

Rimbocchiamoci le maniche

di Francesco Petroni

Gli ultimi sei mesi sono stati molto importanti nella storia del Personal Computing. Sono apparsi, in questo breve volgere di tempo, una serie di prodotti della Microsoft che hanno delineato, in maniera probabilmente definitiva, il futuro nel mondo del software per P.C. Sono usciti, in rapida successione, Word per Windows, Windows 3.0, PowerPoint per Windows, preceduti da Project per Windows (cito anche questo prodotto più specialistico, ma pur sempre significativo della tendenza). E, per la cronaca, dopo poco, sono apparse anche le versioni in italiano. In particolare la risonanza che ha avuto il lancio di Windows 3.0 è stata senza precedenti. E anche il successo di vendite è stato senza precedenti

Nella bellissima videocassetta che ha accompagnato il tour di Bill Gates, tenuto in occasione del lancio di Windows 3.0 in italiano, sono state inserite una serie di interviste ad alcuni responsabili delle principali case di software, Borland e Lotus incluse, dalle quali appare evidente che anche queste case si addegueranno, nel prossimo futuro, a tale interfaccia. Ulteriori annunci «bilaterali» segnalano altresì il perfetto allineamento tra Microsoft ed IBM sulle strategie software, Windows e Presentation Manager compresi.

Questo significa che Windows 3.0 rappresenterà la piattaforma unica per tutti i prodotti software più significativi che saranno sviluppati nei prossimi anni.

È probabile quindi che con Windows 3.0 si apra una nuova generazione nel mondo della Informatica Individuale. E forse l'atteggiamento da assumere più corretto è proprio quello di considerare

Windows 3.0 direttamente come il primo momento di una nuova era. In tale ottica «realistica» i giudizi sui singoli prodotti vanno rivisti e i confronti vanno calibrati.

Una nuova rubrica: la rubrica Windows 3

La nostra rivista, al contrario delle altre, e in più rispetto alle altre, è caratterizzata dalla presenza delle rubriche. Ognuna di queste ha un proprio titolo che ne specifica l'argomento.

Nelle varie rubriche vengono trattati svariati temi, da quelli di «cultura generale» informatica, che prescindono quindi dai prodotti, fino a quelli tecnici avanzati, che invece esplorano aspetti particolari ed evoluti di un prodotto o di una macchina. In altre rubriche vengono condotti degli approfondimenti di problematiche solo sfiorate, per motivi di spazio, nelle tradizionali prove dei prodotti.

Iniziamo, con questo numero, una nuova rubrica dedicata a Windows 3 il cui scopo è appunto quello di seguire da vicino le evoluzioni di tale ambiente e soprattutto del software, di vario genere, dal semplice applicativo al sofisticato toolkit per programmatori, che, sempre più numeroso, ne accetta le regole operative.

In questo primo numero riassumeremo alcuni dei punti fermi su cui si poggia la «filosofia Windows» e che invece, a partire dal prossimo numero, daremo per scontati.

Altro dato che daremo per scontato è l'accettazione di Windows e Presentation Manager come interfaccia grafica standard per i sistemi basati sul DOS e, nel futuro, sull'OS/2.

Non faremo quindi, per lo meno in questa sede, confronti tra prodotti GUI (Graphical User Interface) e prodotti CUI (Character User Interface), dando per noto il fatto che l'adozione della interfaccia grafica comporta molti vantaggi e qualche svantaggio, ormai citati «in letteratura», e sui quali quindi è inutile continuare a discutere.

Il nostro atteggiamento è in conclusione questo. Windows 3 ci accompagnerà, ed accompagnerà i nostri figli (fig. 1), per i prossimi 10 o 15 anni. In

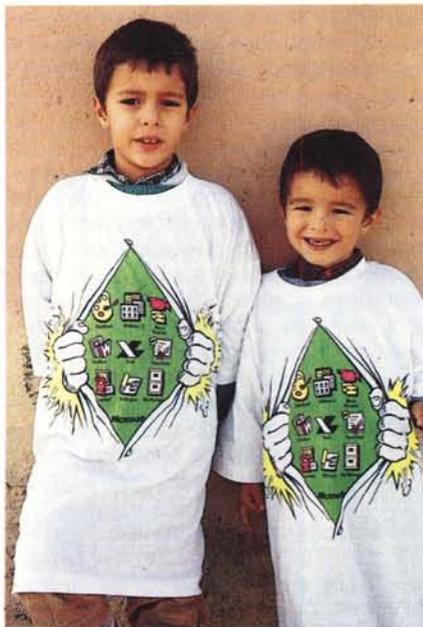


Figura 1 - Microsoft Windows 3 - Il sistema operativo dei prossimi 10,15 anni. I nostri figli, questi sono i miei Paolo e Giovanni, con la T-shirt Windows 3, utilizzeranno sicuramente l'interfaccia grafica Windows, che oggi, nel 1990, va considerata ancora nel momento iniziale, quasi pionieristico, e che ci accompagnerà, e accompagnerà gli utilizzatori futuri, per un bel pezzo.

