

## dBASE IV e Paradox 3.0 Generatori di maschere e di report

di Francesco Petroni

Chiudiamo con questo la serie di articoli scritti in occasione dell'uscita, pressoché contemporanea, del dBASE IV e del Paradox 3.0, nelle versioni originali in americano.

Nei numeri precedenti di MC abbiamo dapprima descritto in termini generali i due prodotti, poi abbiamo analizzato in particolare i loro aspetti «relazionali» e infine, lo scorso numero, abbiamo verificato le possibilità connesse con l'uso del QbE, il Query by Example, che è un linguaggio di interrogazione degli archivi presente in ambedue i pacchetti.

In questo articolo conclusivo, che è l'ultimo condotto in «slalom parallelo», analizzeremo i generatori di maschere e i generatori di report.

Questo non significa assolutamente che siano «esauriti» gli argomenti e che quelli trattati siano stati totalmente approfonditi

Sotto un prodotto di gestione Data Base (DBMS in americano) si nascondono infatti infinite funzionalità, ognuna delle quali potrebbe richiedere decine di pagine di approfondimento, si nascondono infinite tematiche di tipo generale legate alle varie teorie di progettazione dei Data Base, si nascondono infine infinite problematiche applicative, sulle quali, in ultima analisi, il singolo prodotto andrebbe concretamente provato.

Ad esempio il problema che genericamente si può definire Progettazione di un Data Base può essere trattato in maniera teorica per definire le terminologie, i concetti fondamentali, che prescindono quindi dai prodotti e dalle applicazioni. Può anche però essere esaminato nel contesto di un prodotto soft-

ware, che dispone di proprie funzionalità che permettono di sviluppare concretamente il progetto. Può essere infine calato in una realtà applicativa, e in tal modo si effettua la «cosiddetta» analisi dei dati, che sta alla base di ogni progetto informatico.

Al contrario, sotto un prodotto di tipo Grafico, ad esempio un Paint, non ci sono aspetti concettuali, chi lo apprende deve solo imparare funzionalità che altro non sono che l'applicazione di concetti che già possiede. Un DBMS è l'opposto, in quanto l'apprendimento della varie funzionalità è un aspetto del tutto irrilevante rispetto ai concetti che ci stanno sotto e che bisogna comunque padroneggiare.

Ad esempio, nell'impostare una condizione in cui entrano in gioco operatori AND e OR, la difficoltà maggiore consiste non tanto nel saper come si fa con il prodotto X, quanto nel padroneggiare il concetto di AND e OR, che è un concetto del tutto astratto e che esiste indipendentemente dal dBASE IV, dal Paradox, dai computer e anche... dall'uomo.

Stesso discorso sulle teorie relazionali, che sono state trattate da centinaia di studiosi per decine di anni, e che oggi, nei DBMS vengono messe in pratica e rese disponibili come Funzioni da attivare, al pari delle altre, seguendo alcuni semplici passi operativi.

Un DBMS permette di definire ed utilizzare delle Relazioni, ma non serve a nulla capire «come si fa» se prima non si sa perfettamente cosa è una relazione, quali sono i tipi di relazione e così via.

Nel presentare i vari prodotti DBMS, che in questi ultimi anni sono diventati potentissimi, ci troviamo sempre di fronte al dilemma di dover illustrare sia gli aspetti semplici, a beneficio anche dei «pivelli», ma anche gli aspetti avanzati (si veda uno degli scorsi articoli

```
@ <riga>,<colonna> SAY <espressione>
      PICTURE <espressione>
      FUNCTION <lista funzioni>
GET <variabile>
OPEN WINDOW <nome finestra>
      PICTURE <espressione>
      FUNCTION <lista funzioni>
      RANGE <basso>,<alto>
      VALID <condizione>
      ERROR <messaggio>
      WHEN <condizione>
      DEFAULT <espressione>
      MESSAGE <testo>
      COLOR <primopiano>,<evidenziato>
```

