

Borland Delphi 1.0

di Francesco Petroni

Chi deve realizzare applicazioni in ambiente Windows e deve individuare il tool di sviluppo più adatto alle sue necessità può scegliere numerose strade alternative che dipendono da vari fattori. Il primo fattore è costituito dal tipo di applicazione da realizzare, ad esempio una cosa è un'applicazione con finalità Grafiche una cosa è un'applicazione orientata al DataBase, il secondo fattore è costituito dalle sue conoscenze pregresse, in termini sia di linguaggi tradizionali sia di tool in ambiente Windows.

In particolare la categoria Tool in ambiente Windows è caratterizzata da un certo numero di prodotti cosiddetti Visuali, facili da usare e rapidi, nel senso che, in un tempo relativamente breve, si può realizzare il primo prototipo dell'applicazione già pronto per i primi test.

È nato anche il termine programmazione Visuale, il che significa che l'atti-

vità di sviluppo è più vicina a quella svolta da un disegnatore che non a quella svolta da un programmatore classico.

Oggi le due soluzioni estreme sono costituite da una parte dall'utilizzo del linguaggio C, in una delle sue numerose varianti (C, C++, Visual C, Borland o Microsoft o altri) e/o aggiunte, linguaggio che permette di realizzare qualsiasi tipo di applicazione, ma essendo molto «vicino alla macchina», è poco adatto per realizzare applicazioni gestionali. L'altro estremo è costituito dall'utilizzo del Visual Basic, della Microsoft, molto più semplice ed immediato, accessibile anche da parte di un programmatore mediamente preparato.

Il VB è abbastanza dotato, ma non troppo, in materia accesso ai dati. Il suo limite principale è che si tratta tuttora di un linguaggio interprete, per cui il prodotto finale, l'eseguibile, non potrà essere in nessun caso velocissimo.

Ci sono altre alternative intermedie, costituite da prodotti già orientati ai dati, come i vari Tools Gupta della Gupta, PowerBuilder della PowerSoft's, e, al limite, anche i vari Access della Microsoft, dBase V e Paradox della Borland e Object Vision della Computer Associates, che, pur non essendo dei linguaggi in senso tradizionale, possono essere utilizzati per sviluppare applicazioni gestionali.

Il nuovissimo Delphi della Borland si pone come obiettivo quello di essere facile da usare almeno quanto il Visual Basic, ma anche quello di essere molto più potente, in quanto compilatore a tutti gli effetti. Altra differenza sostanziale tra i due sta nel fatto che il Visual Basic, per la parte codice si poggia sulla sintassi del vecchio Basic, molto diffuso e conosciuto sin dai tempi degli 8088, mentre il Delphi usa il Pascal (per l'occasione in versione Object Pascal) meno diffuso (è molto usato in ambito di-

dattico), ma molto più evoluto. D'altro canto la Borland è tutt'ora legata, con i suoi «Turbo», al Pascal.

Una delle cose da scoprire sarà quale è la dose di Pascal nel Delphi, ovvero a che livello occorre conoscere tale linguaggio per poter sfruttare al meglio Delphi. Lo vedremo.

Una caratteristica particolarmente evoluta in Delphi è costituita dalla sezione accesso ai dati. Di Delphi ne esiste una versione Client/Server destinata a chi deve sviluppare applicazioni strategiche in cui i dati risiedono su un Server. Questa modalità di approccio nella realizzazione delle Applicazioni Informatiche si chiama Upsizing: si parte dal basso, dal PC, per arrivare a costruire Sistemi Informativi Aziendali distribuiti in Rete.

Nel Delphi normale ci sono una serie di Componenti specializzati nell'interfacciamento con i dati. E anche in questo Delphi sembrerebbe avere qualche carta in più rispetto al Visual Basic, che solo nella attuale versione 3.0 Professional, dispone di un discreto set di funzionalità di accesso ai Dati, peraltro ben inferiori a quelle di Delphi. Dovremo rifare il confronto a fine anno, quando dovrebbe essere uscito il Visual Basic 4.0 a 32 bit.

Borland Delphi: cosa dice la Borland

Delphi è uno strumento di sviluppo che sfrutta la tecnologia RAD, Rapid

Delphi per Windows

Produttore:
Borland - Francia
Distributore:
I.S.S.
Via Valera, 31/16
20020 Arese (MI)
Tel. 02/93582260

Prezzi (IVA esclusa):
Delphi per Windows
versione completa
aggiornamento

Lit. 949.000
Lit. 399.000

Application Development, il che si concretizza in una serie di caratteristiche tutte orientate alla riduzione dei tempi di sviluppo.

Le versioni di Delphi, come detto, sono due: Delphi e Delphi Client/Server. Quest'ultimo include, oltre a tutto il Delphi «normale», una serie di Driver SQL Links 2.5, per collegare l'applicazione a database Oracle, Sybase, MS SQL Server, Informix ed InterBase. Include il Visual Query Builder verso SQL, poi la versione SQL 2.5 del ReportSmith, ed altro. In pratica permette di sviluppare vere applicazioni Client/Server facilmente ed immediatamente adattabili a differenti tipi di Server Dati. In questa prova vedremo solo Delphi, la versione base, per cui non parleremo, se non per citarlo, di Client/Server.

Le caratteristiche di Delphi, dichiarate dalla Borland, e che verificheremo, quando possibile, nella prova, sono:

- compilatore velocissimo (350.000 linee di codice al minuto) e possibilità di ricompilazione del solo codice modificato,
- eseguibili EXE velocissimi, che non richiedono, se si vuole, librerie DLL alle quali accedere dinamicamente, e questo migliora le prestazioni (a scapito delle dimensioni dell'eseguibile),
- assembler integrato, per permettere eventuali rifiniture del codice,
- eseguibili esenti da royalty,
- Two Way Tools, ovvero la possibilità di sviluppare nell'ambiente Visuale, con automatica costruzione delle righe di codice, e la possibilità contraria di scrivere direttamente righe di codice e di ritrovarne gli effetti nelle viste grafiche,
- reale Object Orientation con possibilità di riutilizzo e di creazione di nuovi componenti,
- tutorial interattivo dedicato alle principali feature.

Borland Delphi: come è fatto

Allora, ricapitolando, Delphi è fatto di tre componenti principali: l'ambiente di programmazione Visuale (alla Visual Basic, per chi lo conosce), il «motore» DataBase, per il collegamento ai dati, e l'Object Pascal Language. Quest'ultimo non è completamente compatibile con

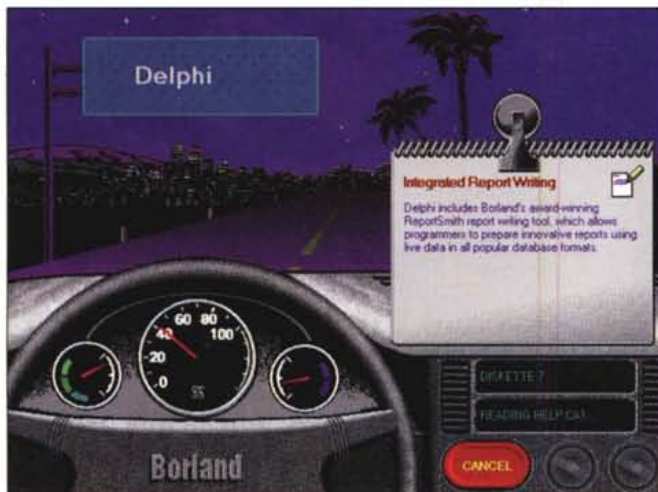
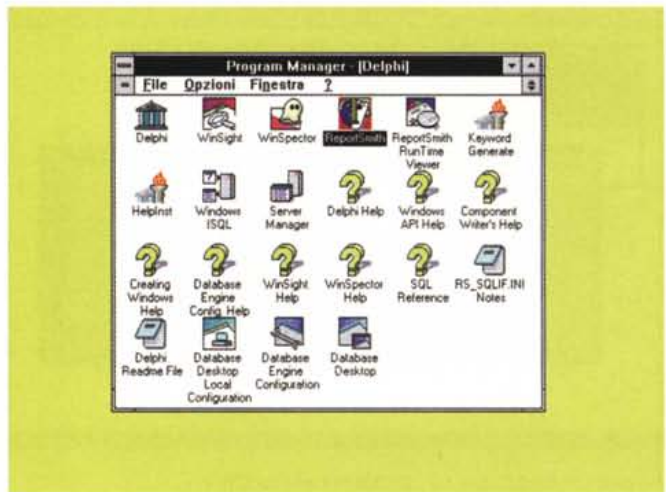


Figura 1 - Borland Delphi 1.0 - Installazione nella notte.

