costruire anche applicazioni self-tailored per diverse classi d'utenza, senza



Una delle fasi di compilazione; si noti, in alto, la finestra del debugger.

Una fase di setup della struttura di un progetto.



una delle fasi di compilazione; si noti, in alto, la finestra del debugger

per questo nascondere dietro il gioco delle password e dei diritti d'accesso la maggiore o minore «maneggiabilità» del file stesso; ovviamente anche con un occhio alla snellezza dell'applicazione stessa (ogni formato non occupa più di qualche k).

Non è certo possibile, in un articolo di qualche pagina, riassumere tutte le possibilità d'intervento o le diverse fasi di realizzazione degli specifici formati; vi possiamo assicurare che le possibilità sono tante e tali che, anche dopo settimane di autoistruzione si ritroverà sempre qualche perla nascosta utilissima (e questo senza ancora scomodare il linguaggio).

In ogni caso i formati fondamentali sono tre o quattro in tutto; sarà poi l'esperienza e la curiosità a farci scoprirgli altri.

Iniziamo dagli screen; essi sono la finestra, è il caso di dirlo, che ci permette di dare uno sguardo ai nostri dati. Guardatelo come una maschera, una

griglia, una chiave di lettura o come meglio credete; fatto sta che senza uno screen i dati sono inaccessibili (a meno di non voler usare il grezzo formato tabella).

A voler essere pignoli, uno screen è una vera e propria tabella; esso è conservato in un file con suffisso [.SCX], e la tabella ha un campo associato di memo con un'estensione [.SCT] (ecco la grandine, pochi chicchi, ma buoni). Ma cari macintoshisti, non storcete il naso, ci si fa l'abitudine a tutto; pensate a quei poveretti che combattono con le estensioni tutto il giorno. Il file SCX contiene informazioni relative alla definizione della finestra, alla sua posizione e dimensione, alle caratteristiche dei campi contenuti in essa e degli eventuali bottoni, ancora informazioni sull'ambiente di utilizzo e sulle relazioni intercorrenti tra lo schermo e il progetto globale.

Dal menu «File» si sceglie «New» e si chiede di creare uno schermo; si

apre una finestra che possiede, di lato a sinistra, una serie di comandi che permettono di definire le parti della maschera stessa: campi, aree grafiche e di editing, bottoni di vario tipo, campi di spinner (un edit field associato a due cursori opposti a freccia che permettono di graduare, secondo un intervallo customizzabile, i valori nel campo stesso), capi di tipo pop-up e liste di scelta. Al momento della definizione ognuna di queste parti (e tutte le altre che non abbiamo nominato) sono associabili con una variabile, o, nel caso di bottoni, con uno script. In questo caso l'ambiente permette di associare automaticamente agli oggetti clausole, come WHEN, VALID, IF e così via. Radio button e check box sono definibili in numero illimitato, e associabili a script anche subordinandone la selezione ad altri bottoni (tipicamente un push_button).

Se lo screen è costruito senza aprire una tabella dati, viene lasciata ampia libertà di realizzazione; se invece, alle spalle è già aperta una Table_DBF (table_data base file) il programma immagina che si stia lavorando già su campi definiti e coordina l'uno e l'altro ambiente rendendo già, ad esempio, disponibili i campi (addirittura è possibile inserire una parte della finestra browse nello schermo). La cosa interessante è che sebbene si lavori sempre in grafica, viene immediatamente creato, passo passo, il codice sorgente in linguaggio, e diventa spesso utile e divertente esplorarlo per «vedere come è fatto».

La seconda tappa importante è quella dei menu. Il lavoro svolto è tanto qualitativamente valido da essere paragonabile, per facilità, all'uso dello stesso tool di PG-Pro. In pratica disponiamo di una finestrella in cui si definiscono le voci di barra e discendenti; ogni voce poi può essere «agganciata» a un comando, a un submenu, a una procedura, a una macro. Il pannello dispone di una opzione, «Try it», che permette di simulare, a schermo, il lavoro man mano realizzato e di testarne l'efficienza.