Figura 1 - dBASE IV - Sintassi dell'istruzione @ <r>, <c> SAT., GET. Si può dire che in dBASE il comando tanto più è corto tanto più è potente. Citiamo ad esempio il comando \$ (segno dollaro) che esegue un test di presenza di una stringa in un'altra, il comando & (e commerciale) che assume il significato di Macro, infine il comando @ (familiarmente detto chiocciolina) che ha una sintassi ricchissima e che permette di risolvere quasi tutti i problemi di visualizzazione e di immissione.

basato su un caso studio con cinque archivi), cercando di definire in due parole le tipologie di problemi che risolvono, sui quali problemi però le centinaia di studiosi su citati hanno passato la loro vita.

### Le nuove generazioni dei generatori in modalità Full Screen

La possibilità di poter disegnare l'aspetto esteriore e contenutistico di una maschera o di una stampa direttamente sul video, è una conquista della Informatica Individuale e rappresenta, come molte delle innovazioni tecniche dovute alla sua diffusione presso le masse, una sorta di «uovo di Colombo».

Ormai tutti i prodotti dispongono di più ambienti Full Screen, in cui si opera con funzionalità tipiche di un editor e attraverso i quali si costruiscono sia maschere di immissione o visualizzazione (è il caso del dBASE e del Paradox) sia, in altri prodotti, l'intero archivio e quindi anche la struttura con le varie tipologie di campi in esso contenuti.

In altre parole mentre ci si occupa dell'aspetto estetico della maschera (scritte fisse, cornici, intestazione dei dati, posizione e formato dei dati) si genera contestualmente un file, che poi sarà possibile richiamare direttamente quando si tratterà di gestire dei dati, se si tratta di una maschera, o di stamparli, se si tratta di un report. I Generatori delle ultime generazioni dispongono anche di funzioni spinte, un tempo risolvibili solamente ricorrendo alla programmazione, ma oggi utilizzabili direttamente al pari delle funzionalità più elementari.

Ad esempio la creazione di dati calcolati, ottenuti mediante formule di calcolo che entrano in gioco in sede di utilizzo della Maschera o del Report, sui quali si possono eseguire ulteriori calcoli insiemistici.

Oppure possibilità di definire, usare e «gettare dopo l'uso» delle variabili indispensabili in fase di gestione della maschera e di produzione del report, ma poi inutili.

È ormai sempre possibile eseguire dei «Lookup» dinamici, mediante i quali vengono prelevati dati anche da archivi esterni.

Questo aspetto, legato all'utilizzo delle prestazioni «relazionali», è molto importante, in quanto è possibile risolvere, direttamente in fase di generazione della Maschera o del Report, aspetti procedurali complessi.

### Il comando «Chiocciolina» del dBASE IV

Col dBASE IV l'utilizzo delle varie strutture assistite genera sempre o un programma o degli pseudoprogrammi in cui entrano comunque comandi che sono anche utilizzabili direttamente.

Ad esempio, come abbiamo visto nei numeri scorsi, il linguaggio di interrogazione Query by Example produce un file in cui vengono scritte le varie istruzioni di apertura archivi, di apertura degli indici, di impostazione di filtri e di dichiarazione delle relazioni.

Lo stesso discorso va fatto per il Generatore di Maschere e il Generatore di Report, che nel dBASE IV è diventato anch'esso un generatore di programmi a tutti gli effetti.

Ma mentre il Generatore di Report deve utilizzare un buon numero di comandi di programmazione, ad esempio per creare dei cicli di scorrimento e dei sottocicli di raggruppamento e di calcolo, il Generatore di Maschere utilizza in

pratica solo il comando chiocciolina, di cui qualsiasi utilizzatore del dBASE è bene che conosca perfettamente la sintassi:

```
@ < r > . < c > SAY
< espressione > GET < specifiche >
```

Il comando «chiocciolina» serve sia in fase di visualizzazione dei dati, sia, soprattutto, in fase di editazione (immissione e/o modifica) dei dati stessi.

