

# Microsoft Visual Fox Pro 3.0

di Francesco Petroni

Fox Pro è, come noto, un prodotto della Microsoft, frutto della recente acquisizione della omonima software house. Il prodotto appartiene alla categoria DBMS, si poggia sulla sintassi xBase, utilizza in altre parole come base di partenza le classiche istruzioni dBase. Sta per uscire questa nuova versione, la 3.0, che sostituisce la 2.6. Nel suo nome è stata aggiunta la fondamentale parola Visual.

Il primo interrogativo che ci si pone è il seguente: come può un prodotto basato sulla sintassi xBase, che si concretizza in una serie di istruzioni che nel dBase interprete si potevano digitare dall'ambiente puntino, diventare Visuale, ovvero un prodotto totalmente Object Oriented, Event Driven, e in cui i vari membri dell'applicazione si realizzano operando in specifici ambienti visuali, dove si disegnano oggetti e se ne definiscono proprietà e metodi, anziché scrivendo «a memoria» righe di codice in un ambiente Editor?

Ci si pongono subito dopo un paio di altri interrogativi.

Il primo è quale siano, oggi, i prodotti concorrenti del Fox Pro, ovvero gli altri prodotti xBase, di pari categoria, proposti dalle altre case software. Il secondo è quale sia la strategia della Microsoft nel campo dei DBMS, in cui ha già inserito ACCESS, diffusissimo anche per il fatto che è presente nel Bundle Office, e in cui entra anche Visual Basic, che pur non essendo un DBMS, ma un linguaggio di programmazione «qualsiasi», è ben dotato in termini di funzionalità di accesso ai Database, e lo è in-

differentemente ai formati DBF, xBase, MDB di Access, DB di Paradox, e, tramite ODBC, a tutti gli altri esistenti sulla faccia della terra. In particolare Fox Pro 3.0 dispone del nuovo Driver ODBC 2.0 a 32 bit, ottimizzato rispetto alle precedenti versioni.

A parte la concorrenza costituita dagli altri prodotti della Microsoft, i rivali di Visual Fox Pro 3.0 sono sostanzialmente due: il dBase V della Borland, ora totalmente Windowsizzato, e l'Object Vision della Computer Associates, per Windows, erede del glorioso Clipper, che rimane tutt'ora il prodotto più utilizzato per lo sviluppo di applicazioni gestionali, a interfaccia carattere, che debbano girare su PC.

Le risposte a questi interrogativi le potremo dare nei prossimi mesi via via che i vari prodotti vedranno definitivamente la luce e li potremo provare e confrontare nelle loro versioni finali.

## Il target del nuovo Visual Fox Pro

Si tratta di un prodotto destinato dichiaratamente agli sviluppatori. Per vari motivi. Il primo è che con Fox Pro si sviluppano applicazioni gestionali complesse affrontabili, soprattutto dal punto di vista relazionale, solo da specialisti della materia. Esistono altri prodotti destinati ad utilizzatori non professionali, e comunque in grado di «reggere» applicazioni di complessità medio-bassa. Inoltre è già un prodotto a 32 bit (lo stiamo provando su Windows NT Workstation) adatto allo sviluppo di applicazioni Client-Server, anzi dotato di specifici stru-

menti per sviluppare tale tipo di applicazioni, in genere molto «ricchio» o come si dice in gergo, «mission critical», il cui fallimento quindi avrebbe gravi conseguenze economiche.

Il secondo motivo è che adotta la tecnologia RAD (Rapid Application Development) che si concretizza in una serie di strumenti operativi e di funzionalità che permettono di costruire rapidamente oggetti e classi di oggetti, immediatamente riutilizzabili in altre parti dell'applicazione o in altre applicazioni.