A questo punto appare logico mettere insieme screen e menu per coordinarli e galvanizzarne l'efficienza. In pratica abbiamo fatto già un gran passo nella creazione di un'applicazione completa; a questa operazione e, soprattutto all'ottimizzazione di questa competenza è dedicato un ampio capitolo della Developer's Guide.

La creazione di un progetto e il suo sviluppo

Abbiamo già detto che la vera caratteristica di FoxPro è quella di lavorare su progetti. In effetti un progetto ha

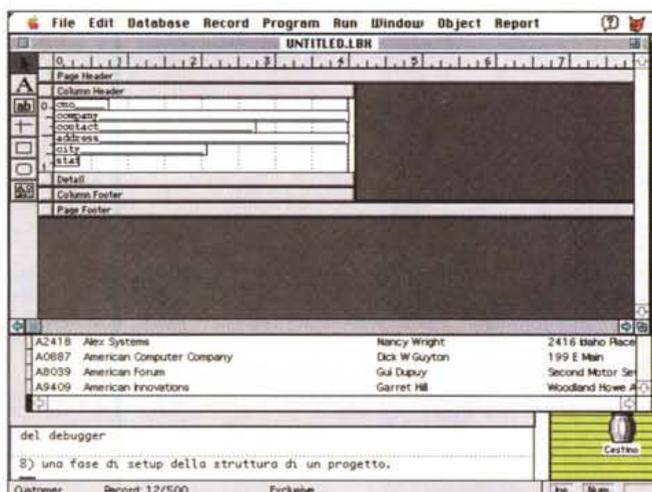
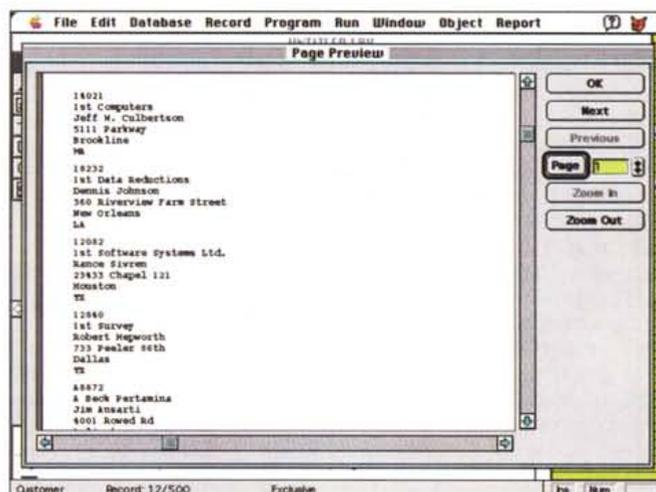
molteplici vantaggi concreti, capaci di far risparmiare tempo e sudore al progettista dell'applicazione. Quando un progetto viene realizzato e assemblato viene tenuta una completa «mappa» di tutte le pieces del programma; FoxPro automaticamente ricorda e individua tutti i componenti del progetto stesso menu, fogli di report, schemi di ricerca, etichette, librerie, ecc.

Da un progetto si può accedere a tutta la componentistica utilizzando procedure globali, organizzare la componentistica in maniera razionale (ad esempio creando cartelle ognuna dedicata a tipi diversi di categorie; poiché il progetto ricorderà la locazione di ogni componente, non ci sono problemi di path). Poiché un progetto, inoltre, permette di avere componenti dislocati in cartelle diverse, è consentito accedere anche a componentistica «lontana», magari dedicata ad altri scopi, e scritta e preassemblata in un linguaggio diverso. Giusto per fare un esempio, è possibile creare un comando per accedere al pannello di controllo, cambiarne le regolazioni o le scelte, e tornare all'applicazione principale; poiché tutto questo avviene senza attivazione di finestre, l'operazione è pressoché istantanea. Un progetto conserva il suo codice oggetto incapsulato nel suo «memo field»; questo permette di evitare, tra l'altro la frammentazione dello spazio sul disco e il conseguente ritardo operativo.