questi anni ci saranno notevoli evoluzioni sia nell'hardware, che sarà sempre più veloce e capace, in termini di RAM, sia del software, Windows e per Windows, che sarà sempre più perfezionato ed affidabile.

Inoltre ci sarà molta concorrenza tra le varie case software, che ormai, essendosi tutte, senza eccezioni, adeguate allo standard Windows, faranno uscire numerosi altri prodotti per Windows, dapprima adattando per l'interfaccia grafica i loro vecchi pacchetti, poi, in seguito, sfornando altre novità, la cui portata è difficile oggi prevedere.

E la concorrenza in pratica fino ad ora non c'è stata, per il semplice fatto che, fino ad ora, solo la Microsoft e poche altre case, hanno supportato Windows con prodotti di prestazioni analoghe a quelle dei corrispondenti prodotti CUI.

Il titolo «rimbocchiamoci le maniche» sta proprio a significare il fatto che saranno tanti gli argomenti da trattare, tanti i prodotti da provare (cosa che continueremo a fare con le tradizionali prove) e da approfondire. Ci rimbocchiamo le maniche volentieri sia perché a noi, come a voi, questa tecnologia piace, sia perché dopo qualche anno di quasi immobilità il mondo del software si è rimesso in moto e noi non ci vogliamo né possiamo far trovare impreparati.

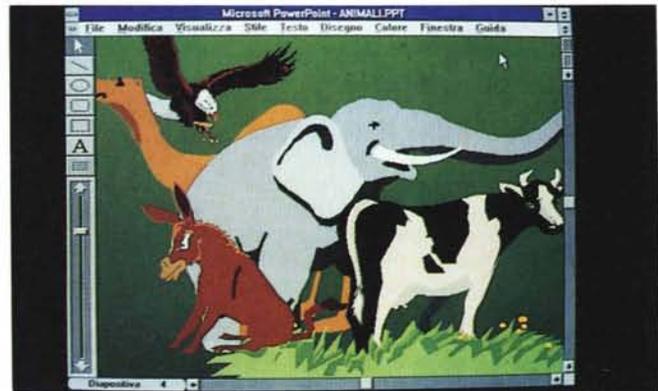
Schematizzazione dei rapporti tra Windows e DOS

In figura 3 abbiamo cercato di esemplificare, in un disegno molto schematico, la posizione di Windows, collocata tra il sistema operativo e l'applicativo, e di elencare per sommi capi le funzioni che gli possono venire delegate. Come al solito facciamo un discorso elementare per farci capire da tutti, anche da quegli utenti, nuovi arrivati, cui capiterà di utilizzare direttamente Windows. Nei prossimi numeri della rubrica invece tratteremo anche argomenti più tecnici.

Fino alla versione 2.11 Windows poteva essere considerato un applicativo DOS, che valeva la pena caricare solo se si pensava di utilizzare un applicativo sotto Windows, che all'epoca erano molto pochi. Il risultato è che, all'epoca, pochissimi hanno adottato Windows, e quei pochi che l'hanno installato l'hanno

Figura 2 - Microsoft PowerPoint per Windows 3 - Operazione di taglia e incolla

Questa figura è stata realizzata, accorpando vari oggetti di un file Clip-Art di PowerPoint, da mio figlio Paolo, di sei anni, al quale, dietro sua insistenza, ho insegnato a fare il Taglia ed Aggiungi sotto Windows e lo scaling degli oggetti selezionati. Questo a conferma della assoluta intuitività di buona parte dei comandi disponibili nell'ambiente Windows.



