

I nuovi controlli di Windows 95 visti da MS Visual Basic 4.0

Vogliamo analizzare, un po' più nel dettaglio di quanto non si faccia nelle tradizionali prove dei nuovi prodotti, i nuovi oggetti, o «controls», presenti in Windows 95, vedendoli sia dal punto di vista di Windows 95, lo stesso dal quale li vede un suo normale utilizzatore, sia dal punto di vista di un tecnico del software, dal quale li vede ad esempio un programmatore in ambiente Windows, che utilizzi un qualsiasi linguaggio di sviluppo specifico per Win 95, come il ben noto Visual Basic. In particolare il Visual Basic 4.0, ancora in versione beta, essendo «double face», in quanto permette di programmare sia per il vecchio ambiente a 16 bit che per il nuovo a 32 bit, già dispone dei nuovi Controls, alcuni dei quali, ovviamente, possono essere utilizzati solo lavorando sulla versione 32, valida, lo ricordiamo, sia per Windows 95 sia per le versioni 3.51 dei due Windows NT, il Server e il Workstation

di Francesco Petroni

I nuovi oggetti di cui ci occuperemo sono una dozzina e, come può constatare chi passa ad utilizzare Windows 95, modificano sensibilmente il modo di lavorare con Windows. Di questo non ci accorgeremo subito (Windows 95 è appena arrivato) ma tra qualche mese, quando le varie case avranno presentato le nuove versioni dei loro prodotti per Windows 95, che conterranno sicuramente in dosi massicci i nuovi oggetti.

Come detto i nuovi oggetti (useremo sia il termine «oggetto» che il termine, più corretto dal punto di vista tecnico, di «controllo», o in inglese «control») sono una dozzina e la nostra intenzione è quella di descriverli con un certo dettaglio, mostrando anche dei piccoli esempi di applicazione in VB4. Prima però facciamo due o tre premesse, necessarie per inquadrare meglio il fenomeno.

Una nuova generazione di controlli

La prima cosa da dire è che i nuovi Controlli sono, in generale, più complessi.

Questo dal punto di vista del programmatore significa che è più lunga la fase di definizione delle loro caratteristiche iniziali, ma meno complicata la loro gestione da programma, in quanto si tratta pur sempre di gestire un unico controllo.

Come esempio vedremo la Lista a Colonne collegata ad una Tabella di Da-

tabase (che Visual Basic 4.0 chiama sinteticamente DBGrid) che è in grado di ospitare una lista di dati, ad esempio i record di una Tabella o di una Query di

un database, e di permetterne la completa gestione. Tutto in un solo controllo e impostando inizialmente solo quattro o cinque proprietà dell'oggetto.

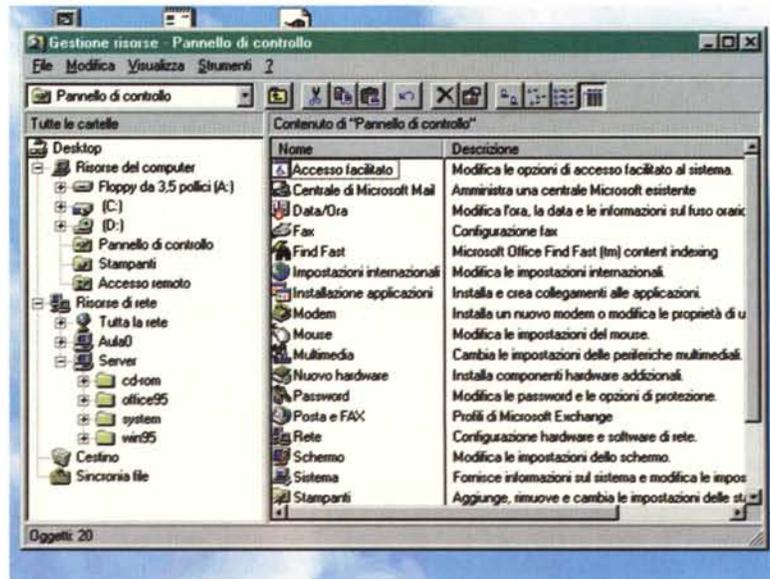


Figura 1 - MS Windows 95 - Gestione delle Risorse del Computer. In questo programma di servizio di Windows 95 vediamo ben quattro o cinque «oggetti» tipici di Windows 95 cui corrispondono altrettanti «controlli» in Visual Basic 4.0. In alto la ben nota Toolbar, in basso la ben nota riga di stato, a sinistra la lista di tipo Outline e a destra una Lista a Colonne, che può essere trasformata in una Lista ad Icone. I primi due controlli, come vedremo tra un po', in una applicazione VB4, vanno ad occupare automaticamente le due posizioni in alto e in basso nella finestra, comunque questa sia dimensionata.