Non è questo il momento per descrivere nel dettaglio le caratteristiche del modello Object Oriented offerte dal nuovo Fox Pro (lo faremo in sede di prova), vi anticipiamo il fatto che permette di costruire Oggetti e Classi di Oggetti (schemi di un oggetto), operando sia in modalità comandi che in modalità Visuale (esiste lo specifico ambiente Visual Class Designer). Permette di gestire l'ereditarietà tra un oggetto padre e un oggetto figlio, in modo che modificando qualche impostazione del padre ne risentano, se lo si desidera, anche tutti i figli presenti nell'applicazione. È previsto l'incapsulamento di proprietà e di comportamenti negli oggetti ed è previsto il polimorfismo, ovvero che un oggetto abbia uno scheletro di base e porzioni di codice specializzate in funzione di certi parametri assegnati all'oggetto stesso.

La programmazione può essere del tutto guidata dagli eventi (Event Driven) il che significa che sono definibili ed intercettabili tutti gli eventi generati dalla tastiera, dal mouse (ovviamente anche il Drag and Drop), dall'applicazione e dai suoi componenti, ecc.

Altre caratteristiche tecniche evolute sono l'adozione incondizionata della tecnologia OLE Automation che permette di inserire nell'applicazione Fox, e di gestire via applicazione, un oggetto realizzato con qualsiasi mini OLE Server disponibile nel sistema.

Visual Fox Pro conserva la possibilità di lavorare, grazie alla Window Command, anche dall'ambiente puntino. È chiaro che esistono centinaia di comandi in più e che alcuni vecchi comandi sono stati conservati solo per garantire la compatibilità con le versioni precedenti.

Infatti le applicazioni «vecchie» funzionano direttamente anche con la nuova versione. È chiaro che sarebbe opportuno, ed esistono funzionalità che aiutano questo passaggio, una loro modifica in senso Visuale.

Le prestazioni sono rimaste, malgrado l'appesantimento dell'ambiente dovuto alla nuova strumentazione, analoghe a quelle,

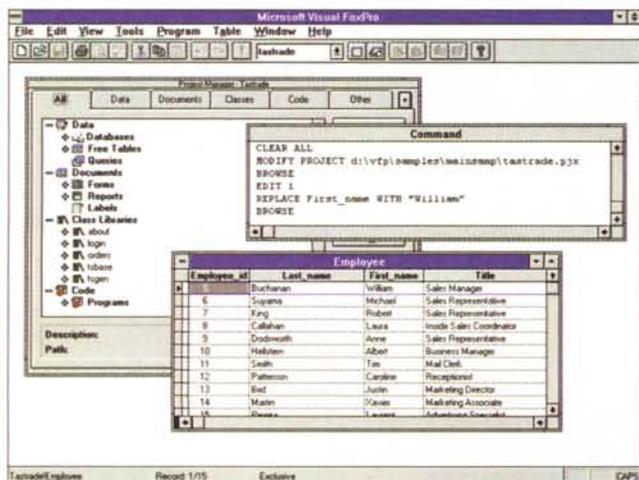


Figura 1 - Microsoft Visual Fox Pro 3.0 - Finestra Project Manager, Finestra Comandi e Browser di una Tabella. Fox Pro conserva la possibilità, malgrado la totale conversione alle regole della Object Orientation, di lavorare nell'ambiente puntino (grazie alla Command Window), cosa gradita agli sviluppatori professionali che spesso preferiscono intervenire direttamente digitando i comandi. I comandi disponibili, rispetto alle versioni precedenti, sono molti di più, anche in relazione al fatto che coprono le necessità delle nuove tipologie di oggetti. Molti dei vecchi comandi sono stati conservati, in

alcuni casi solo per garantire la necessaria compatibilità con le versioni precedenti.