I SERVIZI E LE FUNZIONI SVOLTI DA WINDOWS 3



Figura 3 - Schematizzazione del funzionamento di Windows. Volendo semplificare al massimo il discorso potremmo affermare che Windows si pone tra il DOS, del quale si assume tutti i compiti, e i vari applicativi (i cosiddetti applicativi «sotto Windows») per i quali svolge numerosi servizi. Con la versione 3 sono stati introdotti anche ulteriori servizi per la gestione del sistema, che ormai può essere affidata completamente a Windows.

fatto più che altro per curiosità.

Oggi il discorso cambia per due motivi. Il primo è che si cominciano a vedere un bel po' di applicativi, di tutte le categorie, e quindi Word Processor, Spreadsheet, DBMS, ecc. e di tutte le classi economiche.

Si cominciano anche a vedere Tools di sviluppo per chi vuol programmare sotto Windows. E anche in questo caso ci troviamo di fronte a varie categorie di prodotti, per utenti finali, per utenti evoluti e per specialisti (figg. 4, 5, 6 e appendice).

Il secondo motivo è che Windows 3 dispone di un bel po' di funzioni di ser-

vizio che permettono sia di eseguire tutti i comandi DOS, sia di gestire in maniera semplice ed economica l'organizzazione di tutto il sistema.

Gestire il sistema attraverso Windows

Una innovazione fondamentale di Windows 3, rispetto alle versioni precedenti, è costituita dalla possibilità di generare delle finestre di tipo GRUPPO, e di collocare all'interno di queste delle ICONE che richiamano le singole applicazioni. In questa maniera è possibile gestire attraverso Windows tutto il si-

stema, creando una organizzazione delle applicazioni del tutto indipendente da quella del DOS.

Il Gruppo è una finestra a tutti gli effetti che può essere posizionata sul video alla solita maniera (fig. 7). La costruzione dei Gruppi e l'inserimento nell'interno di questi delle Applicazioni si esegue attraverso il comando di Menu FILE ed è una operazione semplicissima, tutta eseguibile via mouse.

Vengono richiesti, nome dell'applicazione, percorso e nome del file, che sono inseribili percorrendo l'albero delle directory direttamente con il mouse, e l'Icona. Le applicazioni inseribili nei Gruppi e quindi gestibili via Windows sono non solo ovviamente quelle realizzate per Windows, ma anche tutte le altre.

Le prime hanno, nel proprio interno una Icona, che appare «naturalmente», quando si inserisce l'applicazione nel gruppo.

Per le seconde Windows mette a disposizione una serie di Icone standard, che indicano, più o meno, la tipologia dell'applicazione.

È possibile anche «personalizzare» le Icone, in quanto esistono già utility create allo scopo (fig. 8).

Per regolare meglio i rapporti tra Windows e gli applicativi non sviluppati per lavorare sotto Windows, esiste, come noto, l'entità PIF, che è in pratica un piccolo file che contiene le specifiche hardware che Windows deve garantire all'applicativo sottostante (fig. 9).

La costruzione del file PIF non è molto semplice per il fatto che le variabili in gioco, gestibili via PIF, sono moltissime. Sarebbe preferibile che qualcuno, o Windows stesso come faceva nella vecchia versione 2, oppure l'applicativo in questione, fornisse direttamente il file PIF già confezionato al meglio.

È interessante infine notare che anche le case tradizionalmente concorrenti della Microsoft si stanno convertendo. È significativo ad esempio, che la stessa Lotus sul package del suo recente 123 Release 3.1, garantisce la compatibilità con Windows 3. Ma di questo parliamo nella prova dell'123 Rel.3.1 presentata in questo stesso numero.

Non potevamo certo pretendere che la Lotus sfornasse istantaneamente versioni dei suoi prodotti tradotti per Windows. Per ora l'123 delega a Windows la gestione della memoria, e non la gestione dell'interfaccia grafica. Non è difficile immaginare che con successive versioni ci sarà... l'aggancio totale.

In realtà non è obbligatorio né costrui-



Figura 4 - Microsoft - Windows Excel - Dialog Box Editor. Un primo esempio di generatore di «oggetti» Windows, è il Dialog Box Editor rilasciato, come accessorio, insieme ad Excel. In pratica serve per confezionare, operando, con modalità full screen, in un ambiente Editor di Windows, una finestra di dialogo personalizzata, utilizzabile al posto di quelle standard, presenti nelle funzionalità interattive o in quelle Macro di Excel.