Borland Delphi arriva su un CD nel cui interno troviamo una directory Install che contiene le immagini dei quindici dischetti di installazione. Poi c'è una directory Manuals, poi un'altra con i dischi di installazione del Borland Database Engine e un'altra ancora con il ReportSmith (immagini di cinque dischi). Poi c'è la sezione Runimage che contiene gli eseguibili scompattati ed organizzati in sottodirectory. Infine la sezione Video, con due bei file AVI, di tipo pubblicitario.

Figura 2 - Borland Delphi 1.0 - Il Gruppo Delphi.

Questa immagine parla da sé. Oltre a Delphi possono essere installati una serie di programmi accessori, complementari rispetto a Delphi, una serie di procedure di installazione e di configurazione dei vari componenti, necessarie per stabilire i collegamenti con i dati, una serie di Help su svariati argomenti e alcuni documenti tecnici.



l'Object Pascal 7.0, per cui ci sono alcune sezioni dei manuali che indicano le differenze a beneficio di chi volesse riutilizzare preesistenti porzioni di codice.

Oltre al Delphi vero e proprio, il pacchetto contiene una dozzina di programmi ausiliari, di programmi per la configurazione, ecc. Di questa ricchezza ne dà esplicita testimonianza la videata del Gruppo Delphi (fig. 2) generata dalla procedura di installazione.

Anche il Delphi in sé è ricco, nel senso che, essendo organizzato in componenti, dispone di una dozzina di finestre

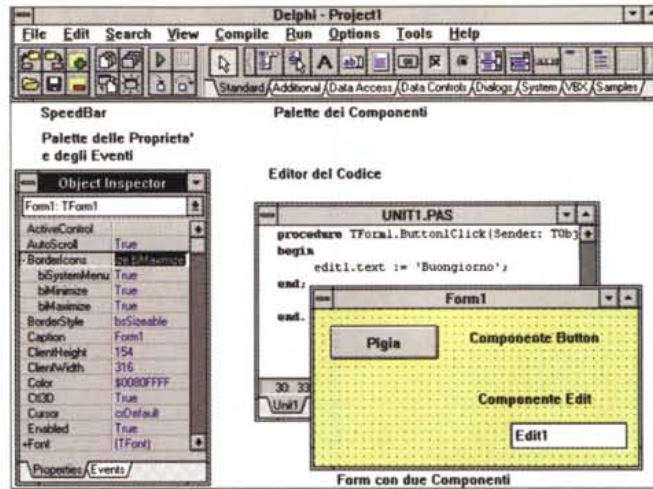


Figura 3 - Borland Delphi 1.0 - Ambiente. Vediamo una normale situazione di lavoro con aperte le finestre principali: la Form su cui si sta lavorando, la finestra per l'editazione del Codice, l'Object Inspector che è la finestrella che mostra le Proprietà e gli Eventi legati all'oggetto selezionato, nel nostro caso l'intera Form. In alto la zona comandi con due sezioni, quella di sinistra, che si chiama SpeedMenu, che contiene una serie di pulsanti d'uso generale, e quella di destra, che è la Palette, organizzata a linguette e quindi suddivisa per categorie, dei vari Controlli e Componenti usabili in un'applicazione.

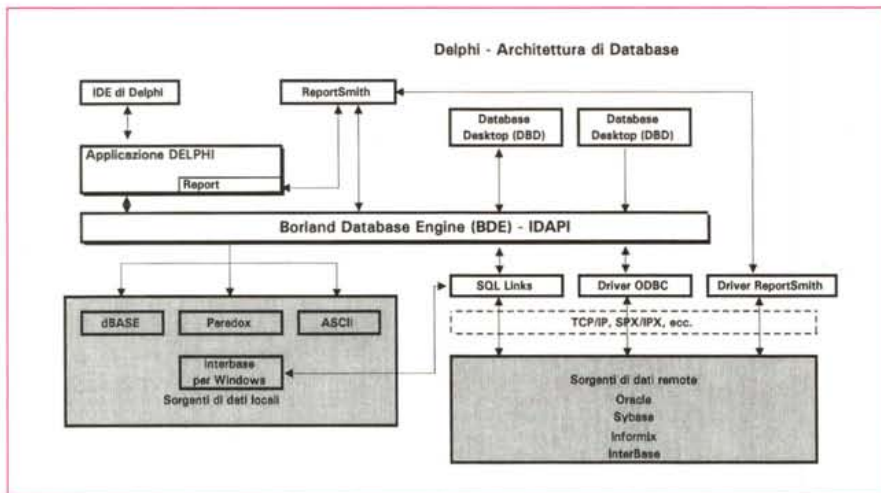


Figura 4 - Borland Delphi 1.0 - Architettura per l'accesso ai Dati. La specializzazione di Delphi è l'accesso ai dati, facilitato da una parte dall'abbondanza dei componenti specifici e dall'altra dall'esistenza di una sottostante architettura, appoggiata sul motore BDE, che permette l'accesso a qualsiasi tipo di fonte dei dati. Tale modalità è condivisa dagli altri prodotti della Borland di categoria DBMS, dal dBase V e dal Paradox.

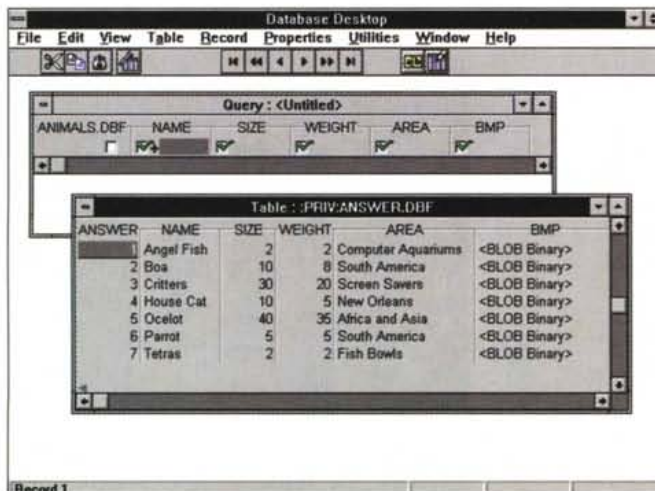


Figura 5 - Borland Delphi 1.0 - DataBase Desktop (DBD).

Tra i vari accessori di cui è dotato Delphi ce ne sono tanti che riguardano l'accesso dai Dati. Questo, che si chiama DataBase Desktop, è un programma che serve per costruire delle Query in un ambiente QbE analogo a quello presente nei prodotti Borland di tipo DBMS (ad esempio nel dBase V visto nel precedente numero di MC) e che poi scarica in formato «istruzione SQL» facilmente utilizzabile in un Oggetto TQuery dentro una Form Delphi.

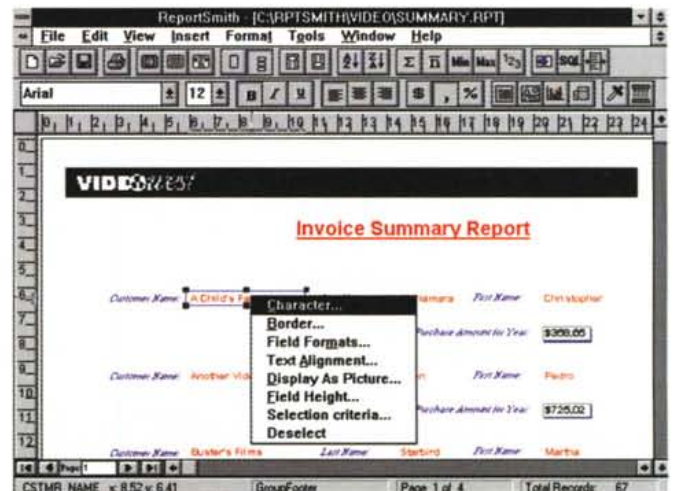


Figura 6 - Borland ReportSmith - Costruzione del Report.