In figura 1 vediamo l'applicazione di Windows 95 Gestione delle Risorse (sarebbe il famoso Explorer che sostituirebbe, grosso modo, il precedente File Manager) in cui sono concentrati ben quattro o cinque dei nuovi Controls di Windows 95. La parte a destra della videata è proprio una Lista a Colonne, quella a sinistra una Lista Outline (vedremo anche questa), in alto c'è una Toolbar di Windows 95 e quindi spostabile (anzi quella dell'Explorer fa eccezione) e in basso una Status Bar, che c'era anche prima ma ora diventa obbligatoria.

Poi, nella figura 2, vediamo, in fase di impostazione delle proprietà, i tre nuovi controlli (DBList, DBCombo e DBGrid) presenti in Visual Basic 4.0 e collegabili direttamente a dati di un Database attraverso un Data Control, che fa da intermediario.

Nella figura 3 invece vediamo solo una DBGrid e una MessageBox inserita per dimostrare il fatto che, pur trattandosi di un solo oggetto, è possibile individuarne i singoli componenti, cioè le singole celle.

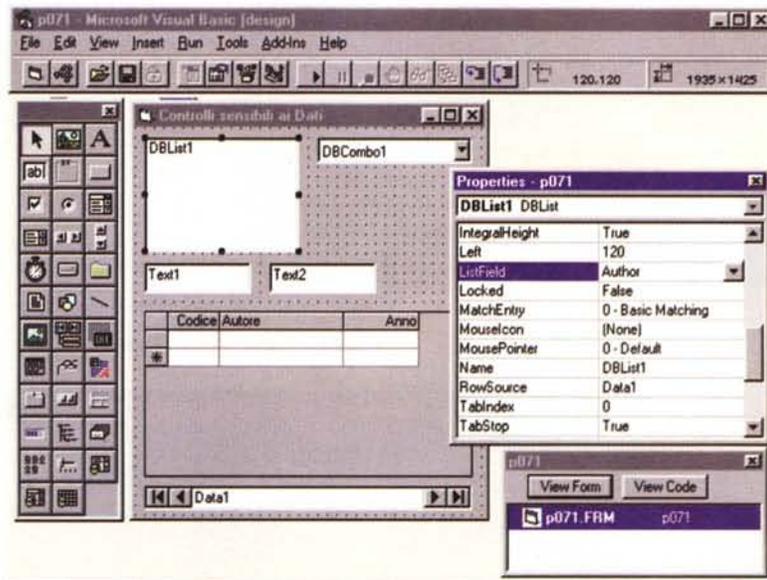


Figura 2 - MS Visual Basic 4.0 - Controlli collegabili ad un RecordSet. In Visual Basic 4.0 esistono tre nuovi controlli che si possono riempire direttamente con i dati provenienti da un RecordSet, ovvero da una Tabella di un Database oppure da una Query, definite in un Data Control. Sono la Lista semplice, la Combo e la Lista a Colonne, che può mostrare più campi di uno stesso RecordSet.

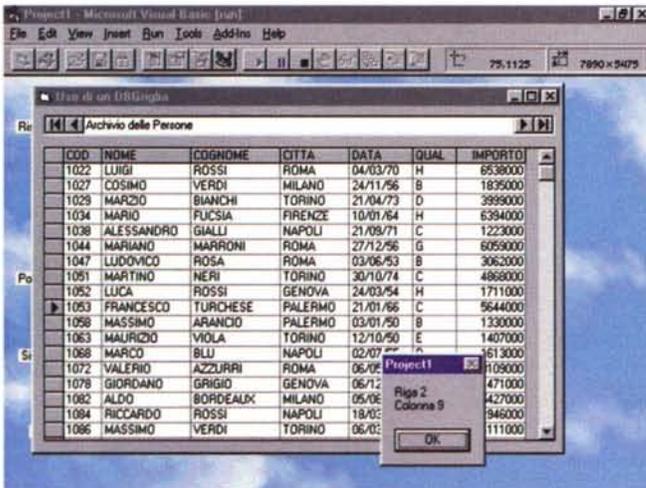


Figura 3 - MS Visual Basic 4.0 - La DBGrid. Io sono, perdonate se parlo di fatti personali, un po' fissato con il concetto di Form Multitable, che, in un'applicazione di tipo gestionale, mostra in una unica Maschera dati provenienti da più Tabelle del Database. Se, di una Tabella, debbono essere mostrati più record, questi possono essere inseriti in un unico controllo di tipo DBGrid, che ne permette anche la gestione. È più semplice la fase di definizione (direi di caricamento) dell'oggetto, mentre la sua gestione è comunque complessa se, come probabile, si vogliono gestire direttamente dalla Tabella inserimenti, cancellazioni, modifica di record.