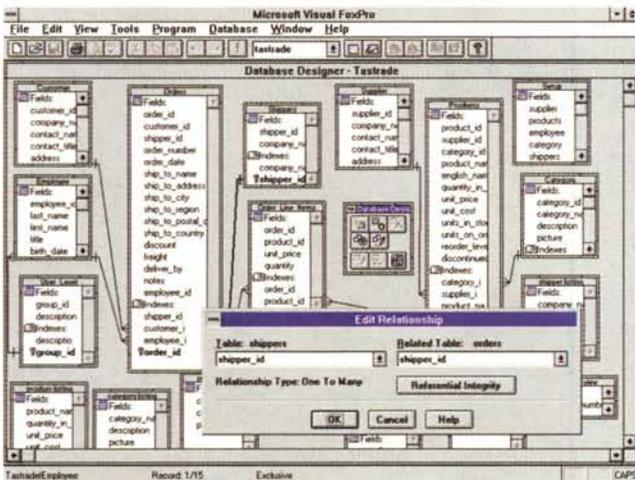
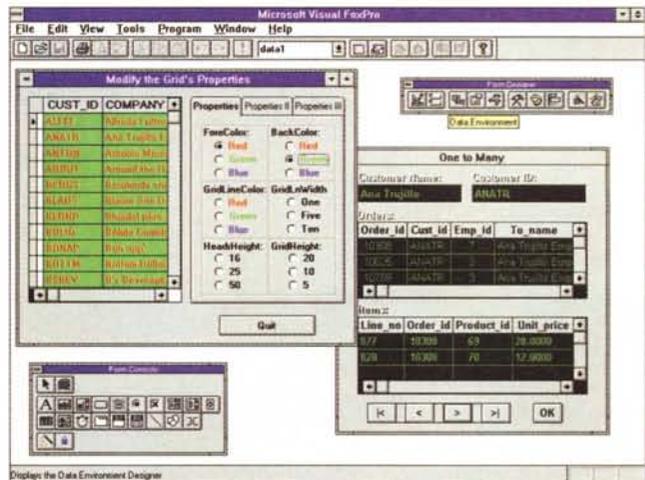


Figura 2 - Microsoft Visual Fox Pro 3.0 - Database Designer.

Pur essendo basato sulla classica sintassi xBase, che prevede l'impostazione «manuale» degli indici e delle relazioni, Fox Pro permette un controllo ben più sofisticato degli aspetti relazionali. Si può usare questo ambiente Database Designer per impostare le relazioni tra le tabelle e le eventuali regole di integrità referenziale. Le relazioni che possono anche diventare «persistenti» e quindi attive in ogni parte dell'applicazione.

Figura 3 - Microsoft Visual Fox Pro 3.0 - Disegno diretto delle Form.

La parte del leone, quando si parla di un prodotto Object Oriented, è riservata alle Forms e agli strumenti per generarle, sia perché costituiscono il «front end» verso il mitico utente finale, sia perché possono inglobare buona parte delle funzioni di validazione dei dati immessi. Una Form contiene Control, ce ne sono di nuovissimi interessantissimi (la Grid, che è facilmente riempibile di dati, specie con quelli di una tabella «lato molti» in una relazione uno a molti, la Pagina con le linguette, ottima per organizzare grossi volumi, ecc.). Possono essere generate, Forms e Controls, sfruttando qualche decina di Wizard e di Builder.



ottime grazie all'adozione della tecnologia Rushmore, delle versioni 2.5 e 2.6.

## Un primo giro

Per scrivere questa anteprima abbiamo installato i dieci dischetti della versione Beta (ma nella versione definitiva, prevista per la prossima primavera, saranno di più) su Windows NT Workstation, su cui, tra le altre cose, abbiamo installato anche un Excel 32 bit per NT per provare OLE Automation. Per curiosità abbiamo «girato» Fox Pro con successo anche da Windows per Workgroup, sfruttando il Win32S, «motorino» che permette alle applicazioni pienamente a 32 bit, di girare anche con Windows 16 bit.

Avendo avuto, per questa anteprima, il limite tassativo delle due facciate, il che significa quattro figure, abbiamo configurato Windows alla risoluzione 800 per 600 in modo da catturare videate più ricche (in ogni videata mostriamo più cose), anche se poi in definitiva le stesse risultano poco «fotogeniche».