Ormai ha preso piede la moda di far svolgere il compito di disegnare i report e di stamparli ad un programma a sé stante, che potrebbe, al limite, anche lavorare in modalità «stand alone», per conto suo insomma. Borland ha realizzato il ReportSmith, evolutissimo, sia perché dispone di un potente modulo per la costruzione di Query di accesso ai dati (anche in questo caso il linguaggio è SQL), sia perché permette tutti gli interventi di tipo estetico consentiti dall'ambiente Windows. Mi chiedo solo perché invece nel dBase V sia stato inserito il Crystal Report, che non è della Borland. Anche in fase di disegno del Report con ReportSmith si vedono direttamente i dati.

Figura 7 - Borland ReportSmith - SQL.

Siamo sempre in ReportSmith e siamo alle prese con una Query SQL che deve prelevare i dati da più tabelle a disposizione del Report. È evidente che ReportSmith è in grado di «farsi da sé» le interrogazioni, anche se queste sono complesse, alleggerendo di un bel po' di lavoro il programma che lo ospita. In alto vediamo la pulsantiera che mostra la sequenza logica delle operazioni da compiere quando si crea un nuovo report.

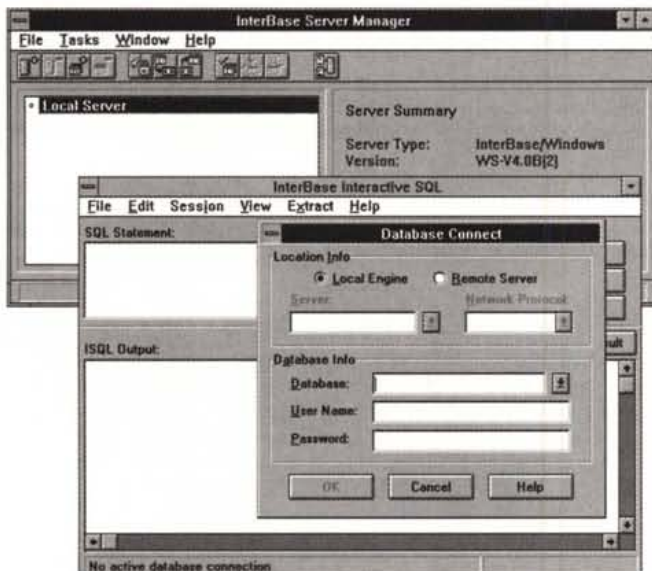
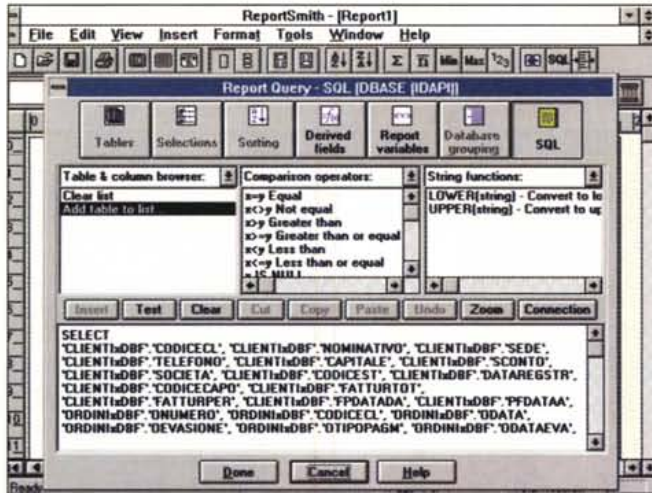


Figura 8 - Borland Local InterBase Server Manager e ISQL - Esempio di accesso ai dati in locale.

Nella versione «normale» di Delphi ci sono due strumenti che consentono di sperimentare l'accesso ai dati attraverso i servizi di InterBase, pur rimanendo in locale. Vediamo solo le Box «vuote» del Server Manager, con il quale si definisce il tipo di connessione, e dell'ISQL, InterBase Interactive SQL, che, una volta stabilito il collegamento, consente di costruire ed eseguire delle Query SQL. Abbiamo intenzione di provare, in un prossimo e specifico articolo, le varie funzionalità disponibili in Delphi per l'accesso ai dati.

operative, apribili, iconizzabili e, in certi casi, chiudibili a seconda delle necessità. Queste si affiancano alla Form su cui si sta lavorando, che è il «foglio da disegno», in cui si costruisce l'applicazione. Più tardi descriveremo in dettaglio l'ambiente operativo, potete già vederlo in figura 3, e i suoi componenti.

La specializzazione di Delphi è l'accesso ai Dati, accesso che avviene attraverso un «motore» che si chiama Borland DataBase Engine (ne vedete la collocazione in fig. 4) e che fa da interfaccia tra la nostra applicazione e il DataBase, di qualsiasi tipo esso sia, da un insieme di file DBF disponibili sul PC locale, fino alle complesse sorgenti di dati residenti, in formato SQL, su un Server. In Delphi esistono una serie di componenti che permettono di definire il DataBase a cui collegarsi, le tabelle da richiamare, le eventuali query da eseguire, e poi, scendendo più in basso, di indicare quali componenti elementari col-

legare a quali campi. Anche di questo parleremo un po' più dettagliatamente.

Il materiale

Lo scatolotto, in perfetto stile Borland, contiene il CD (il cui contenuto totale risulta essere di ben 110 megabyte) e i manuali, tutti in italiano meno uno.

Borland ha adottato l'intelligente provvedimento di lasciare il prodotto in inglese ma di tradurre in italiano i manuali. Questo facilita l'apprendimento e l'uso del prodotto da parte di quei molti che non conoscono benissimo l'inglese ma conoscono bene quello strano nuovo linguaggio, legato all'uso del computer, zeppo di termini inglesi, lasciati in inglese o italianizzati.

I manuali sono tanti, anche nella versione normale di Delphi. Una prima panoramica la facciamo consultando il piegholevo «Start Here» che elenca la do-

cumentazione disponibile, su carta, e quindi i manuali, o su CD. Start Here, pur essendo fatto di sole sei pagine, non è stato tradotto. Contiene anche una tavola sinottica in cui su due colonne sono riportati Argomenti e Sezioni dei Manuali in cui gli stessi sono trattati. Le colonne si chiamano For Information on... See...

I manuali, identici nel look, nell'ordine logico di utilizzo sono i seguenti.

Borland Delphi per Windows Guida all'Uso (460 pagine)

È diviso in sezioni. La prima descrive gli Elementi Fondamentali di Delphi e del suo ambiente operativo. La seconda, intitolata Nozioni Fondamentali, in pratica descrive le modalità di costruzione e di utilizzo dei vari componenti dell'Applicazione e quindi parla di Schede, di Menu, di Codice, di Progetti, ecc. La seconda sezione è dedicata agli Elementi di Programmazione. Tratta del linguaggio Object Pascal, della programmazione degli Oggetti e poi dei vari strumenti di supporto alla programmazione come il Debugger, il Compilatore, ecc.

La quarta sezione descrive una serie di applicazioni di Esempio che spaziano su svariati argomenti. Vengono anche trattati argomenti specifici come la gestione dei File, l'uso delle tecniche DDE e OLE, ecc.

Borland Delphi per Windows Guida per lo sviluppo di applicazioni di DataBase (210 pagine)

Delphi è particolarmente specializzato in applicazioni di interfaccia verso i DataBase e questo manuale è specificamente dedicato alle funzionalità di DataBase. Viene descritta l'architettura attraverso la quale l'applicazione Delphi vede i dati, in qualsiasi formato siano essi disponibili, e vengono descritti i componenti dell'architettura. Vengono poi descritti i vari oggetti presenti nella Palette Component di Delphi nelle due sezioni Data Access e Data Control. Nella prima sezione ci sono i componenti che servono per l'aggancio con DataBase, Tabelle e Query, nella seconda i componenti «sensibili ai dati», che mostrano quindi, direttamente, e nelle varie forme proprie del tipo di componente (semplici caselle di testo, liste, liste combinate, check box, picture box per i campi Blob, ecc.) i dati presenti nelle tabelle di origine.