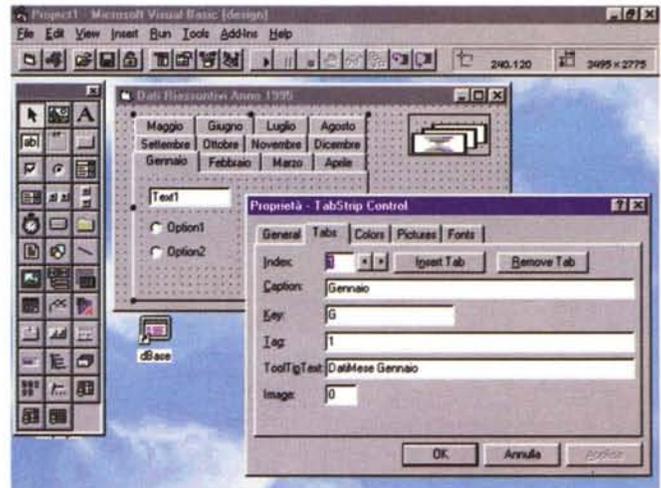


Figura 4 - MS Visual Basic 4.0 - Impostazione delle proprietà in controllo che ha... figli.

I nuovi controlli introdotti con Windows 95 e disponibili in Visual Basic 4.0 sono molto complessi. Questo significa che sono definibili per ciascuno di questi numerose proprietà. Inoltre molti controlli dispongono di serie di proprietà che si ripetono per i propri N componenti. Ad esempio la Form Tabbed attiva una serie di proprietà per ciascuna linguetta, oppure la Lista a Colonne una serie di proprietà per ciascuna colonna. Visual Basic 4.0 mette, molto opportunamente, a disposizione una ulteriore finestra nella quale impostare le proprietà in serie.

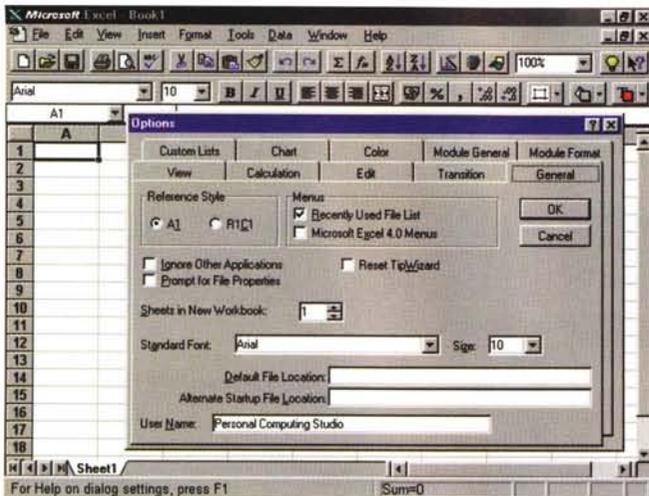


Figura 5 - MS Excel 95 - Impostazione delle Opzioni. Mostriamo come le finestre Tabbed, ovvero le finestre a pagine linguettate, in realtà fossero già state «inventate» prima che Windows 95 le promuovesse ad oggetti di sistema. La casa pioniera è stata la Borland che ha messo per prima le linguette al suo Quattro Pro, poi la Lotus con il suo Notes, che archivia i documenti in Registri linguettati, infine la Microsoft che ha introdotto Box a Linguette, necessarie in quei casi, come quello costituito dalle impostazioni delle opzioni di Excel, in cui le impostazioni da inserire siano tantissime ed organizzabili per argomento.

La DBGrid dispone di una serie di proprietà impostabili nella fase «design» oppure nella fase «runtime». Altre caratteristiche, ad esempio il ridimensionamento di una colonna, possono essere impostate direttamente sull'oggetto durante l'esecuzione del programma.

L'applicazione più immediata di un controllo di questo tipo è come Subform in una Form Multitable, in cui appaiono i dati provenienti da due tabelle di un Database, poste in relazione Uno a Molti tra di loro.