Figura 5 - Asymetrix Toolbook - Menu a icone. Al pacchetto Windows 3, versione inglese, era allegato un dimostrativo del Toolbook della Asymetrix, che ci sembra significativo di un filone di prodotti che si diffonderanno moltissimo e del quale, è in preparazione una prova. In pratica si tratta di un generatore di applicazioni sotto Windows.

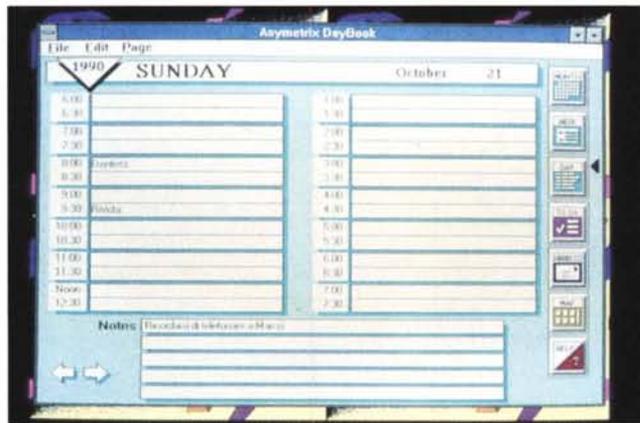
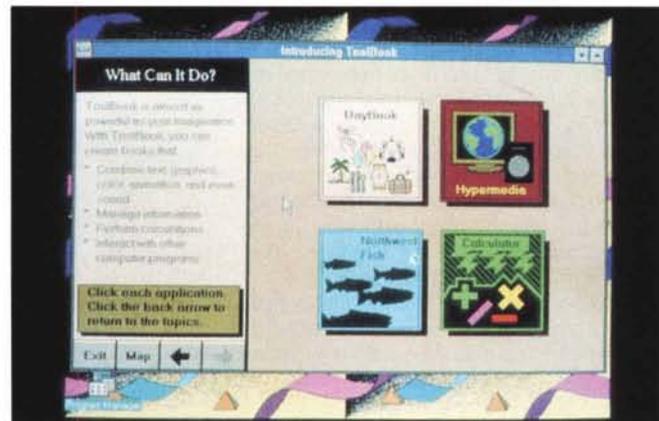
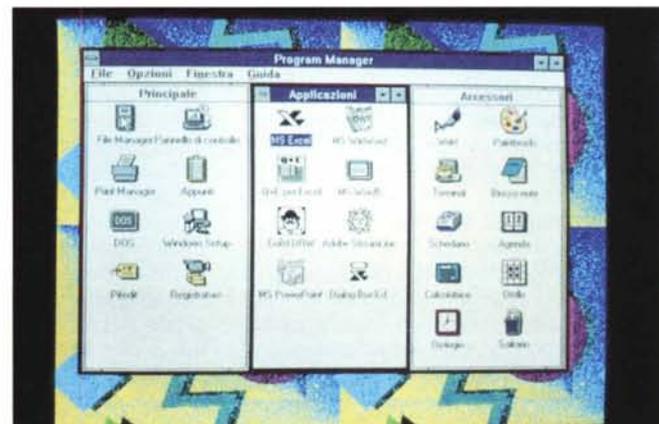


Figura 6 - Asymetrix Toolbok - Applicativo di tipo agenda personale. La collocazione del Toolbook, destinato agli utilizzatori finali evoluti, è intermedia tra il Windows 3 Toolkit, che permette, ai programmatori, l'utilizzo della interfaccia e dei servizi di Windows in applicazioni scritte con linguaggi tradizionali, come il C, e le varie funzionalità Macro, presenti nei vari applicativi.

Figura 7 - Microsoft - Windows 3 - File gruppi ed applicazioni. Una innovazione fondamentale di Windows 3, è costituita dalla possibilità di generare delle finestre di tipo GRUPPO, e di collocare all'interno di queste delle ICONE per richiamare le singole applicazioni. In questa maniera è possibile gestire, attraverso Windows, tutto il sistema, creando una organizzazione delle applicazioni del tutto indipendente da quella del DOS.