Viene descritto l'uso del linguaggio SQL, che può essere usato direttamen-

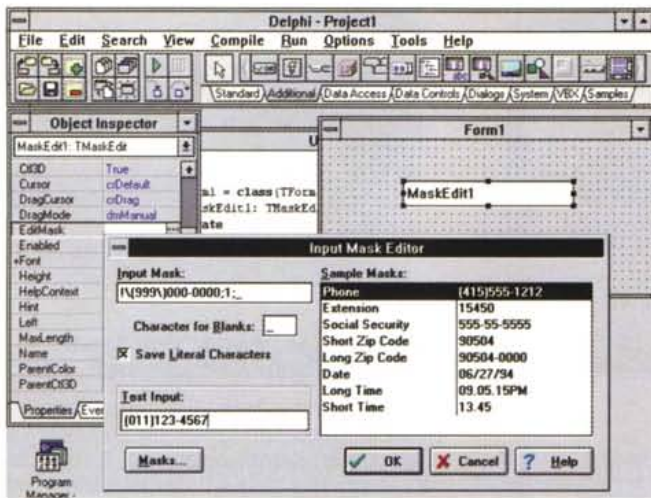


Figura 9 - Borland Delphi 1.0 - Definizione «fine» delle caratteristiche di una Input Mask.

Una pagina interessante nella Palette delle Componenti è quella Additional, che comprende una serie di control un po' meno... standard di quelli presenti nella pagina Standard. Qui vediamo come sia possibile sprofondare nelle varie finestre che permettono la definizione fine delle proprietà di un oggetto di tipo «Input Mask», ovvero della Casella di Modifica in cui i dati digitati devono rispettare un dato formato.

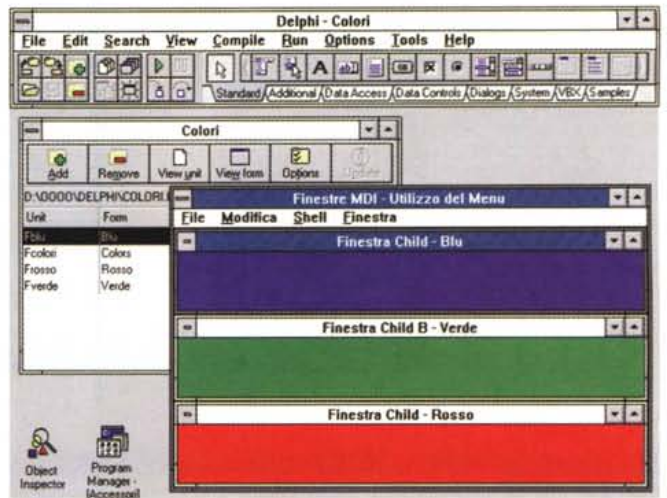


Figura 10 - Borland Delphi 1.0 - Applicazione MDI.

Quando si inizia una nuova applicazione si può decidere se creare un'applicazione MDI, Multiple Document Interface, come quella di prova che vi stiamo mostrando, oppure un'applicazione SDI, Single Document Interface, oppure un'applicazione con interfaccia carattere. Sullo sfondo vediamo la finestra Project che mostra l'elenco delle varie Form che costituiscono l'applicazione e che può servire come gestore delle Form stesse.

te da Delphi per creare delle Query sul database e quindi produrre delle Tabelle «virtuali». Viene trattato anche l'argomento Client/Server.

In appendice viene descritto il prodotto accessorio DataBase Desktop (fig. 5) che serve per creare e mettere a punto, al di fuori di Delphi, delle Query da riutilizzare... da dentro.

Borland Delphi per Windows Guida alla scrittura di componenti (170 pagine)

La programmazione Visuale è basata sull'assemblaggio di componenti pre-costruiti. Delphi ne mette a disposizione parecchie decine, secondo una tecnologia chiamata VCL (Visual Component Library). Ma con Delphi è anche possibile sviluppare propri componenti, che, una volta realizzati, sono utilizzabili al pari degli altri.

È chiaro che la realizzazione di componenti è un'attività di livello superiore, nel senso che è praticabile solo da un utilizzatore particolarmente esperto o da un programmatore.

Inoltre, e questo avvalora quanto appena detto, lo sviluppo di componenti non si può eseguire con gli strumenti visuali.

È possibile usare Delphi per creare non solo nuovi componenti interni ma anche librerie DLL, utilizzabili al di fuori di Delphi. È anche possibile un completo accesso alle API di Windows.

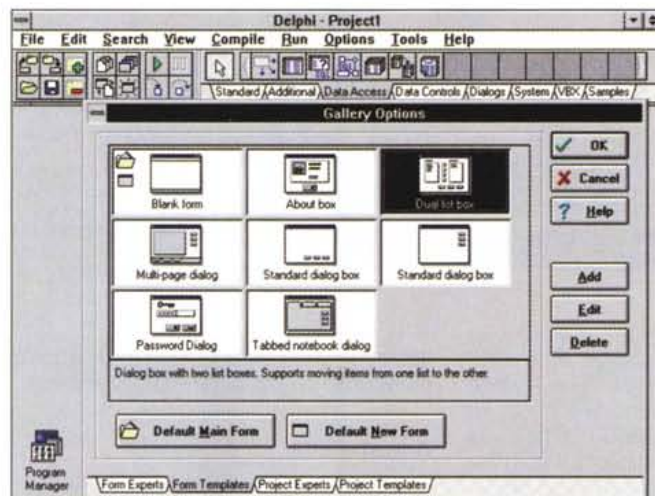


Figura 11 - Borland Delphi 1.0 - Creazione di una Nuova Form. Molto comodo lo strumento Gallery. Ad esempio quando si apre una nuova Form è possibile sceglierne il tipo, nel campionario che ne propone inizialmente otto. Ciascuna Form della Gallery è già riempita con una serie di Control da personalizzare a seconda delle necessità dell'utilizzatore.

Si pratica la vera programmazione Object Oriented. Si lavora partendo da uno dei componenti disponibili, variano alcune caratteristiche ed ereditano delle altre.

Entrano in gioco i concetti alla base della Object Orientation, classi, ereditarietà, ecc.

Questo manuale parla quindi di come creare i nuovi componenti, della programmazione OOP, della creazione delle proprietà, degli eventi e dei metodi. La seconda parte del manuale, molto opportunamente, presenta dei significativi esempi pratici.

Borland ReportSmith per Windows Creazione di Report (380 pagine)

Il ReportSmith è il generatore di Report della Borland. Per generatore di Report si intende un prodotto in grado di accedere ad una fonte di dati, di estrarli magari eseguendo su di essi delle unioni, delle selezioni, degli ordinamenti, dei calcoli di nuovi campi, poi di organizzarli, rielaborarli e riprodurli in un stampa.

ReportSmith è molto sofisticato, per-

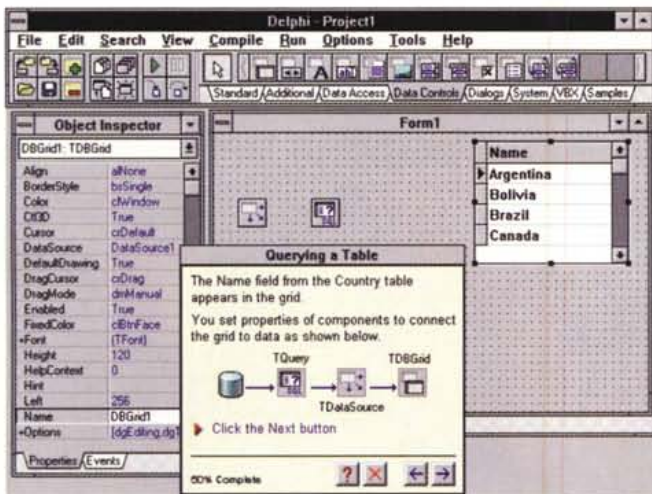


Figura 12 - Borland Delphi 1.0 - Tutorial interattivo.

Delphi mette a disposizione numerosi strumenti per imparare ad usarlo. C'è un Help molto sensibile al contesto e agli oggetti dell'ambiente, ci sono, nei manuali, sezioni dedicate all'apprendimento. C'è poi questo Tutorial interattivo, realizzato con la stessa tecnologia con la quale è realizzato quello presente in dBase, che permette di eseguire una serie di esercizi lavorando in maniera assistita, passo passo, direttamente nel reale ambiente Delphi.