Ad esempio, in comandi diretti o in un programma, le istruzioni:

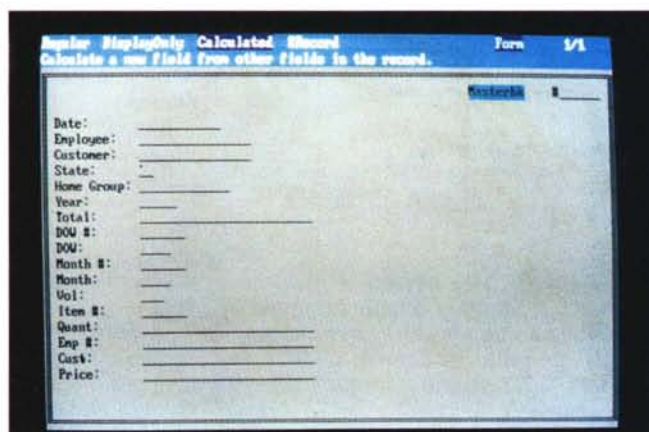
```
A=12345.678
@ 10,40 SAY A PICTURE
"###.###.###.###"
```

producono la scrittura, a partire dalla riga 10 e colonna 40, del numero, nella forma 12.345,678. Per inciso avvertiamo i vecchi «dBtreisti» che in dBASE IV si può (finalmente) usare il punto per separare le migliaia e la virgola per separare i decimali.



Figura 2 - dBASE IV Costruzione della maschera in Full Screen. Il generatore di Forms del dBASE IV lavora in modalità Full Screen e utilizza una serie di dialog-box in cascata. Il risultato dell'operazione è trasferito in un file di tipo testuale, che ha desinenza \*.FMT, e che contiene solo istruzioni «chiocciolina», e che quindi è molto opportuno saper interpretare.

Figura 3 - Paradox 3.0 Costruzione della maschera Full Screen che permette di disegnare direttamente sul video, mediante una funzionalità di editor, sia l'aspetto esteriore della maschera che l'aspetto strutturale «sottostante», il formato, gli attributi dei vari campi, e definire i vari controlli da eseguire in fase di immissione o modifica. Qui vediamo la Maschera Standard costruita dal Paradox, dalla quale conviene partire per qualsiasi personalizzazione.



Num.	Data	Art.	Descrizione	Quant.	Prezzo	Sconto	Importo
001	11 11 07	0003	MARBINO	2	1500	00	300000
004	12 02 09	0004	COLLA	100	1000	00	1000000
009	1 03 09	0001	GOMMA	20	300	00	60000
012	21 03 09	0005	CANCELLINO	30	700	70	147000
010	22 04 09	0006	PAPIRINE	100	500	75	375000
022	24 05 09	0000	TEMPERINO	4	500		

Figura 5 - Paradox 3.0 Generatore di Report. Paradox 3.0 e dBASE IV dispongono di due generatori di Report molto simili tra di loro, sia come modalità operativa, che è anch'essa Full Screen, che come possibilità funzionali. Permettono tutti e due di lavorare per «fette orizzontali» in cui suddividere logicamente i dati stampati.

Numero Codice Ordine	Descrizione	Data Ordine	Quantità	Prezzo	Sconto	Importo Calcolato
TOTALE DEL CLIENTE						
TOTALE GENERALE						

Figura 4 - Paradox 3.0 Maschera Multitable e Multirecord. Le applicazioni in cui occorre gestire più archivi e/o più record contemporaneamente sono frequentissime. Il Paradox versione 3.0 consente di costruire, anche in modalità interattiva, maschere uniche che alimentano il file Master (un solo record in alto) e uno o più file Linked (più record, uno per riga e con le righe che «scrollano») in basso. Tra Master e Linked deve esistere una relazione di tipo «Uno a Molti».

zione se questa richiede l'uso di funzioni più spinte di quelle accessibili dal generatore.

### Maschere Multitable e Multirecord con il Paradox 3.0

Premettiamo a questo argomento l'informazione che per ogni Table (in Paradox gli archivi si chiamano così) si possono costruire fino a 16 Maschere e fino a 16 Report differenti (in figura 3 vediamo la maschera Standard disponibile automaticamente per ogni Table, come prima delle sedici).