Girandoci intorno abbiamo potuto dare uno sguardo alle novità più evidenti. La prima è il Project Manager, che fa da contenitore di tutti i «pezzi» dell'applicazione. La finestra mostra le Tab, ovvero le linguette, per selezionare le famiglie di elementi, e gli elenchi mostrati appaiono informa strutturata (fig. 1).

Ebbene linguette, organizzazione strutturata delle risorse dell'applicazione, cornice della finestra, sono tutte caratteristiche tipiche di Windows '95.

Il nuovo modello Object Oriented mette a disposizione ambienti e comandi per generare Classi. Nella figura 2 vediamo il Designer delle Classi, in cui è in lavorazione

una classica classe costituita da una pulsantiera VCR like, immediatamente riutilizzabile in tutte le Form di manipolazione dati. Accanto alla finestra Designer vediamo anche la ricca Toolbar, con alcuni strumenti nuovi, rispetto ad esempio a quelli di Access e/o Visual Basic e la classica, e anche questa, ricchissima Finestra delle Properties.

Proseguendo il giro ci imbattiamo nel nuovo Database Designer (fig. 2). C'è poco da fare! Il miglior modo per capire un'applicazione che gestisca un complesso database relazionale è costituito dalla vista grafica e schematica delle tabelle e delle relazioni. Le Relazioni si possono definire e possono diventare persistenti, il che vuol dire che sono attive in ogni punto dell'applicazione senza che lo sviluppatore si debba ogni volta ricordare di aprire tabelle, indici e relazioni.

Da esplorare sono gli ambienti visuali in cui si generano Forms, Report e Label. Ora è possibile definire immediatamente, con uno speciale Wizard, una Form con SubForm (fig. 3). La SubForm è costituita da un oggetto Griglia. Interessante è la possibilità di impostare «runtime», quindi in fase di esecuzione del programma, alcune caratteristiche delle singole celle della griglia, magari condizionandole al valore che contengono.

I Controls più complessi dispongono di specifici Wizard o Builders.

## Client Server

In un'applicazione Client, Server Fox Pro svolge il compito di front end, verso l'utente finale, dei dati che risiedono sul Server, in qualsiasi formato raggiungibile tramite il driver ODBC.

Fox Pro dispone di una serie di funziona-

lità, che rientrano nel Visual Fox Pro 3.0 Connectivity Kit, che permettono varie soluzioni tecniche a seconda che si voglia privilegiare il lavoro sul Server, oppure quello sul Client, oppure si preferisca minimizzare il «traffico» sulla linea.

Comunque lo strumento più interessante è costituito dall'Upsizing Wizard, che consente di praticare il «prototyping» dell'applicazione. In pratica si realizza sul Client, la si testa con pochi dati significativi, la si fa sperimentare dall'utente, che ne verifica tutti gli aspetti operativi. Una volta che tutto è OK il Wizard si incarica di costruire, a partire dal modellino, la struttura del Database sul Server, ad esempio in formato MS Sql Server, oppure Oracle, e tutte le connessioni che servono per vedere questo nuovo database dall'applicazione Client.

Molti nuovi strumenti favoriscono questo approccio Client Server. Esistono, ad esempio, le Viste e le Query. Le seconde differiscono dalle prime in quanto consentono l'aggiornamento dei dati anche nel caso questi provengano da basi dati remote.

Per concludere puntualizziamo due aspetti che dovrebbero essere emersi da tutto quanto detto. Il primo è che Visual Fox Pro 3.0 consente tre tipi di utilizzo. Dal puntino, con i nuovi comandi e con lo sfruttamento della possibilità, propria della Object Orientation, della impostazione via comando delle proprietà degli oggetti (con la classica sintassi object.property=x). Con i vari Designer, ambienti di disegno con ricche barre di strumenti e con ricche finestre delle proprietà, e con la possibilità di associare codice ai vari eventi. Infine con i vari, sono tantissimi, Wizard e Builder, che, se l'applicazione è abbastanza tradizionale, dovrebbero riuscire a coprire la quasi totalità del fabbisogno di oggetti e codice. MS