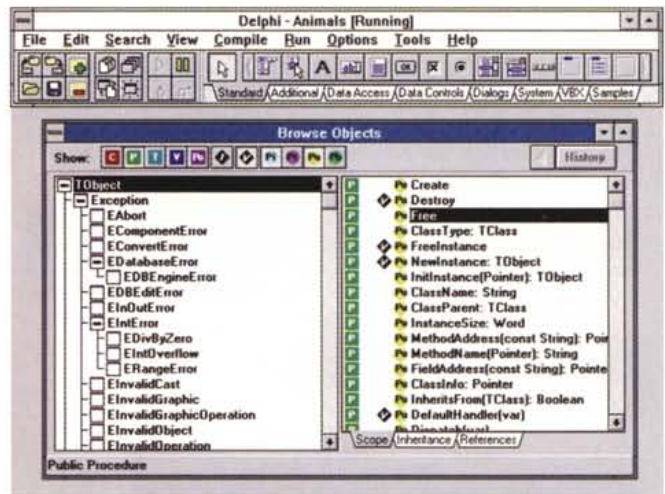


Figura 13 - Borland Delphi 1.0 - Il padre di tutti gli oggetti.

L'oggetto TObject è il primo antenato di ogni oggetto e quindi di ogni componente presente in Delphi. Chi sviluppa nuovi oggetti deve partire da un «antenato», al limite dal TObject, dal quale è possibile ereditare una serie di proprietà e dei metodi.

mette di:

- selezionare i dati escludendo quelli che non servono,
- combinare dati da più tabelle,
- ordinare rispetto ad uno o più campi, e raggruppare i dati,
- calcolare somme, medie, ad altro sui gruppi creati,
- inserire testate e piè di pagina,
- formattare i dati, anche ricorrendo a stili predefiniti,
- inserire nella stampa oggetti OLE, ad esempio Grafici, oggetti Blob, ad esempio immagini.

Inoltre, per una proceduralizzazione del processo, è possibile realizzare delle Macro e creare delle Dialog Box per la richiesta di parametri eventualmente utilizzati dal Report per selezionare i dati da stampare.

La lunghezza, in pagine, del manuale del ReportSmith la dice lunga sulla sua ricchezza.

Il manuale parla dell'aggancio al Data Source, della organizzazione dei dati, dei possibili interventi di tipo estetico e dei vari aspetti evoluti.

Se il Report deve essere distribuito ai vari utenti dell'applicazione si può usare il ReportSmith Viewer, anch'esso in dotazione.

Borland Local InterBase Server for Windows User's Guide (210 pagine)

Nella versione base del Borland

Delphi è presente anche la versione «single-user» del Borland InterBase Server. InterBase Server è la tecnologia che permette alle applicazioni DBMS della Borland, e quindi anche a Delphi, di accedere, tramite SQL-Link, a DataBase in formato SQL.

È chiaro che la versione Local rappresenta una drastica limitazione rispetto alla versione Remote. Può servire fondamentalmente in fase di sviluppo di un'applicazione, per verificare la correttezza delle modalità di accesso ai dati SQL, prima di convertirla all'uso in rete e prima di convertirla in una vera applicazione Client/Server. Comunque se l'applicazione deve rimanere solo locale è probabile che non convenga svilupparla prevedendo basi dati SQL.

Il Local InterBase Server si compone di due moduli: il Windows ISQL e il Server Manager.

Il primo è un applicativo a sé stante che consente di collegarsi al Server Manager, di scrivere l'istruzione SQL e di ottenerne in un'apposita finestra i dati risultanti (fig. 8).

Le procedure per accedere da Delphi, tramite il Server Manager, ai vari tipi DataBase SQL sono descritte nel secondo manuale, quello dedicato all'accesso ai dati.

Citiamo per completezza alcune delle caratteristiche più interessanti di InterBase:

- conformità con l'SQL ANSI Standard 1992,

- accesso simultaneo o DataBase multipli,
- ottimizzazione delle Query, per migliorare le performance,
- riconoscimento dei campi BLOB (Binary Large Object),
- riconoscimento dell'integrità referenziale dei dati, di tipo dichiarativo,
- possibilità di utilizzare «Stored Procedures», residenti ed eseguite nel Server,
- possibilità di utilizzare «Triggers», che sono moduli di programmi attivati automaticamente in fase di gestione del DataBase,
- riconoscimento dei vari tipi di Join,
- gestione degli accessi concorrenti,
- ecc.

Ribadiamo che l'utilizzo di base dati SQL, e quindi l'utilizzo dei servizi del Server InterBase, ha senso solo in applicazioni di tipo Client/Server, in cui Delphi svolge i compiti di Front End verso l'utente che accede, in maniera ottimizzata a seconda del tipo di operazioni che deve compiere, ai DataBase sul Server.

L'ambiente operativo di Delphi

Nell'analizzare l'ambiente operativo di Delphi occorrerebbe considerare due casi: che già si conosca il Visual Basic della Microsoft, che è attualmente il prodotto di sviluppo visuale più diffuso (o qualche prodotto simile) oppure che non si conosca nessun prodotto di tale

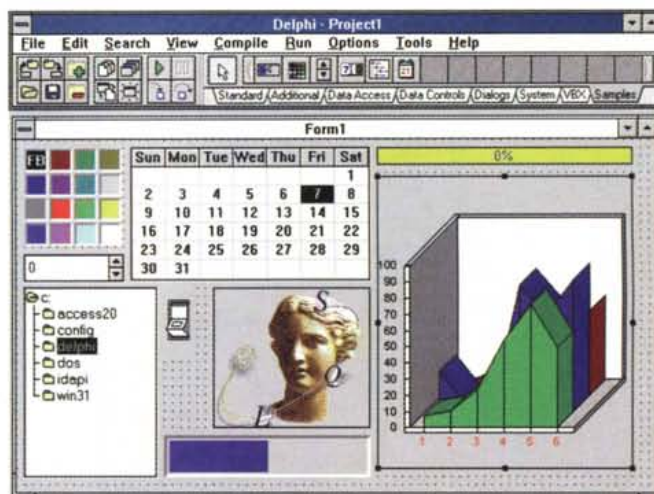
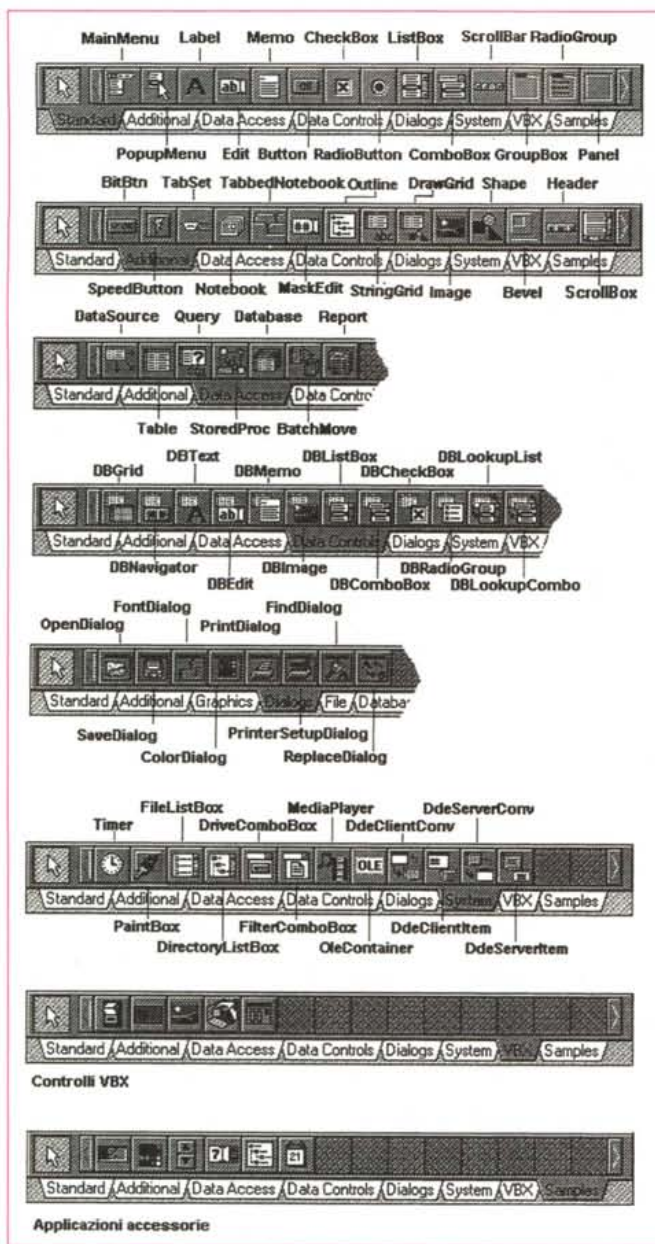


Figura 15 - Borland Delphi 1.0 - Oggettistica. Abbiamo piazzato in un'unica Form tutti i componenti presenti nella pagina VBX e nella pagina Samples della Palette dei Componenti. Sono degli oggetti preconfezionati di una certa complessità che possono tornare molto comodi, probabilmente mai tutti insieme, in applicazioni particolari.