In tutte le applicazioni di complessità media od elevata è necessario lavorare con più archivi collegati tra di loro da regole di relazione. E in buona parte di queste occorre lavorare contemporaneamente su più archivi e/o su più record di uno stesso archivio. In tale caso occorre costruire maschere uniche nelle quali si visualizzano e si gestiscono dati di più archivi e più record.

Il Paradox 3.0 offre alcune funzionalità che permettono di costruire questi particolari tipi di FORM, e, molto opportunamente, nei manuali sono anche esposte le teorie Relazionali che stanno sotto a queste problematiche che sono oggetto di studio continuo da parte dei tecnici.

In dBASE IV invece non esiste la possibilità di costruire, con apposite funzionalità, maschere multiarchivio e/o multirecord. Il che non vuol dire che non si possa in assoluto, ma che richiede una maggiore conoscenza del prodotto, un maggiore tempo di sviluppo, ma permette di risolvere qualsiasi tipo di problema.

Torniamo al Paradox 3.0 per dire che consente di costruire, anche in modalità interattiva, delle maschere uniche che alimentano il file Master (un solo record in alto) e uno o più file Linked (più record, uno per riga e con le righe che «scrollano») in basso. Tra Master e Linked deve esistere una relazione «Uno a Molti», cosa che è anche indicata in alto in tale tipo di maschera.

Il processo costruttivo è abbastanza ovvio e richiede alcuni passi da eseguire nella giusta sequenza.

Dapprima si disegnano le due strutture dei due archivi, rispettando la regola che nel secondo archivio (quello che apparirà «a righe») ci sia un campo chiave, o, se la chiave è composta, tale campo sia il primo, che deve corrispondere ad un campo (non necessariamente chiave) del primo archivio.

Questa regola è praticamente ovvia, anzi è presente nel «decalogo» di numerose teorie relazionali.

La istruzione PICTURE, collegata alla chiocciolina, che permette numerose specifiche, serve in questo caso per modificare l'aspetto esteriore del dato visualizzato.

La chiocciolina usata con il comando GET, invece, assume la funzione di vero e proprio filtro attraverso il quale validare il dato in immissione. Permette inoltre numerose varianti nella messaggistica. Ad esempio:

```
A= "
B=1
@ 10,40 GET A PICTURE "!!"
WHEN (B=1)
VALID (A$"SI NO")
MESSAGE "Immetti SI oppure NO "
ERROR "Risposta sbagliata"
READ
```

La PICTURE «!!» traduce in maiuscolo qualsiasi carattere alfabetico immesso (sia minuscolo che maiuscolo). WHEN permette di definire quando richiedere l'immissione e in questo caso viene richiesta in quanto B=1. VALID esegue un test che se positivo fa accet-

tare l'immissione, altrimenti visualizza un messaggio di errore indicato dopo la specifica ERROR. MESSAGE manda un messaggio in riga 24, in corrispondenza della digitazione del dato.

Le possibilità sono numerose altre e per necessità di spazio sorvoliamo, rimandando alla figura 1 che espone la sintassi completa.

In pratica con la funzione chiocciolina è possibile impostare direttamente una buona parte dei controlli e della messaggistica necessari in qualsiasi lavoro di immissione o modifica dati in un archivio.

Il lavoro in modalità Full Screen (vedi figura 2) è molto assistito e rappresenta sicuramente il mezzo più rapido per costruire una maschera, salvo poi, se necessario, affinarla editando il file che viene generato, oppure utilizzare lo stesso file in un programma più tradizionale.

In altre parole il generatore di maschera può servire anche al programmatore per preparare una bozza iniziale del lavoro, che poi può essere utilizzata come base per lo sviluppo dell'applica-

Poi si costruisce la maschera del secondo archivio, in cui si mettono una o più righe di intestazioni, linee, ecc. e una, una sola, linee con i dati. Nel nostro esempio a questo punto dobbiamo aggiungere una cosa che, per non perdere il filo del discorso, diremo poi.