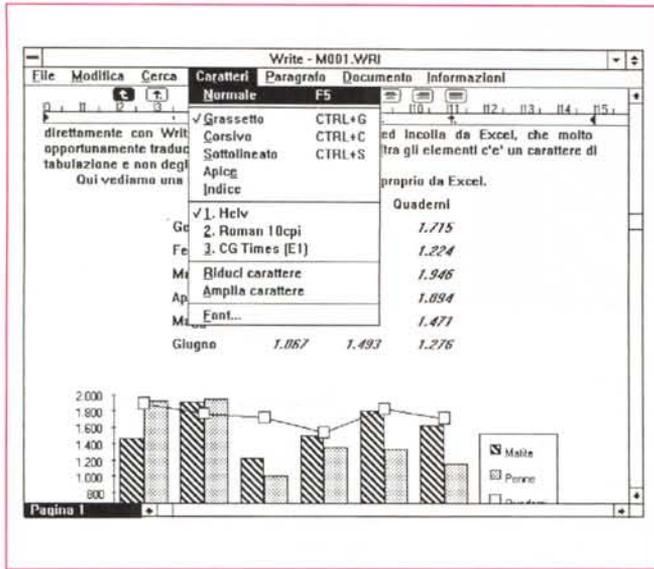


Figura 12 - Microsoft Write per Windows 3 - Esempio di utilizzazione 1. Write, fornito in dotazione con Windows 3, è il più elementare dei Word Processor, ma dispone sia di un ambiente operativo ad Icone, ripreso successivamente dal fratello maggiore WORD per WINDOWS, sia un aspetto «quasi WYSIWYG». Nè è una prova questa hard copy del video. Questo fatto, unito alla bontà dei Driver per la gestione delle periferiche, apre Windows Write ad un discreto range di utilizzazioni.

Figura 13 - Microsoft Write per Windows 3 - Esempio di utilizzazione 2. Ed ecco qui in uscita una pagina realizzata con Windows Write e stampata con una LaserJet III. Nel collage abbiamo inserito elementi tabellari, di provenienza Excel ed elementi grafici, di provenienza Excel, a dimostrazione dell'ottima integrazione di tutto l'ambiente.

fine deve conoscere il significato del comando che esegue, sia che lo richiami dal prompt del DOS sia che lo richiami dalla tendina del menu.

Win Write — Cosa gli manca per essere il vostro Word Processor

Trattandosi di un capitoletto di mezza pagina abbiamo preferito scriverlo direttamente con Windows Write. Per cui rimandiamo all'inserto che si autocommenta. In figura 12 al lavoro con Write e in figura 13 il risultato su carta.

Da notare che la figura 12 è una hard copy del video, e che assomiglia molto alla figura 13, che è invece la stampa vera e propria, a conferma del rispetto della filosofia WYSIWYG.

A cosa serve e come funziona il File WIN.INI

Chi conosce abbastanza bene il DOS sa che esiste un filetto (si legge failletto) che si chiama CONFIG.SYS in cui vanno impostate alcune specifiche di funzionamento del sistema. In particolare si inseriscono nel CONFIG.SYS tutte le informazioni relative al collegamento con

«pezzi» hardware non standard. In Windows 3 (ma anche nel vecchio 2.xx) esiste, in più rispetto al CONFIG.SYS, anche il file WIN.INI in cui sono memorizzate tutte le caratteristiche di funzionamento del Windows e degli applicativi che via via vengono installati sotto Windows stesso. Essendo un file testuale è abbastanza facile (ed istruttivo) leggerlo, anche con il Blocco Notes o con il Write di Windows stesso.

Sotto OS/2 Presentation Manager il WIN.INI confluisce in pratica nel CONFIG.SYS a conferma del fatto che S.O. e interfaccia grafica, che sotto OS/2 si chiama Presentation Manager, sono considerate come un unico oggetto.

Il file WIN.INI viene generato in fase di prima installazione di Windows e viene modificato ogni volta che si cambia la configurazione (ad esempio per aggiungere una Stampante) o si aggiunge un applicativo. In questo caso WIN.INI memorizza la disposizione dell'applicativo, memorizza la configurazione e il nome dei vari programmi accessori, aggiunti all'applicativo, ecc.

Durante l'installazione dell'applicativo viene in genere alimentata anche la directory SYSTEM generata da Windows,

WRITE PER WINDOWS PUO' ESSERE IL VOSTRO W.P. ?

Questa paginetta è stata interamente scritta con l'applicativo Write, presente in dotazione con Windows, ed è stata realizzata allo scopo di dimostrare che Windows 3, anche da solo, è utilizzabile con profitto, da un utente alle prime armi. La pagina è stilisticamente un po' «incasinata» per verificare anche le possibilità in termini di funzionalità e di estetica disponibili con Write.