Figura 14 - Borland Delphi 1.0 - La Libreria VCL al gran completo. Dal punto di vista dell'utente un componente è qualcosa da scegliere dalla Palette dei Componenti per essere usato nell'applicazione, e poi da manipolare nell'ambiente visuale di programmazione oppure attraverso il codice. Da un punto di vista più tecnico si tratta di un discendente, più o meno lontano, del TObject. La Palette dei Componenti Base di Delphi ne presenta inizialmente oltre 75.

gere o modificare) le proprietà dell'oggetto selezionato e gli eventi generati dall'oggetto. Le proprietà sono organizzate in modo gerarchico per facilitarne la gestione, ad esempio alla voce di primo livello BorderIcons, corrispondono una serie di voci di secondo livello (System, Max e Min). In molti casi (lo vediamo nella fig. 9, dedicata all'oggetto Input Mask) la definizione di una proprietà richiama proprie specifiche finestre di dialogo (ad esempio quella per scegliere i Font, quella per scegliere i Colori, ecc.);

- in secondo piano la finestra con il codice. Trattandosi di programmazione Object Based, il codice è legato al verificarsi di un evento su un oggetto. Nel nostro caso l'evento è il click sul pulsante e l'azione conseguente è la modifica della proprietà Text della casella Edit.

La sintassi completa è: ▼

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    edit1.text := 'Buongiorno';
end;
```

In Visual Basic invece la sintassi sarebbe:

```
Sub Command1_Click ()
    text1.Text = "Buongiorno"
End Sub
```

Come si vede, anche se l'esempio indicato è il più semplice possibile, si tratta di due strutture di programmazione identiche, con minime differenze dovute al fatto che Delphi usa come lin-

categoria. Nel primo caso si può sicuramente procedere per analogie, che tra Delphi e Visual Basic sono parecchie, nel secondo caso occorre fare una descrizione più dettagliata dell'ambiente e dei suoi componenti.

Per questo tipo di descrizione facciamo riferimento alla figura 3. In tale figura vediamo i principali componenti dell'ambiente di sviluppo Delphi:

- in alto il pannello che comprende il menu, in alto, e un po' più sotto due altri elementi. A sinistra la SpeedBar, pulsantiera con i comandi di tipo generale, che riguardano accesso ai file, organizzazione degli oggetti, e così via. Sulla

destra la Palette dei Componenti, organizzata in pagine specializzate per argomento.

Le pagine sono selezionabili tramite le linguette poste in basso;

- in primo piano la Form in lavorazione, che contiene due oggetti: un Pulsante (Button), su cui è stata posta la scritta Pigia, e una Casella di Testo (Edit), nella quale proveremo a scrivere qualcosa o direttamente da tastiera oppure da procedura;

- sulla sinistra la finestra Object Inspector dalla quale si possono gestire (leg-

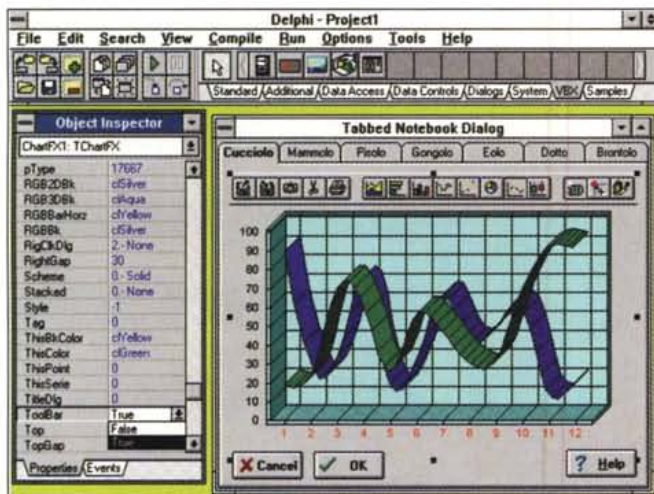


Figura 16 - Borland Delphi 1.0 - Finestra Tabbed e VBX Control.

Qui vediamo due cose. La finestra a linguette, nella quale si possono definire numero e caratteristiche delle varie pagine individuate dalle linguette. Poi vediamo il fenomenale componente ChartFX, che, essendo un VBX che rispetta le specifiche di Microsoft Visual Basic, si può utilizzare anche al di fuori di Delphi. È quasi un programma di Charting a tutti gli effetti, infatti dispone di una pulsantiera operativa che permette di cambiare, anche interattivamente, il tipo di grafico e le varie caratteristiche dei suoi elementi.

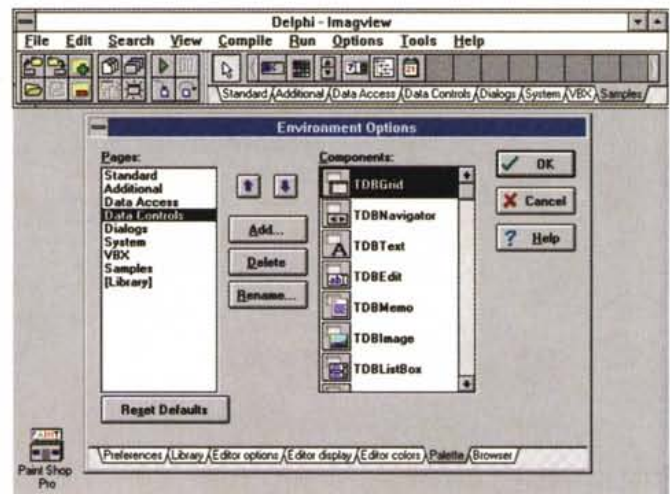


Figura 17 - Borland Delphi 1.0 - Personalizzazione del Pannello VCL.

La finestra a linguette Environment Options serve per configurare le varie opzioni che riguardano il modo di lavorare con Delphi. Sono, tra le altre cose, personalizzabili sia la SpeedBar che la Palette dei Componenti. L'operazione si esegue molto semplicemente da questa finestra. A sinistra l'elenco, allungabile, delle Pagine della Palette, a destra i vari componenti inseribili nelle pagine.

guaggio l'Object Pascal e VB il Basic. Nei due blocchetti di istruzioni le uniche righe che deve scrivere il programmatore sono quelle che indicano l'azione sulla casella di testo.

Esistono altre finestre di servizio e di utilità. Ad esempio nella figura 10 vediamo, oltre ad un'applicazione MDI, la finestra ProjectManager, che mostra l'elenco delle Form che compongono l'applicazione.

Nella figura 11 vediamo la videata della Gallery che mostra le varie tipologie di Form «pretrattate», tra le quali scegliere quella desiderata quando si inserisce, nell'applicazione in corso, una nuova Form. In pratica si tratta di Form con alcuni oggetti standard già piazzati a bordo.

Le Gallery riguardano sia le nuove Applicazioni, che possono essere di tipo MDI, di tipo SDI, oppure di tipo CRT (orientate alle videate DOS), che le nuove Form.

Per imparare

Indichiamo tre possibili metodi per imparare ad usare Delphi.

Il primo è l'uso del Tutorial Interattivo che permette di eseguire degli esercizi in maniera guidata ed operando direttamente nell'ambiente Delphi. La tecnologia utilizzata è la stessa del Tutorial del dBase V, provato nel numero scorso di MC. Purtroppo il numero delle lezioni è

esiguo rispetto alla vastità del prodotto e alle tipologie di applicazioni che permetterebbe di sviluppare. In figura 12 vedremo un momento di una lezione riguardante applicazioni orientate alla gestione dei DataBase.

Il secondo metodo, più approfondito, è quello di eseguire gli esercizi proposti nei manuali. Non è un metodo interattivo... e farà storcere il naso a chi non vuole studiare sui libri.

Il terzo metodo consiste nel ricorso all'Help di Delphi, che è molto efficace in quanto è sensibile al contesto ed agli oggetti. Ad esempio se si seleziona un qualsiasi componente, sia sulla Palette dei Componenti sia sulla Form, qualora vi sia già stato piazzato, e si preme F1 appare la finestra dell'help che ne descrive significato, ne elenca proprietà, eventi e metodi. Sulla finestra Object Inspector basta selezionare una qualsiasi proprietà e poi premere F1 per avere la spiegazione di quella proprietà, a cosa serve, come settarla, ecc. In fase di scrittura del codice basta selezionare una istruzione, una funzione o quant'altro per avere, premendo F1, tutte le spiegazioni del caso.

La libreria dei Componenti

Gli oggetti visuali inseribili nelle varie Form che costituiscono un'applicazione Delphi si chiamano Control (anche in Visual Basic si chiamano così). Si tratta di

componenti visuali appartenenti ad una serie di precise categorie. Da qui la sigla VCL, utilizzata in Delphi, che è l'acronimo di Visual Component Library, nel senso che Delphi propone una sua libreria di ben 75 componenti preconfezionati, che esplicita nell'apposita Palette dei Componenti, sempre al suo posto in alto a destra della videata, in cui questi vengono organizzati per argomento.

Esistono due grosse categorie di componenti, quelli che si vedono, possono essere attivati, propongono o ricevono dati, che si chiamano Windowed Control (es. una casella di testo), e gli altri, Nonwindowed Control, che si vedono e non sono attivabili (es. un'etichetta). Altri poi, ad esempio quelli da usare per stabilire gli «agganci» con le Tabelle di un DataBase, scompaiono in fase di esecuzione del programma.

Tutti i Control ereditano una serie di proprietà da un oggetto astratto «padre di tutti gli oggetti» (si chiama TControl). Questo fatto deve essere ben compreso da chi volesse sviluppare propri Control. Il nuovo Control deve comunque partire dal TControl o da un suo discendente (nella figura 13 vediamo l'Object Browser che mostra tutta la gerarchia).

In figura 14 vediamo un collage con tutti i Control standard. La pagina cui appartengono la potete vedere leggendo la linguetta. Molto sinteticamente elenchiamo le varie categorie:

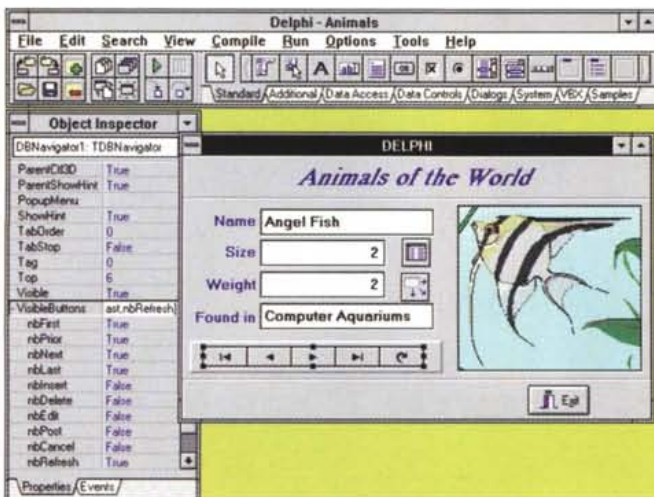
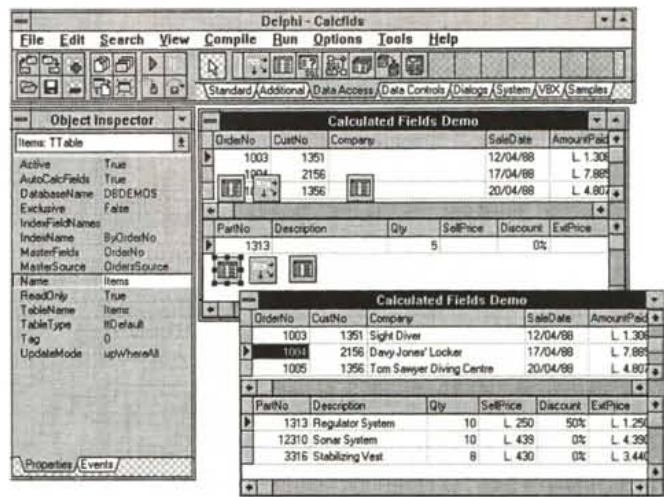


Figura 18 - Borland Delphi 1.0 - Controlli per la gestione del DataBase.

Una delle specialità di Delphi è l'accesso ai dati che viene favorito dalla presenza di una specifica pagina, anzi due, della Palette dei Componenti. I controlli DataSource e Table, che vedete nella Form su cui stiamo lavorando, scompaiono quando il programma viene eseguito. Sulla Form si può inserire il controllo DBNavigator che contiene fino ad un massimo di dieci pulsanti, che si possono utilizzare o meno impostando opportune specifiche nell'Object Inspector.

Figura 19 - Borland Delphi 1.0 - Gestione evoluta del DataBase.

Questo è un fotomontaggio in cui vediamo sia la Form in lavorazione che la Form in esecuzione. Il particolare che ci interessa è la possibilità di definire una SubForm indicando la modalità di collegamento con la Form Master. In alto vediamo i vari Control della sezione DataAccess, ai quali vanno aggiunti quelli della sezione Data Control, in cui troviamo i componenti «sensibili ai dati».



- Standard, ovvero la collezione degli oggetti standard di Windows, menu, caselle di modifica, combo box, etichette, pulsanti di opzione, ecc. L'unico problema è decidere in che lingua indicare i vari oggetti, per il resto sono facilissimi da usare;

- Additional, oggetti Windows... meno standard, anche se molti di questi sono abbastanza diffusi. Quindi Pulsanti con immagini che variano, Caselle di testo con testo formattato, Griglie di due tipi, Immagini e Profili grafici posti sulla Form, Caselle di testo a pagine, poi Cornici a linguette, ed altro;

- Data Access, per definire tutte le specifiche di collegamento ai dati;

- Data Control, una serie di elementi «sensibili ai dati», in grado di ricevere e produrre dati;