I controlli... con i figli

Un'altra novità negli oggetti di Windows 95 è costituita dalla presenza di controlli complessi, in cui una serie di caratteristiche si ripetono per una serie di volte. Come esempio proponiamo la già citata Lista a Colonne, in cui possono essere impostate N colonne e per ciascuna di esse una serie di caratteristiche, oppure la Form a «pagine», identificate da una linguetta, nella quale ogni pagina ha le sue caratteristiche e i suoi oggetti, che «vivono» solo all'interno della singola pagina (il control si chiama Tabbed). In figura 4 vediamo una Form con una serie di Pagine (una per ciascun mese dell'anno) e la finestra delle caratteristiche che permette di definire le caratteristiche di ciascuna pagina. In questa finestra c'è un contatore che indica di quale pagina si tratta e due pulsanti Inserisci Pagina o Cancella Pagina.

È evidente che in caso di «controllo con figli» la sintassi del generico comando VB si arricchisce di un ulteriore elemento che serve per indicare quale sia la pagina, ad esempio:

FORM1.FRAME1.PAGE1.TEXT1=«Buongiorno»

Inoltre va detto che anche il numero delle colonne oppure il numero delle pagine sono caratteristiche dell'oggetto modificabili comunque «runtime». Si pensi ad esempio ad una finestra a pagine, in cui ogni pagina contenga i dati di un record (ovviamente in una situazione con pochi record).

Per quanto riguarda in particolare il controllo chiamato Tabbed facciamo alcune precisazioni. La prima è che il controllo Tabbed già esisteva come VBX aggiuntivo realizzato da produttori indipendenti per la versione precedente di Visual Basic (la 3.0 Professional). Tanto è vero che in Visual Basic 4.0 ce ne sono ben due e che uno di questi è stato acquisito proprio da un produttore indipendente. La differenza più macroscopica

Figura 6 - MS Visual Basic 4.0 - Programma che definisce al volo un Oggetto Tabbed. Esistono due differenti tipi di Control Tabbed. La differenza più importante è che in uno (SSTab) gli oggetti posti su una delle pagine sono attivi solo su quella particolare pagina mentre nell'altro (TabStip) gli oggetti sono attivi su tutte le pagine. È chiaro che è il programmatore che poi, da programma, attiva e disattiva a seconda delle necessità gli oggetti che vuole.

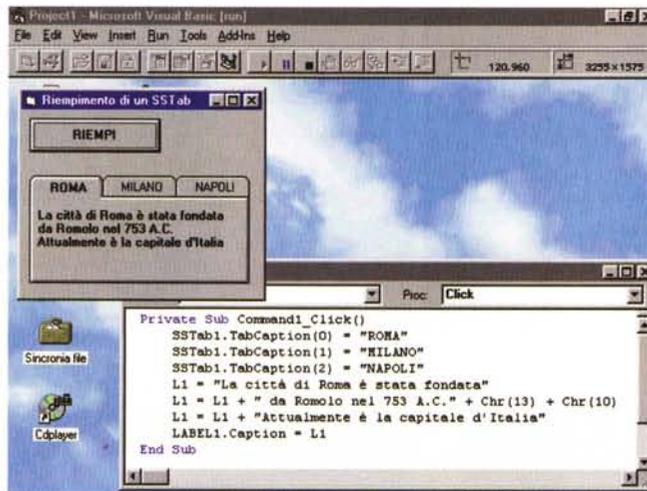
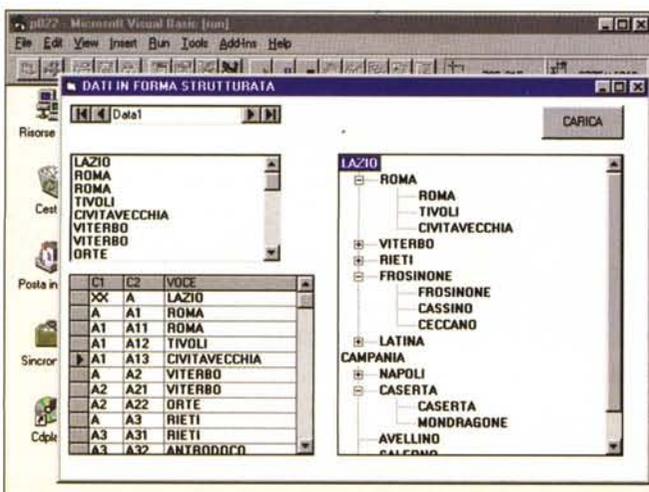