Costruita la prima riga, con la funzione Multirecord, la si duplica per un certo numero di righe che dipendono da quante righe si vuole riservare all'archivio superiore.

Si fanno un po' di prove di immissione, si inseriscono i criteri di validazione dei dati, il loro formato estetico, ecc. Quando tutto è a posto si passa a costruire la parte superiore della maschera che riguarda l'archivio individuato come «Master».

Durante la costruzione di questa stessa maschera, si attiva la funzione Multi-Table che chiede il nome della seconda maschera e quale campo del primo archivio corrisponda con il primo campo (quello chiave) del secondo. Se non sono stati fatti errori, la maschera Multi-Table è pronta (figura 4).

A proposito delle formule di validazione che si possono utilizzare per ciascun campo, citiamo la possibilità, che noi abbiamo utilizzato nel nostro esempio, di eseguire del Lookup su archivi esterni, dai quali caricare uno o più campi.

Nell'esempio in figura i dati delle varie righe Ordine che riguardano l'archivio Articoli sono stati caricati direttamente da quest'ultimo. Citiamo la possibilità di eseguire questa operazione in maniera guidata premendo F1 in corrispondenza del campo che permette l'aggancio (nel nostro caso il codice articolo). Appare la maschera dell'archivio Articoli sul quale si può cercare il dato voluto e premendo F2 provocare lo scarico dei dati, dei vari campi in comune, sulla riga dell'ordine.

I più esperti in DBMS avranno «appizzato le orecchie» (a Roma si dice così) di fronte a questa novità. Per i meno esperti diremo che in pratica si alimentano contemporaneamente due archivi separati, nel nostro caso quello dei Clienti e quello degli Ordini, e Paradox si occupa di riempire il campo Codice Cliente che deve essere presente, ma non visualizzato, nell'archivio secondario, che è quello che garantisce il collegamento.

Mentre la funzionalità ci è parsa molto sofisticata, non ci ha del tutto convinto l'aspetto operativo riguardante il controllo dei dati in immissione.

Poiché Paradox è utilizzabile anche direttamente nel formato Tabellare, le funzioni di controllo dei dati sono attivabili dall'ambiente Edit e non da quella

Figura 6 - Paradox 3.0 Report a video.

Tutti i Report possono essere inviati direttamente su carta, oppure inviati sul video per un controllo da eseguire, in fase di predisposizione, sull'estetica o sui contenuti, oppure su file, per una stampa successiva. Qui vediamo il Report, visto in preparazione sulla foto precedente, inviato sul video.

Now Viewing Page 1 of Page Width 1  
Press any key to continue...

Eseguito il 6/08/89  
Page 1

STAMPA DEGLI ORDINI PER CLIENTE

Numero Ordine	Codice Art.	Descrizione	Data Ordine	Quantità	Prezzo	Sconto	Importo Calcolato
CLIENTE C001 PERKINS							
001	A003	QUADERNO	11.11.87	2	1.500	90	270.000
004	A009	COLLA	12.02.89	330	1.200	80	31.680.000
009	A001	GOMMA	1.01.89	20	300	80	480.000
TOTALE DEL CLIENTE C001 PERKINS							32.430.000
-----							
CLIENTE C002 JOHNSON							
002	A004		1.03.89	10			
003	A003	QUADERNO	1.02.89	30	1.500	80	3.600.000
006	A009		1.03.89	10			
007	A001		1.04.89	6			
011	A004		1.02.89	12			
TOTALE DEL CLIENTE C002 JOHNSON							3.600.000

Report Fields Bands Words Go To Print Exit 6:25:35 am

Page n. 999

CLIENTE CODICE NUMERATIVO

VEDI/TT CODICE NUMERATIVO

ORDINE N.999	DEL. DD/MM/YY	XXXX	XXXXXXXXXX	XXXX	XXXXXXXXXX
articolo descriz.	quantita'	costo unit.	importo		
XXXX XXXXXXXXXX	9,999	9,999	999,999,999		
TOTALE IMPORTO DELL'ORDINE		SCONTO	99	999,999,999	
		IMPORTO SCONTATO	999,999,999		