- Dialog, accesso alle Dialog Box standard di Windows. File Apri, Scelta dei Colori, dei Font, Dialogo con la Stampante, ecc.;

- System, accesso a tutte le funzioni e Servizi di sistema, File Manager, DDE, OLE, Timer, servizi Multimediali, ecc.

Le ultime due sezioni riguardano i controlli VBX, e quindi la possibilità di usare librerie VBX che rispettano lo standard Microsoft. Delphi ne porta, e mostra nella sezione VBX, una piccola serie. L'ultima sezione Samples contiene esempi di controlli autoconstruiti. C'è una palette per la scelta del colore, un calendarietto per la scelta della Data, ed altro.

Nella figura 15 vediamo un assemblaggio in un'unica Form di tutti i Control presenti nelle due ultime pagine della Palette.

Nella successiva, la 16, vediamo una Form a linguette e, sulla pagina scelta, un Grafico realizzato con l'eccezionale VBX ChartFX in dotazione. L'aspetto che più ci ha impressionato di questo Componente VBX è che dispone di una Toolbar con una quindicina di pulsanti attivi che agiscono sul grafico visualizzato, al limite anche per cambiarne il tipo. Il VBX è standard MS Visual Basic, e infatti, lo abbiamo provato, funziona anche con VB 3.0 Professional.

La VCL risulta ricchissima, contiene tutti i componenti che potrebbero essere necessari, oppure solamente utili, in un'applicazione Windows. Ci sono anche quelli più evoluti come l'Outline, per proporre liste di dati organizzati gerarchicamente, o il Tabbed, per la creazione di finestre a pagine «linguettate». Si tratta di due Control tipici di Windows '95. Anche la sua organizzazione (stiamo parlando della VCL) ci sembra razionale al punto che sembra improbabile la necessità di dover personalizzare la sua Palette, operazione comunque possibile (si veda la figura 17).

A proposito di Windows '95 va detto che Delphi 1.0 utilizza gli stessi oggetti tipici del nuovo sistema operativo. Applicazioni sviluppate con Delphi 1.0 potranno conseguentemente essere direttamente compilate nella prossima ver-

sione '95, senza dover eseguire interventi sul codice.

L' Accesso ai Dati

La Borland ha una tradizione consolidata nel settore DataBase e quindi è naturale che anche il nuovo nato, il nostro Delphi, sia specializzato in questa attività così importante.

Delphi dispone di una grande quantità di Controlli specializzati nell'accesso ai dati. I Controlli sono resi disponibili in due pagine della Palette dei Componenti. Nella pagina Data Access troviamo i Controlli che servono per stabilire il collegamento tra l'applicazione e i dati. Occorre definire quale è il DataBase (nel caso utilizzo di file DBF, il nome del DataBase è il nome della Subdirectory), quali sono le Tabelle, quali sono le Query. Si possono definire, in caso di applicazioni Client/Server, le Stored Procedure da lanciare sul Server.

Nel caso in cui si voglia realizzare una Form con una SubForm occorre inserire più controlli e tra le specifiche di questi indicare quale siano i campi di collegamento tra le due Tabelle e/o Query.

Una volta definiti i collegamenti usando i vari Control che durante l'esecuzione del programma spariscono, si passa alla sezione Data Control in cui troviamo una serie di elementi «sensibili ai Dati». Caselle di Modifica, Etichette, Liste e Combo Box, Griglie, Pulsanti di Opzione, Picture Box, ecc., per i quali va defi-

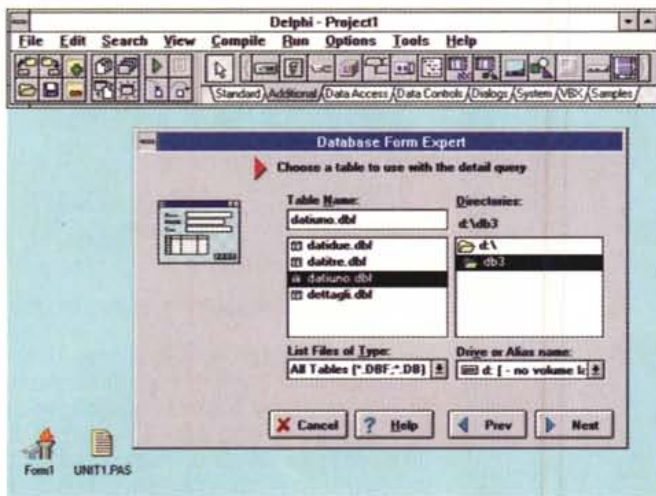


Figura 20 - Borland Delphi 1.0 - DataBase Form Expert.

Per facilitare la predisposizione delle Form necessarie in un'applicazione orientata alla gestione dei dati, Delphi mette a disposizione un Form Expert, ovvero una procedura di autocomposizione con la quale si possono realizzare alcuni tipi fondamentali di Form. Il tipo più evoluto è quello Multitable che si usa per mostrare in un'unica scheda i dati di due tabelle in relazione Uno a Molti tra di loro.

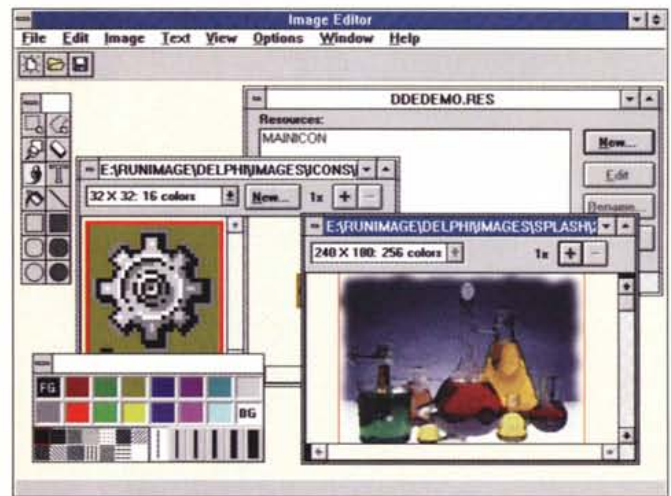


Figura 21 - Borland Delphi 1.0 - Strumenti ausiliari: Image Editor.

Tra gli strumenti accessori troviamo l'Image Editor, che serve per generare BitMap, Cursors, Icone, ma soprattutto Immagini di Risorse, utili nell'applicazione che si sta sviluppando.

nita l'origine dei dati.

Lo strumento principale per la navigazione tra i record è il controllo DBNavigator che propone un massimo di dieci pulsanti per scorrere, per cancellare, inserire, modificare i record (figg. 18 e 19).

In ogni caso il prototipo della Form, in un'applicazione che interfaccia dei dati, può essere costruito con una procedura automatica, tipo autocomposizione (fig. 20). Si richiama dal menu Help e permette anche di realizzare una Form di tipo Master-Detail, in cui ci sia una testata, una SubForm con i dati in colonna, e una zona dettagli mostrati in una griglia.

Gli accessori

Vediamo di nuovo la figura 2, che mostra le varie Icone scaricate nel gruppo Delphi, per controllare se abbiamo parlato di tutto. C'è Delphi, si sono il ServerManager e l'ISQL, propri della versione locale di InterBase, c'è il ReportSmith con il suo Viewer, c'è il DataBase Desktop e l'icona per la configurazione del Borland DataBase Engine. Troviamo poi una serie di Help riguardanti una serie di argomenti, anche di tipo più tecnico, che non possiamo trattare in sede di prova, poi un paio di ReadMe.

Ci sono alcuni strumenti evoluti per la diagnostica dell'applicazione, il WinSpector, o, per l'analisi dei Messaggi Windows, il WinSight. C'è l'Image Edi-

tor, che serve per preparare, parallelamente rispetto a Delphi, eventuale materiale grafico, di vario tipo, necessario all'applicazione (fig. 21).

Conclusioni

È troppo presto per pronunciarsi in maniera precisa su un prodotto così nuovo, così complesso e così interessante. Personalmente conosco bene il Visual Basic della Microsoft per cui nel vedere Delphi della Borland non ho potuto evitare di fare continui confronti.

Le modalità di utilizzo di Delphi sono abbastanza analoghe a quelle del Visual Basic, anche se, ripetiamolo, il primo è un compilatore ed il secondo un interprete. Gli elementi principali in gioco sono pressappoco gli stessi.

Delphi, che è di due anni più giovane del VB 3.0, ha molte cose in più. Innanzitutto molti controlli, tutti subito disponibili, poi molte proprietà in più. Ad esempio, tanto per citarne una tra le più evidenti, ogni oggetto ha la proprietà Hint e ShowHint, con le quali è possibile definire una spiegazione di ciascuno oggetto, spiegazione che appare nei pressi dell'oggetto stesso quando il mouse ci passa sopra. Anche l'ambiente in cui si opera è molto più evoluto. È possibile eseguire selezioni multiple di oggetti per cambiare le proprietà di tutti con un'unica operazione. Sono disponibili funzionalità di allineamento degli oggetti nella Form ed altri strumenti che

velocizzano il lavoro.

È stata adottata la tecnica «Two Way Tools» che consiste nella possibilità di lavorare, il più possibile, nell'ambiente Visuale e ottenere il Codice e, viceversa, nello scrivere direttamente il codice ed ottenere il risultato sulla Form.

Bisognerà vedere, conducendo specifiche e più approfondite prove, quanto possano essere spinte le funzionalità di accesso ai dati, che, come detto più volte, si avvalgono di un gran numero di controlli specifici, e quindi, alla fin fine, se Delphi possa essere utilizzato per sviluppare complesse applicazioni gestionali, anche in caso di aggancio a DataBase SQL su Server e quindi condivisi in rete da molti utilizzatori.

In particolare pensiamo di eseguire una prova di approfondimento della sezione accesso ai Dati già dal prossimo numero di MC.

Il linguaggio Object Pascal non presenta difficoltà, nel senso che non occorre saperlo prima o impararlo prima, lo si impara via via che, usando Delphi, occorre scrivere del codice.

L'impressione è che Visual Basic, oggi il più usato strumento di Sviluppo Visuale per Windows, abbia trovato un temibile concorrente. Vedremo se la prossima versione 4.0 di VB offrirà nuove caratteristiche tali da far riequilibrare il confronto, che oggi vede favorito Delphi. Anche i primi dati relativi alle vendite in USA confermano questo giudizio.