Figura 7 - MS Visual Basic 4.0 - Vista di una Lista in modo TreeView. In Visual Basic 4.0 ci sono ben due controlli di tipo Outline (sono un po' fissato anche con gli Outline). C'è il vecchio, diciamo così, MSOUTLINE.VBX, e c'è questo nuovo controllo (il suo nome proprio è TreeView) più adatto ad utilizzazioni di tipo Database, in cui la gerarchia che regola i vari elementi è direttamente indicata in specifici campi. In basso a sinistra vediamo come è fatta la Tabella. La colonna C2 indica il nome dell'Item mentre nella C1 c'è il nome dell'Item gerarchicamente di livello superiore. Ad



esempio Civitavecchia (A13) è in provincia di Roma (A1), che a sua volta è una provincia del Lazio (A).

pica tra i due sta nel fatto che un oggetto semplice, ad esempio una TextBox, nel controllo che si chiama SStab vive solo in una delle sue pagine, nell'altro, che si chiama TabStrip, vive in tutte le sue pagine.

L'altra considerazione è di tipo filosofico. L'organizzazione di un'applicazione cambia radicalmente a seconda che si decida di utilizzare il controllo Tabbed o meno. Si pensi ad un'applicazione in cui ci sia una Maschera che debba mostrare parecchi dati. La soluzione può essere quella di realizzare più Form specializzate (soluzione alla Windows 3x) oppure un'unica Form a pagine linguettate (soluzione alla Windows 95). E questa è una decisione che deve essere presa praticamente all'inizio del lavoro di sviluppo.

In figura 5 mostriamo come le finestre Tabbed in realtà fossero già state «inventate» prima che Windows 95 le promuovesse ad oggetti di sistema.

In figura 6 un microprogramma Visual Basic 4.0 che mostra come alimentare «runtime» una serie di Pagine di un Control Tabbed.

Liste... di tutti i tipi

Come detto numerosi oggetti di Windows 95 e conseguentemente numerosi nuovi controlli di Visual Basic 4.0 sono destinati a mostrare, nelle più svariate forme, insiemi di dati. Nella precedente versione di Visual Basic c'erano le classiche List Box, Combo Box e la classica Griglia, che occorreva in un certo senso caricare da programma con i dati desiderati scrivendo apposite routine. In VB4 troviamo, oltre ai precedenti controlli, le nuove versioni «sensibili ai dati» (DBList, DBCombo e DBGrid). Il che significa che il collegamento con i dati, ad esempio alla Tabella di un Database, va più semplicemente definito come caratteristica dell'oggetto e non da programma.

Una lista più particolare è quella che si chiama Outline, che serve per mostrare liste di dati organizzati gerarchicamente. Il controllo, presente sotto forma di VBX aggiuntiva nella versione precedente di Visual Basic, è rimasto ma è stato affiancato da un altro più adatto per applicazioni Database (si chiama TreeView) in quanto il livello gerarchico di ciascun Item può essere letto direttamente in un campo dalla Tabella di origine (predisposto per tale scopo). A questa facilità di caricamento si affianca la possibilità, che in questa sede non esploriamo, di modificare l'aspetto della lista in modo che appaiano simboli grafici che indichino il livello

Figura 8 - MS Visual Basic 4.0 - Vista di una Lista in modo TreeView - Listato.

Questo è il programma che alimenta la lista TreeView, che si chiama TV. I dati sono forniti dal Data Control Data1 e in particolare dai tre campi che, come detto e come visto nella figura precedente, si chiamano C1, C2 e VOCE. C'è da dire che si tratta di un oggetto complesso per cui le istruzioni che permettono di gestirlo (nel nostro caso lo abbiamo semplicemente caricato di dati) non sono semplicissime.