Riguardo all'operatività del programma oggetto, occorre tenere a mente che FoxPro tiene adeguata traccia di tutte le variazioni apportate al codice sorgente. In altri termini, se si esegue una sola, anche piccola, variazione ad un solo formato, tutto il progetto viene ricompilato e ne viene quindi garantita l'assoluta funzionalità. Infine, quando si genera un'applicazione da un progetto, tutta la componentistica è assemblata in un'unica applicazione (.APP, o file eseguibile), rendendo, manco a dirlo, la distribuzione senz'altro più pratica.

Microsoft Graph

Anche in FoxPro Microsoft ha inserito, graziosamente, Microsoft Graph, un piccolo add-in destinato a produrre grafica commerciale. Si tratta di un pacchetto credo già abbastanza noto per doverne parlare qui; come è noto esso funziona solo dall'interno di un'applicazione Microsoft (Word, Excel o, appunto, Fox), ed è basato su un datasheet (una specie di foglio elettronico contenente i dati, permette di costruire grafici secondo i formati classici già visti in Excel). Buona la risoluzione e le opzioni grafiche (ivi compreso il complesso orientamento in 3D) ma si tratta, come ovvio, di un add-in che può essere gradito ma non toglie o aggiunge nulla alla valutazione generale del package.



La costruzione di un foglio di etichette, in dettaglio e in preview.

Giusto per semplificare le cose, si possono creare progetti multipli, e stabilire fin dall'inizio che un progetto sarà portabile; in questo caso il pacchetto offre, come utility aggiuntiva, la possibilità di creare una cartella speciale, «Home»; tutta la componentistica po-

trà essere nidificata in questa cartella che sarà riconosciuta, al momento della compilazione, come una struttura unica, e quindi assemblata immediatamente. Il nome Home, sebbene vincolante al momento della realizzazione del progetto, non lo sarà più dopo la compilazione, e l'applicazione distribuita potrà essere sistemata dove meglio aggraderà all'utente finale.

Altre utility di FoxPro

FoxPro possiede un potente debugger sia a livello di compilazione che di runtime. Ovviamente, in questo secondo caso, non è possibile garantire l'eliminazione di tutti gli errori attraverso il debugger, ma finestre dedicate, come DEBUG e TRACE possono aiutare moltissimo; come pure sono molto utili i monitor di ambiente, di memoria e di file (ad esempio è possibile verificare

passo passo la gestione di file anche in un network).

SQL Select è un potente strumento, utilizzabile sia in ambiente di sviluppo che di debug. Attraverso di esso è possibile «agganciare» pezzi del programma (liste, bottoni, variabili) e controllarne le funzioni a richiesta. Ancora con ReportWriter molte funzioni specifiche di questo ambiente sono già bell'e pronte. Esiste la possibilità di creare e gestire array mono e bidimensionali, pubbliche e private, recuperare da esse valori e dati da passare a funzioni definite dall'utente, e così via. Per utenti ben avanzati, FP offre una serie di comandi e funzioni per operazioni di I/O a basso livello (si può accedere a file di qualunque tipo e lavorare su di essi,

tranne, stranamente file FoxPro). L'OLE (Object Linking and Embedding), specifico dei sistemi operativi Apple e Windows, ha qui facility di accesso molto potenti e facili da gestire; è inoltre possibile costruire help personalizzati, linkabili nell'oggetto. E ancora costruire documentazioni complete per le applicazioni attraverso l'applicazione FoxDoc.

FoxApp è l'applicazione che permette di costruire, perdonate il gioco di parole, applicazioni. FoxApp è abbastanza intelligente, visto che controlla la componentistica del progetto cercandone le parti principali perché possa funzionare (un'applicazione minima richiede una tabella, uno schermo, una palette e almeno un menu).