Report Summary Band Page Footer Band

Report C:\dbase\corso\NREP06 Band 1/7 File:Prova Cap:10

Alt Field:F5 Select:F6 Move:F7 Copy:F8 Size:Shift-F7

Figura 7 - dBASE IV Generatore di Report. Viene abbandonato il vecchio e superato Report Generator del dBASE III per uno nuovo di zecca, che permette più tipologie di default (Tabulare, Scheda e Lettera), un maggior controllo strutturale, un più spinto utilizzo delle funzionalità di calcolo e infine, e anche questo non guasta, un miglior utilizzo ai fini estetici delle stampe.

Forms che serve per la generazione della maschera e questo costringe in fase di confezionamento della maschera a passare più volte da una funzionalità all'altra.

### I Generatori di Report

Paradox 3.0 e dBASE IV dispongono ormai di due generatori di Report molto simili tra di loro, sia come modalità operative, che sono anch'esse Full Screen, che come possibilità funzionali.

Anche in questo ambito, si nota nei due prodotti una tendenza alla unificazione nel linguaggio, cosa che si manifesta in molti altri aspetti. Si veda per esempio l'adozione del Query by Example, come linguaggio universale per l'interrogazione degli archivi.

Al di là degli aspetti operativi, che sono improntati alla massima intuitività dovuta alle varie strutture di supporto (menu, tendine, dialog box, ecc.), che presentano le varie opzioni possibili momento per momento, va fatto un discorso più generale. Il Report è ormai una funzionalità non di stampa, pura e sem-

plice, dei dati ma una funzionalità cui delegare buona parte delle necessità elaborative di una procedura.

Se nel Report si possono inserire campi calcolati, si possono definire e utilizzare variabili, si possono utilizzare campi presi mediante dei collegamenti dinamici da altri archivi (operazione che da parte dell'utente si risolve molto semplicemente nell'indicare in quale archivio prendere il nuovo dato, ma che il programma esegue facendo ricorso alle proprie prestazioni relazionali), allora significa che, per problematiche di semplice o media complessità, non si deve ricorrere alla programmazione.

In caso di necessità elaborative più pesanti si può suddividere il carico di lavoro tra il Report e il Query, in quanto con quest'ultimo si può già eseguire una preelaborazione, raccogliendo i dati dai vari archivi e sottoponendoli già a dei calcoli.

Sia in dBASE IV che in Paradox, una operazione di Query, che può lavorare su più archivi, produce in uscita un archivio, che può essere virtuale o reale, e che può essere utilizzato in qualsiasi

Pagina n. 1

18/06/89

		CLIENTE		VENDITORE	
		CODICE NOMINATIVO		CODICE NOMINATIVO	
ORDINE N.1	DEL. 01/01/89	C001	PERKINS	V002	ROSSI
articolo descriz.	quantita'	costo unit.		importo	
A001 GOMMA	1	300		300	
A002 PENNA	12	1,000		12,000	
A004 MATITA	4	500		2,000	
<b>TOTALE IMPORTO DELL'ORDINE</b>		<b>SCONTO</b>	<b>20</b>	<b>14,300</b>	
		<b>IMPORTO SCONTATO</b>		<b>11,400</b>	

ORDINE N.2	DEL. 02/01/89	C002	JOHNSON	V003	BIANCHI
articolo descriz.	quantita'	costo unit.		importo	
A002 PENNA	5	1,000		5,000	
A004 MATITA	5	500		2,500	
<b>TOTALE IMPORTO DELL'ORDINE</b>		<b>SCONTO</b>	<b>15</b>	<b>7,500</b>	
		<b>IMPORTO SCONTATO</b>		<b>6,375</b>	

ORDINE N.3	DEL. 03/01/89	C002	JOHNSON	V001	VERDI
------------	---------------	------	---------	------	-------

Cancel viewing: ESC, Continue viewing: SPACCELA

Figura 8 - dBASE IV Stampa a video. Come detto già altre volte il vecchio Report Generator del dBASE III... è dimenticato. Il nuovo permette tre forme di Layout standard, quella a forma di classico tabulato, quella a forma di scheda e infine il tipo Mailmerge che permette di scrivere un testo inserendovi campi dell'archivio o campi calcolati.

si altra funzionalità che lavori su un archivio.