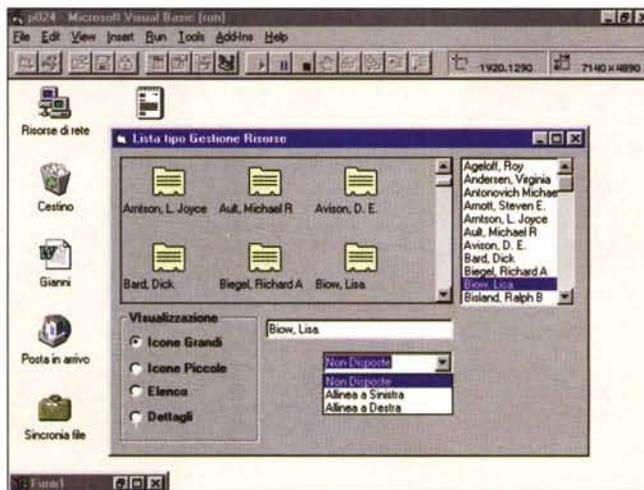
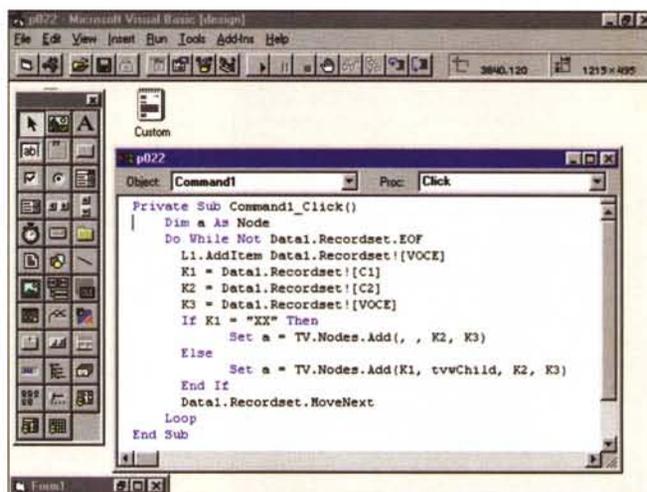


Figura 9 - MS Visual Basic 4.0 - Costruzione di una Lista a Icone.

Ecco un ulteriore tipo di Lista, anche questa disponibile ormai a livello di sistema al punto che costituisce la modalità di visualizzazione del contenuto delle cartelle presenti sulla scrivania di Windows 95. In questo caso ad ogni elemento della lista può essere associata una Icona grande o piccola, oltre che la scritta grande o piccola. Si tratta ovviamente di un controllo di tipo complesso, che necessita di una serie di comandi per impostarne le parti testuali e quelle grafiche. Sono inoltre individuati tutti gli eventi che possono avvenire a livello di singolo Item della lista.

Figura 10 - MS Visual Basic 4.0 - Costruzione di una Lista di Immagini. In Visual Basic 4.0 esiste un nuovo Controllo, di tipo invisibile (che scompare durante l'esecuzione del programma), che serve per fornire serie di figure BitMap a chi ne fa richiesta. Nella figura ne vediamo la definizione iniziale, attraverso la specifica Finestra di Dialogo, cui segue una fase di inizializzazione per il collegamento tra la lista e l'oggetto che la richiede. Ad esempio controlli serviti dalla Lista di Icone possono essere il Tabbed, il ListView.





Figura 11 - MS Visual Basic 4.0 - Zoom sul Toolbox. In Visual Basic 4.0 è stato, come noto, abbandonato il concetto di VBX (file corrispondenti a librerie con Controls aggiuntivi specificamente realizzati per VB), ed è stato sostituito da quello, più generalizzato, di OCX. Il componente aggiuntivo è utilizzabile da qualsiasi linguaggio compatibile VBA e si comporta come piccola applicazione OLE (i componenti OCX si chiamano anche MiniOleServer). Qui vediamo uno zoom 2 per della ToolBox di Visual Basic 4.0 caricata con una serie di OCX in più rispetto a quelli standard. O perché non caricati dalla procedura standard di installazione o perché aggiunti da altri applicativi installati sul sistema.

dell'Item e il fatto che abbia o non abbia degli Item sottostanti.

Del nuovo Control TreeView vediamo un'applicazione significativa nelle due figure 7 e 8.

Mi sembra il caso di sottolineare ancora una volta la potenza dello strumento Outline (che corrisponde alla parte sinistra dell'applicazione Gestione delle Risorse di figura 1) che può essere utilizzato come strumento per la navigazione, potente ed intuitiva, alla ricerca dei dati desiderati.

Altro tipo di Lista è la ListView che può mostrare il proprio contenuto in una delle quattro forme previste nelle Cartelle di Windows 95: Icone grandi,

Icone piccole, Lista o Tabella (figura 9). La ListView si può caricare, al pari delle altre, con dati provenienti da un Database. Il problema più complesso è quello di associare ad ogni Item una Icona. Visual Basic 4.0 lo risolve utilizzando un altro Controllo, di tipo invisibile (che scompare durante l'esecuzione del programma) e che serve per fornire serie di figure Bitmap a chi ne fa richiesta (in figura 10 ne vediamo la definizione attraverso la specifica Finestra di Dialogo).

In definitiva quello che ci interessa è sottolineare che il concetto di Lista è un concetto generico, e che, indipendentemente dal tipo di lista (Lista semplice, Combo, Lista DB, Griglia, Lista ad Icone, Lista ad Albero), esistono una serie di caratteristiche generali valide per tutti i tipi ed altre variabili a seconda del tipo specifico di oggetto.