Il linguaggio di FoxPro

Un manuale linguaggio di mille pagine, del peso di un chilo e mezzo e delle dimensioni di un elenco telefonico formato B4 la dicono lunga sulla potenza dell'idioma di FP. Oltre settecento comandi fanno paura, ma proprio paura. Fortunatamente, almeno all'inizio, non è necessario imparare tutto; la maggior parte di un'applicazione, certo non molto complessa, sarà costruita direttamente dalle window, e se proprio è necessario manipolare qualche procedura o qualche bottone con uno straccio di programmino, i comandi e le istruzioni più usuali e semplici funzionano e «suonano» come quelle dei più noti linguaggi. Microsoft, con una scelta co-

Cosa offre la concorrenza

A livello di FoxPro ci sono solo tre altri pacchetti, capaci di offrire le stesse prestazioni; 4Dimension, Omnis 7 ed Helix Express. Del primo package (giunto alla versione 3 International, equivalente alla 5 Italiana), prodotto dalla ACI, abbiamo già parlato diffusamente in un articolo qualche mese fa. Si tratta di un pacchetto per così dire «strano»; sembra essere stato creato da un bambino capriccioso e geniale, che ha alternato picchi elevati di intuitività e preziosismi raffinati a idiosincrasie e strane ostinazioni difficili da comprendere. Fin dalla sua comparsa è stato oggetto di attente cure per la protezione della copiatura; il principio di base del produttore è che nulla viene dato gratis; anche il più piccolo di più occorre pagarlo. E così sono fioriti una serie di pacchetti aggiuntivi tra cui un wp, uno spreadsheet e un package di grafica (cari e di prestazioni inferiori a quelli offerti dalla concorrenza), un compilatore (non facciamoci confondere dal-

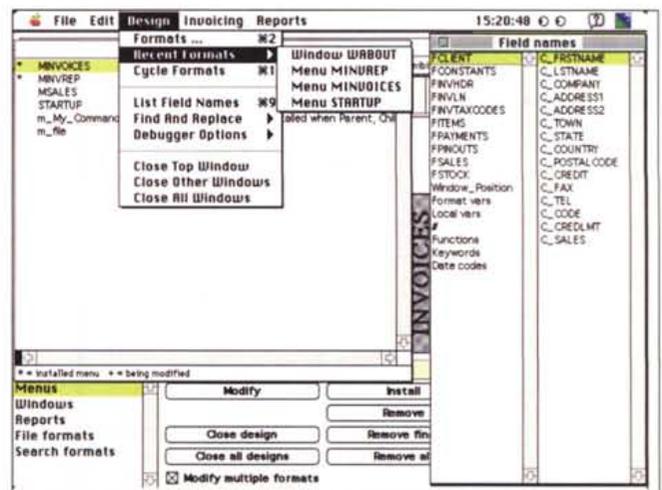
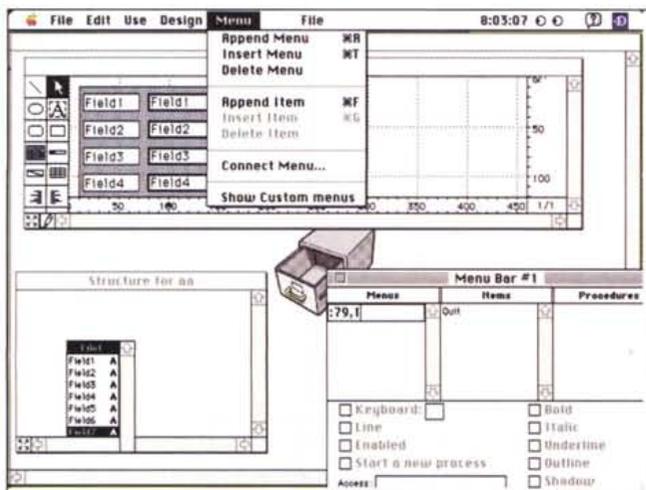
la parola, visto che non crea un'applicazione stand-alone, ma solo ottimizza certi particolari calcoli e routine), e varie altre cosucce. Mentre fino alla penultima versione il pacchetto serviva anche il multiutente, la nuova ha separato i due campi e il client server viene venduto a parte a suon di bigliettoni.