Per quanto riguarda il dBASE IV, va detto che viene abbandonato il vecchio e superato Report Generator del dBASE III per uno nuovo di zecca, che permette più tipologie di default (Tabulare, Scheda e Lettera), permette un maggior controllo strutturale, un più spinto utilizzo delle funzionalità di calcolo e, infine, e anche questo non guasta, un miglior utilizzo ai fini estetici delle stampanti.

In ambedue i prodotti il Report, parliamo del più diffuso che è quello in forma

tabulare, presenta delle «fette orizzontali» (in dBASE si chiamano BAND, termine che in Paradox è usato invece per le colonne verticali), che rappresentano le varie zone strutturali classiche in un tabulato.

Al centro c'è la zona del dettaglio, dove vanno posizionati le intestazione dei vari campi e i campi stessi. Il posizionamento si esegue con l'editor nell'ambiente Full Screen, e può essere seguito, campo per campo, dalla definizione degli aspetti estetici, come Pictur, se si tratta di numeri o allineamenti,

attributi dei caratteri, se si tratta di Stringhe.

Il dBASE IV, permette anche di definire anche gli attributi estetici in stampa, in funzione della stampante installata, nonché di memorizzare, in un file specifico, l'insieme dei settaggi della stampante come il numero di righe, di colonne, margini, caratteri di controllo. In altre parole tutto ciò che non riguarda il contenuto del Report.

Le altre zone strutturali del Report (fette orizzontali) sono i titoli generali, le intestazioni in cima e in fondo alla pagina (testatine e piedini), le zone di raggruppamento, che può essere eseguito su più livelli.

Anche nella definizione di queste fette orizzontali si lavora in modalità Full Screen, e si possono inserire scritte fisse, nomi dei campi, ad esempio all'inizio e alla fine di un raggruppamento, numeri di pagina, data e orario, ad esempio a fine pagina, ecc.

Ambedue i prodotti utilizzano il concetto di Summary, ovvero la possibilità di posizionare nelle varie zone un campo (anche un campo calcolato) che viene totalizzato, oppure solo contato, o mediato, ecc. (si ottengono totali di gruppo, di pagina, totali generali, ecc.).

Nelle varie illustrazioni (da 5 a 9), ampiamente commentate, vediamo fasi di lavoro all'interno dei due Generatori e uscite sia sul video che su stampa.

Figura 9 - dBASE IV Stampa dal generatore di Report. Abbiamo voluto provare le nuove possibilità di intervenire sull'estetica di stampa a livello di singolo campo. Questa stampa è stata realizzata con una semplice stampante ad aghi, sulla quale sono stati impostati differenti attributi estetici per i differenti elementi in stampa.

Pagina n. 1

18/06/89

		CLIENTE		VENDITORE	
		CODICE NOMINATIVO		CODICE NOMINATIVO	
ORDINE N.1	DEL. 01/01/89	C001	PERKINS	V002	ROSSI
articolo descriz.	quantita'	costo unit.		importo	
A001 GOMMA	1	300		300	
A002 PENNA	12	1,000		12,000	
A004 MATITA	4	500		2,000	
<b>TOTALE IMPORTO DELL'ORDINE</b>		<b>SCONTO</b>	<b>20</b>	<b>14,300</b>	
		<b>IMPORTO SCONTATO</b>		<b>11,440</b>	

ORDINE N.2	DEL. 02/01/89	C002	JOHNSON	V003	BIANCHI
articolo descriz.	quantita'	costo unit.		importo	
A002 PENNA	5	1,000		5,000	
A004 MATITA	5	500		2,500	
<b>TOTALE IMPORTO DELL'ORDINE</b>		<b>SCONTO</b>	<b>15</b>	<b>7,500</b>	
		<b>IMPORTO SCONTATO</b>		<b>6,375</b>	