Lavorare comunque sotto Windows, anche su applicativi semplici, può essere considerato un approccio valido, in quanto buona parte delle funzionalità che si sperimentano sono del tutto analoghe a quelle dei prodotti più evoluti.

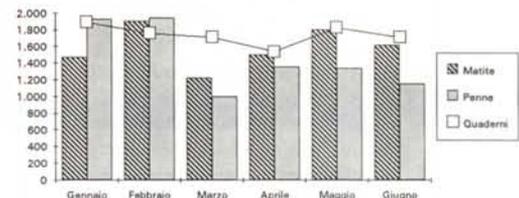
Ad esempio sono moltissime le analogie tra il modesto WRITE e il sofisticato WinWord, sia in termini di organizzazione dei comandi operativi, ad esempio quelli attivabili via Barra del Menù e via Icone.

L'utente lavora in un ambiente già (abbastanza) WYSIWYG, in cui può inserire e manipolare anche figure importate via Clipboard. Comincia ad imparare il concetto di Font, di Attributi, di Giustificazione, di Interlinea, con le Testatine ed i Piedini, ecc.

Può anche cominciare a muoversi nel terreno minato delle tabellazioni, sia lavorando direttamente con Write, sia eseguendo un copia ed incolla da Excel, che molto opportunamente traduce i dati in un formato tabellare (ha gli elementi c'è un carattere di tabulazione e non degli inutili spazi).

Qui vediamo una tabellina e un grafico importati proprio da Excel.

	Matite	Penne	Quaderni
Gennaio	1.856	1.506	1.715
Febbraio	1.045	1.001	1.224
Marzo	1.301	1.762	1.946
Aprile	1.491	1.910	1.894
Maggio	1.179	1.926	1.471
Giugno	1.067	1.493	1.276



Microsoft Windows 3 e il Software Development Kit

di Claudio Emanuele e Susanna Benedetti

A pochi mesi di distanza dalla presentazione di Windows 3 da parte della MICROSOFT e, visto l'enorme successo di «pubblico e di critica» ottenuto da tale ambiente, ci sembra doveroso rivolgere la nostra attenzione anche ai tecnici dell'informatica, illustrando per sommi capi il prodotto Microsoft Software Development Kit (SDK), che diventerà sicuramente, e in breve tempo, uno standard per la realizzazione di applicazioni su PC e contemporaneamente un buon trampolino di lancio verso la programmazione in ambiente OS/2 Presentation Manager.

L'SDK è un tool di sviluppo che mette a disposizione del progettista software un insieme di funzioni (circa 500), alcuni file d'installazione, contenenti le dichiarazioni delle funzioni e delle strutture di dati da queste utilizzate, tre debugger diversi e un linker, sui quali basare la realizzazione di applicazioni in ambiente Windows. L'ambiente di sviluppo per queste applicazioni dovrà essere costituito quindi dall'SDK, da un linguaggio di programmazione tradizionale, ad esempio Assembler, Pascal o C, da un editor e da Windows stesso.

Per ovvie considerazioni di diffusione, standard e riutilizzabilità sulla piattaforma OS/2 PM, il linguaggio più indicato per «programmare in ambiente Windows» risulta essere il C, ed è a questo che faremo riferimento.

La programmazione in questo ambiente sarà una programmazione di tipo «object-oriented» e seguirà le regole e le specifiche di quest'ultima. Il che, in parole povere, significa che un oggetto è definibile come un insieme di elementi che, una volta associati fra loro, possono essere considerati come una entità unica.

Ad esempio in Windows un oggetto è la WINDOW, costituita da una struttura dati e da un insieme di funzioni che operano su questa struttura. Altri oggetti, correlati dell'oggetto principale WINDOW, sono gli elementi del menu, le barre di scorrimento, ed ogni altro particolare associato al controllo della finestra. Utilizzando una semplice chiamata a Windows si viene in possesso di un oggetto (ad es. la finestra principale dell'applicazione) e le consuete operazioni che si eseguono su tale oggetto (ad es. lo spostamento, il ridimensionamento, ecc...) agiscono su tutti gli elementi associati all'oggetto stesso.

Ciascun oggetto, per le regole «isomorfiche» della programmazione object-oriented, sarà identificato univocamente da un nome riservato, chiamato HANDLE (traducibile in maniglia, appiglio), in modo tale da poter far riferimento ad esso mediante l'handle stesso. In Windows un handle è

un valore univoco di 16 bit (unsigned short del C), non è mai collegato con specifiche posizioni nella memoria, è assegnato da Windows e può cambiare in ogni esecuzione del programma. Questo intero non ha alcun significato se non per il kernel stesso di Windows (il cuore di Windows 3 è il programma KERNEL.EXE).