4D è probabilmente il database, di questa classe, più venduto; è il più agevole da imparare, ma soffre di una grave lentezza nelle operazioni di ordinamento e di ricerca. È disponibile un runtime per creare applicazioni stand-alone.

Omnis 7, giunto alla versione 3.0, è il più aristocratico dei quattro. Prodotto e commercializzato dalla Blyth Software è costosissimo, e dispone, come FoxPro, della analoga versione DOS. Non è protetto, possiede una buona documentazione, e dispone di un'interfaccia molto simile a quella di 4D, anche se più potente ed elastica. Non è più importato da tempo in Italia e purtrop-

po la stessa casa non assiste amorevolmente gli sfortunati clienti (me compreso) che possiedono un vecchio package e desiderano upgraderlo. Forse a causa di questa scadente assistenza il pacchetto è piano piano sparito dai cataloghi dei grandi distributori, come Mac Connection, MacWarehouse e MacCenter. E tutto questo è un peccato, visto che si tratta di un prodotto eccellente, di poderose prestazioni, dotato di una documentazione ineccepibile e ben articolata.

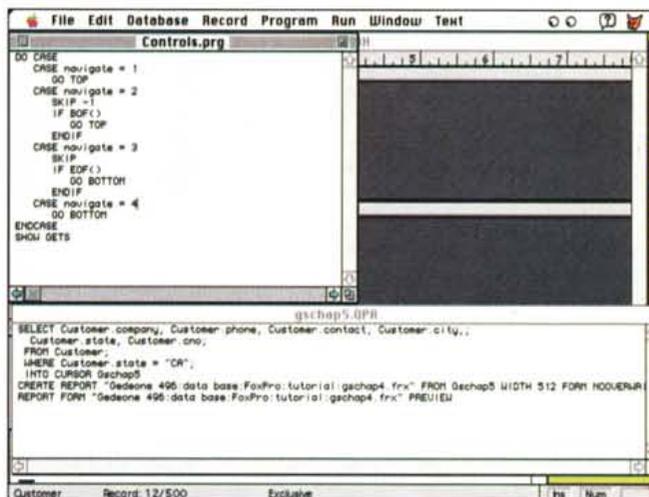
Helix Express (versione 2.0) è la naturale evoluzione, durata dieci anni, di uno dei più vecchi database per Mac. Prodotto dalla Helix Technologies, Illinois, è un package curiosissimo; spinge al massimo la concezione grafica Mac, essendo dotato di una interfaccia del tutto grafica. Un'applicazione è qui chiamata «collezione», una raccolta, appunto di icone, che contengono formati, campi, abachi, report, liste, tutte definite attraverso, e solo attraverso simboli e comandi gra-



raggiata, ha abolito e/o cambiato radicalmente molti comandi presenti nel vecchio FoxBASE+ for Mac, rendendoli più logici e maneggiabili, ma onestamente la potenza e la complessità del linguaggio è tale da fornire i migliori risultati solo in mano a gente dal polso fermo. Il manuale, comunque, è ineccepibile; ogni comando o funzione è illustrato nella sua sintassi e in una esemplificazione ben realizzata e collegata con riferimenti ad altri comandi. Gli esempi sono sia realizzati come linee singole di programma, sia, in molti casi, come vere e proprie pieces di sorgente, utilizzabili tal quali, o con minimi aggiustamenti, in numerose occasioni in programmi custom.