Un buon esercizio, parliamo dal punto di vista didattico, è quello di caricare,

con lo stesso insieme di dati, più liste di tipo differenti.

Scopriamo i nuovi oggetti in Windows 95 e i nuovi Controls in VB4

Ricapitolando: in Windows 95 ci sono una serie di nuovi oggetti di sistema, alcuni già presenti nelle precedenti versioni ed opportunamente aggiornati, altri del tutto nuovi. Visual Basic 4.0, in quanto linguaggio di programmazione per l'ambiente Windows, tiene conto di tutti questi oggetti e li mette a disposizione del programmatore sotto forma di Controls. Vediamo in figura 11 una vista ingrandita del ricchissimo Toolbox in cui possiamo notare:

Controls preesistenti non modificati

Controls preesistenti modificati per Windows 95

Controls del tutto nuovi

Figura 12 - MS Visual Basic 4.0 - Slider e Progress Bar. Ecco due controllini di piccolo cabotaggio, lo Slider, che è una specie di ScrollBar più elegante, e la Progress Bar che serve per dare informazioni sulla durata di un certo processo in corso. Il nostro programmino lega l'avanzamento dello Slider e della Progress Bar ad un timer. Abbiamo legato la Progress Bar anche all'evento Change dello Slider.

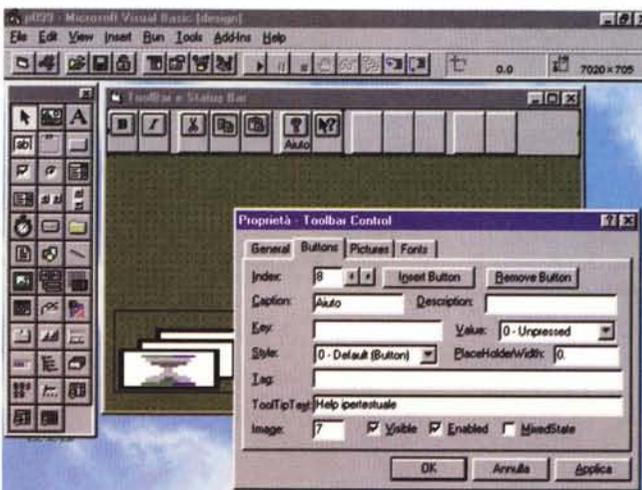
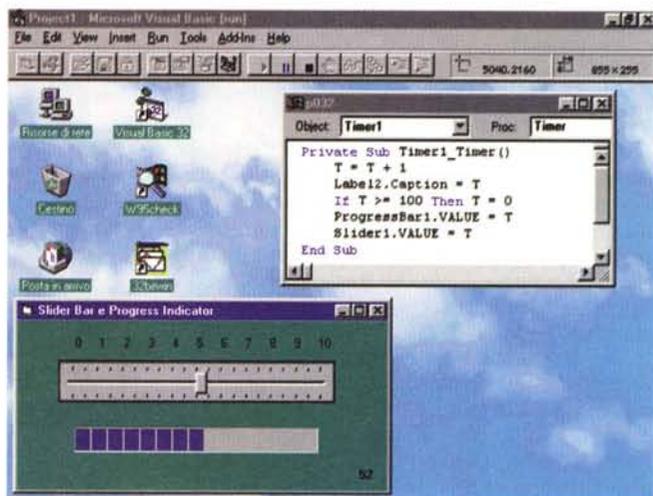


Figura 13 - MS Visual Basic 4.0 - Toolbar e StatusBar.

Ecco altri due Controls indispensabili in un'applicazione per Windows 95 che voglia essere degna di questo nome. Si tratta di Controls complessi in quanto hanno ambedue dei figli (una Toolbar può contenere più indicazioni, che variano durante l'esecuzione del programma a seconda della situazione). Conseguentemente sia le proprietà che le istruzioni per gestire gli oggetti in questione sono moltissime.

In questo articolo stiamo approfondendo principalmente questi ultimi.

Visto che ci siamo vediamo alcuni altri che non hanno nulla a che fare con le Liste e con i Database.

Cominciamo con due Controls di piccolo cabotaggio, lo Slider e la Progress Bar (in figura 12). Il primo è una versione più aggiornata della classica Scroll-Bar. Vanno indicate le due proprietà Valore Minimo e Valore Massimo e va indicato cosa succede al variare della posizione del suo cursore, che si può «agganciare» con il mouse e trascinare avanti ed indietro. Il secondo serve per indicare la percentuale di avanzamento di un processo in corso.