Come sicuramente i nostri lettori già sapranno, Windows realizza, in modo preemptive, una sorta di multitasking che consente l'esecuzione di più applicazioni contemporaneamente, ed in particolare più copie del medesimo programma.

Per questo motivo, le applicazioni Windows devono conservare informazioni relative alle diverse occorrenze (in gergo «instance») di se stesse, dove per occorrenza si intende l'handle di Windows associato alla copia dell'applicazione che si sta usando in un determinato momento.

Conservando tali informazioni è possibile ottimizzare la gestione della memoria. Infatti quando vengono eseguite due copie della stessa applicazione, Windows mantiene una sola copia dei segmenti di codice, e assegna invece segmenti multipli a quelli dei dati.

Altri concetti tipici della programmazione in Windows sono gli «eventi» e i «messaggi». Questi sono gli unici mezzi a disposizione di Windows per scambiare informazioni con le applicazioni.

Un evento è un qualcosa che accade e che influenza gli oggetti ad esso correlati. In Windows, ad esempio, un evento è causato dallo spostamento del mouse o dalla pressione di un bottone.

Windows si accorge dell'evento ed invia un messaggio all'applicazione nella quale l'oggetto (la finestra) è stata selezionata. I messaggi sono quindi strettamente collegati all'oggetto influenzato dall'evento.

Ad esempio, supponiamo di selezionare una opzione di un menu pull-down. Windows invierà un messaggio contenente l'handle associato alla finestra che contiene il menu selezionato in quanto considera il menu come oggetto collegato alla finestra.

Di conseguenza, un evento è collegato ad un messaggio, il messaggio è collegato ad un handle ed un handle è collegato ad un oggetto. Programmare in questo ambiente, non lo nascondiamo, equivale ad assumere un nuovo modo di «vedere» le cose, e inizialmente richiederà sicuramente un notevole sforzo da parte del programmatore, pur rimanendo un affascinante «sfida» nel campo delle innovazioni.

Mostriamo, a titolo di esempio, qualche riga di un programma, che non fa «niente», comparandolo con un equivalente programma nullafacente scritto semplicemente in

C. Chi già conosce le regole del C sa che la prima funzione in un qualsiasi programma è la funzione main(), per cui un programma di tal genere sarà costituito dalle seguenti istruzioni:

```
/* niente.c*/
main()
{
}
```

In Windows esiste una funzione equivalente: la WinMain() e lo stesso programma verrà così scritto:

```
/* niente.c*/
#include "windows.h"
int PASCAL WinMain
( hInstance, hPrevInstance,
  lpszCmdLine, nCmdShow)
HANDLE hInstance, hPrevInstance;
LPSTR pszCmdLine;
int nCmdShow;
{
return FALSE;
}
```

Dove la prima istruzione è una istruzione di inclusione per il file windows.h e deve essere inserita in ogni programma Windows.

È necessaria per le definizioni di PASCAL, HANDLE, LPSTR e FALSE, non riconosciute dal linguaggio C. WinMain() è il punto di inizio del programma ed è equivalente alla funzione main() del C.

Windows inizia l'esecuzione del programma in corrispondenza di questa funzione e le passa quattro diversi parametri: **hInstance** che rappresenta l'handle dell'occorrenza dell'applicazione;

hPrevInstance che è l'handle relativo a qualsiasi altra occorrenza precedente all'applicazione;

lpszCmdLine che è un puntatore ad una stringa, terminante con NULL, nel cui interno è contenuta la linea di comandi successiva al nome dell'applicazione. Se per esempio, l'utente attivasse il programma, chiamato con il nome «NIENTE», utilizzando l'opzione File Esegui di Windows e digitasse «NIENTE» seguito da una qualsiasi stringa, ad esempio «di niente», lpszCmdLine punterebbe alla stringa «di niente», che, nel nostro caso, non verrebbe interpretata. In altri casi, ad esempio in WinWord, verrebbe usata per indicare il nome del file da aprire insieme alla applicazione.

nCmdShow è un intero riguardante il modo in cui Windows visualizzerà, in fase di lancio, l'applicazione.

Da questo semplice esempio, si comprende quali siano le diverse modalità di approccio e il diverso grado di difficoltà nel-