Accanto alle istruzioni e ai comandi, il

Due esempi di script (program) completamente editabili attraverso il text editor incorporato.



fici. La cosa all'inizio è davvero sconcertante (anche gli stessi script sono fatti da icone, incatenate l'una all'altra da legami consequenziali o logici somiglianti ad autostrade piastrellate), ma ci si fa subito la mano alla intrinseca comodità di questo ambiente abbastanza veloce, molto ben strutturato, e dotato di utility curiose e gradevoli (ad esempio esiste un comando, Why?, che indica perché l'ultima operazione corrente sia stata rifiutata, o abbia generato un errore).

Sebbene non distribuito in Italia, Helix Express gode di un servizio di supporto internazionale molto efficiente e gentile (una mia interlocutrice si sforzava di parlare un passabile italiano per chiarire i termini di una risposta); gli upgrade sono poco costosi (dalla nascita ne ho ricevuti almeno cinque o sei) e i miglioramenti, nelle successive edizioni sono sempre stati costanti e notevoli, segno di un'elevata cura prestata al prodotto. Helix addirittura ha una linea diretta per i

suggerimenti, che, se riconosciuti validi, vengono apprezzati e ricompensati con un piccolo bonus. Non possiede, e questa è una piccola pecca, runtime, mentre viene fornita col pacchetto una versione ridotta del client server, che funziona alla perfezione anche se solo per due utenti. È dotato di una collezione di esempi molto articolata ed efficace, da cui poter trarre spunti e suggerimenti per il lavoro giornaliero.

Tutti e tre i package offrono, a pagamento, con costi che vanno dal centinaio di dollari fino alle molte centinaia di lire, l'abbonamento a una rivista dedicata; generalmente si tratta di articoli molto specialistici, poco utili per l'utente normale. Viceversa, negli USA si trovano diversi tutorial più semplici e piacevoli dei manuali, talvolta criptici.

Tre bellissimi package, quindi, che contengono degnamente lo scettro a Fox; chissà che non sia il caso di riparlarne in un prossimo futuro.

linguaggio offre una cinquantina di funzioni di sistema, molto utili. Indipendentemente comunque dall'uso di esse in FP, c'è da ricordare che è la prima volta che vediamo elencate tutte queste variabili con il loro numero, e l'elenco potrebbe essere molto utile a chi programma in un linguaggio tradizionale.

Conclusioni

FoxPro ha tante qualità e le abbiamo tutte descritte; ma ci siamo tenuta la migliore per queste conclusioni; è di gran lunga il più veloce db dell'area Mac (ad esclusione di Panorama, la cui velocità è però legata al fatto che tutto il file è mantenuto in RAM). Database relazionale dell'ultima generazione, dotato del suo equivalente in DOS (con cui può scambiare file e buona parte del codice), dotato di una documentazione, sia software che cartacea, superba, è comunque destinato a un professionista, capace di tirar fuori la grinta da questo package così raffinato.

Molto più veloce di 4D in molte occasioni, può paragonarsi per prestazioni a Omnis 7 (dal costo però stratosferico); non è facile da usare a livello avanzato, mentre, per usi più modesti, è più utile ricorrere a mezzi meno potenti, come appunto Panorama o FileMaker. Supporta pienamente AppleEvents e, ovviamente, il Publish-Subscribe. Richiede una certa «assuefazione», specialmente per chi è abituato allo standard corrente delle applicazioni Mac, ma offre il livello più avanzato oggi raggiungibile nel campo dei database (e non parlo solo dell'area personal). In altri termini è un pezzo super-raffinato, una Lamborghini o una Bugatti, per intenderci; per chi è abituato a rispettare i limiti di velocità, meglio comprare una Punto o una Twingo!

I tre ambienti a confronto, nell'ordine 4D, Omnis 7 ed Helix Express; si noti l'originalità e la particolarità dell'ultimo, complicato solo all'apparenza.