ORDINE N.3	DEL. 03/01/89	C002	JOHNSON	V001	VERDI
articolo descriz.	quantita'	costo unit.		importo	
A001 GOMMA	10	300		3,000	
A002 PENNA	2	1,000		2,000	
A003 QUADERNO	12	1,500		18,000	
<b>TOTALE IMPORTO DELL'ORDINE</b>		<b>SCONTO</b>	<b>15</b>	<b>23,000</b>	
		<b>IMPORTO SCONTATO</b>		<b>19,550</b>	

### Conclusioni

È indubbio che mentre nei prodotti di tipo Spreadsheet o di tipo Word Processor si è raggiunto un livello di perfezione tale da rendere problematiche e quindi costosissime ulteriori evoluzioni. Nei prodotti di tipo DBMS, al contrario, esistono ancora margini di possibile evoluzione qualitativa: sia in termini di aumento della facilità d'uso, e i vari generatori ne sono un esempio evidente; sia in termini di potenza elaborativa, soprattutto sotto l'aspetto relazionale.

E anche questo, come emerso nei vari articoli sull'argomento, risulta evidente nelle ultime versioni di dBASE e Paradox. Altro aspetto confortante è l'unificazione dei linguaggi, il che rende abbastanza praticabile una conversione di un'applicazione o di un utilizzatore da uno all'altro, e, per i tecnici, la padronanza di ambedue i prodotti.

La cosa migliore sarebbe, ovviamente, una unificazione totale dei linguaggi, raggiunta come accordo tra i vari produttori e non solo come esigenza strategica o di mercato. È sperabile che piano piano ci si arrivi.



# POWER & QUALITY

ROBERTO  
RUBINO

**Personal computer UNIDATA, al vertice delle prestazioni e della qualità in una gamma completa e flessibile.**

## Serie PX, Personal computer da tavolo:

**PX 3000**, CPU NEC V20 12MHz, floppy disk 3,5/5,25 pollici, hard disk da 20 o 40MB.

**PX6000**, CPU 80286 13 o 18MHz, ram da 512K a 8MB, floppy disk 3,5/5,25 pollici, hard disk da 20 a 330MB tipo MFM, RLL, ESDI.

**PX7000**, CPU 80386SX 16MHz, floppy disk 3,5/5,25 pollici, hard disk da 40 a 330MB tipo MFM, RLL, ESDI.



## Serie AX, Super Personal computer tipo Tower:

**AX6000**, CPU 80286 18MHz, ram da 512K a 8MB, floppy disk 3,5/5,25 pollici, hard disk da 40MB a 1,5 GB, tipo MFM, RLL, ESDI, SCSI, interleave 1:1.

**AX7000**, CPU 80386SX 16MHz con cache memory, ram da 1MB a 8MB, floppy disk 3,5/5,25 pollici, hard disk da 60MB a 1,5GB tipo MFM, RLL, ESDI, SCSI, interleave 1:1.

**AX8000**, CPU 80386 20/25MHz con cache memory, ram da 1 a 16MB, floppy disk 3,5/5,25 pollici, hard disk da 60MB a 1,5GB tipo RLL, ESDI, SCSI, interleave 1:1.



**LAN Ethernet**, 10 Mbit secondo, cavo coassiale o doppio telefonico.

**LAN Server**, 16/32 bit CPU 18-25MHz, Hard disk 90-1,5 GB disk ram cache.

**LAN WORKSTATION**, CPU 8088/ 80286/ 80386SX, diskless station con Ethernet.

**VGA** adapter con risoluzione 640x480 o 1024x768.

**Grafica ad alta velocità** per applicazioni CAD, modello MX velocità di tracciamento di 30.000 vettori al secondo e HX con velocità di 100.000 vettori al secondo.

**Video Monitor ad alta risoluzione** per applicazioni CAD e Image Processing.

*Ethernet è marchio della Xerox e Digital Equipment Corporation.*