Quest'ultimo controllo dispone di una proprietà che rappresenta il valore numerico di tale percentuale. Per sperimentarlo abbiamo fatto due cose. Abbiamo definito un Timer e un Evento Timer, che ogni x secondi fa avanzare l'indicatore. Poi abbiamo legato la Progress Bar allo Slider in modo da poterla riportare indietro. Insomma con due o tre semplici istruzioni si gestiscono i due oggetti.

Altri due controlli, fondamentali in qualsiasi applicazione, sono la Toolbar e lo StatusBar. Si tratta di oggetti con «figli», per usare un'espressione citata prima.

Per la Toolbar (figura 13) vanno indicati i vari pulsanti, il cui aspetto può essere prelevato da una Lista di Immagini, analoga a quella vista in precedenza in figura 10, e gli eventi conseguenti ai click sui vari pulsanti. Ciascun pulsante può avere più aspetti a seconda della situazione, ad esempio Acceso, Spento o Disattivo.

Per il controllo Barra di Stato vanno indicate inizialmente le varie sezioni e le loro caratteristiche. Poi, in fase di esecuzione del programma, ci saranno istruzioni che serviranno ad attivare, nelle varie sezioni, Scritte o Icone che indicano lo stato dell'applicazione. È tutto quello che ci si aspetta che ci sia.

Ecco la Casella di Testo con il testo Formattato

Il controllo standard per visualizzare un singolo dato è la TextBox che può contenere una stringa lunga 255 caratteri, oppure un testo lungo fino a 65.535 caratteri, caricabile ad esempio da un file di tipo testuale.

Il controllo TextBox dispone di una serie di proprietà di tipo estetico, che servono per definire l'allineamento, il tipo carattere e le sue caratteristiche, di tutta la casella e quindi di tutto il testo. Inoltre può visualizzare il suo testo su



Figura 14 - MS Visual Basic 4.0 - Casella con un testo in formato RTF.

In Visual Basic 4.0 c'è il nuovo controllo Casella di Testo Formattato, in cui può entrare un testo formattato secondo le specifiche del formato Rich Text Format (RTF). Questo può essere prodotto anche dal Word Pad di Windows 95 che, come si vede in secondo piano, è in grado di salvare in tale formato. Questo controllo permette di realizzare applicazioni, anche di tipo Database, in cui vengono visualizzati dei documenti esteticamente organizzati.

Figura 15 - MS Visual Basic 4.0 - Specifiche del formato RTF. In questa finestra vediamo come appare un testo salvato in formato RTF letto... da DOS (scusate... è l'abitudine) ovvero letto come sequenza di caratteri. Il Control RichTextBox dispone di funzioni interne che permettono, ad esempio, di selezionare parti di testo, poi da programma si può inserire qualche codice, proprio delle specifiche RTF, nel testo. In altre parole una Box di RichText può servire non solo per visualizzare un testo formattato preesistente ma anche per realizzarlo, scrivendo il testo e inserendo in qualche maniera, ad esempio costruendo una Toolbar per tale scopo, i codici RTF (citati nell'Help di VB4).

più righe magari aiutando lo scorrimento nelle due direzioni con le due Scroll-Bar.

Il nuovo controllo RichTextBox permette di visualizzare un testo «arricchito» che include internamente quindi anche le più svariate formattazioni valide anche su porzioni di testo e non solo su tutta la casella che lo contiene (figure 14 e 15).

Un testo in formato RTF è facilmente producibile con qualsiasi Word Processor e addirittura con il WordPad di Windows 95.

Gli utilizzi di un controllo di questo genere sono facilmente immaginabili. In una Form possono essere inserite RichTextBox, che mostrano testi lunghi ed esteticamente organizzati, ovvero con titoli centrati, paragrafi variamente giu-

stificati e con vari tipi di rientri, elenchi puntati, ecc.

Si può ipotizzare, con un po' di programmazione in più, di usare una RichText Box per realizzare un mini Word Processor. In pratica in tale box si può scrivere normalmente un testo, si può selezionare parte del testo, e, ad esempio, usando una serie di pulsanti esterni (organizzabili in una Toolbar), assegnare un opportuno codice di formattazione a tale parte.

La conoscenza dei codici propri del formato RTF è facilitata dall'Help che li mostra tutti. Sono tantissimi, molto più di quanto ci si aspetti, ma non è necessario utilizzarli contemporaneamente tutti.

MS