

Stampanti

di Sergio Polini

Uno degli aspetti più comodi della programmazione sotto Windows è rappresentato dalla indipendenza dal tipo dei dispositivi di output che il programma si troverà ad usare. Si può dimenticare di ragionare in termini di video CGA, Hercules, EGA o VGA. Si può soprattutto fare a meno di preoccuparsi della stampante che userà l'utente del programma. Pensate ad un qualsiasi programma di trattamento testi sotto MS-DOS, alla folta schiera dei driver per stampanti che inesorabilmente lo accompagna; driver che servono a qualcosa solo con quel programma e sono del tutto inutili con un altro. Chiunque abbia realizzato un'applicazione professionale sotto MS-DOS ha dovuto affrontare tali problemi, scegliendo o di adattarsi alla stampante del suo cliente, o di imporgli uno o due particolari tipi di stampante, o di perdersi nella preparazione di un'infinità di driver. Se sono certo inadeguate le prime due soluzioni, la terza aumenta sensibilmente tempi e costi di programmazione

Molto più semplice la vita sotto Windows. Almeno in linea di principio. Windows ha infatti i suoi driver, per un gran numero di stampanti; se compare sul mercato una stampante nuova, questa sarà dotata del suo driver per Windows, diventando così, a tutti gli effetti, una risorsa liberamente accessibile come quelle già previste.

Le cose non sono in realtà del tutto automatiche, in quanto il programma dovrà preoccuparsi di verificare alcune caratteristiche e le effettive potenzialità sia di una data stampante che del suo driver, ma potrà farlo con tecniche che, pure se di uso non facilissimo, saranno almeno sempre le stesse. Si tratterà quindi semplicemente di seguire gli stessi passi, quale che sia la stampante di volta in volta scelta dall'utente; la OOP, inoltre, ci può aiutare ad incapsulare i dettagli delle operazioni in una o due classi facilmente utilizzabili in qualsiasi programma. Come vedremo tra qualche tempo. Per ora, ci proporremo in primo luogo di dotare un'applicazione Turbo Vision, quindi MS-DOS, di una flessibilità quanto più possibile vicina a quella di cui si gode sotto Windows, almeno per stampe di testi. Per raggiun-

gere un tale obiettivo, analizzeremo per prima cosa i diversi aspetti dell'utilizzo di una stampante sotto Windows, per mettere bene a fuoco i diversi momenti della soluzione cui tendiamo.

Installazione di una stampante

Partiamo da zero, da un sistema Windows senza alcuna stampante installata. Quando si voglia iniziare ad usare una data stampante, si sceglie l'opzione *Stampanti* del Pannello di Controllo. Si apre quindi una dialog box che, tra l'altro, comprende un pulsante *Aggiungi stampante*, premendo il quale la dialog box si allunga fino a comprendere una list box con cui si può percorrere un elenco molto lungo di periferiche delle marche e dei tipi più disparati (figura 1).

Selezionata una stampante, va premuto il pulsante *Installa*; Windows chiede a questo punto l'inserimento del dischetto contenente il driver, che provvede a copiare sul disco rigido. Si può quindi proseguire con il pulsante *Configurazione*, con il quale si accede ad una seconda dialog box; in essa si possono indicare la porta cui la stampante è con-

Figura 1 - Installazione di una o più stampanti mediante l'apposita opzione del Pannello di Controllo.



nessa (una delle porte parallele o seriali, FILE: per redirigere l'output su un file invece che alla periferica, None per nessuna porta), nonché i tempi di attesa per segnalare che la stampante non è in linea o che, una volta inviati dei dati, non si è in grado di trasmetterne altri. Vi è anche un pulsante *Imposta*, con il quale si provoca l'apertura di una terza dialog box.

Si tratta questa volta di una dialog box gestita direttamente dal driver della stampante; richiede infatti informazioni variabili secondo la marca, il modello e il tipo della stampante (il tipo di alimentazione, la risoluzione grafica, i font, la quantità di memoria, il formato del foglio, ecc.), che richiedono quindi una qualche conoscenza delle caratteristiche tecniche della periferica che si intende installare (figura 2).

Una volta completate l'installazione e la configurazione, magari per più di una stampante, si può tornare al Pannello di Controllo per scegliere quale sarà la stampante di default.

Le scelte operate mediante le dialog box gestite dal Pannello di controllo vengono registrate in tre sezioni del file WIN.INI.

Nella sezione *Windows*, introdotta dalla riga «[Windows]», la riga device mantiene l'indicazione della stampante di default in tre campi, separati da virgole: una descrizione sintetica, il nome del driver, la porta. Nelle sezioni *PrinterPorts* e *devices* vengono invece registrati descrizione, nome del driver e porta di tutte le stampanti installate; nella prima delle due, creata dal Pannello di Controllo ai suoi propri fini, compaiono anche i tempi d'attesa, mentre è inteso che un'applicazione, per sapere tra quali stampanti può scegliere, legga la seconda.

Le scelte operate mediante la dialog box gestita dal driver vengono anch'esse scritte in WIN.INI, in sezioni intestate e articolate in modo variabile. Anche in questo caso si tratta di informazioni che, di norma, non vengono utilizzate direttamente dalle applicazioni. Il device driver, oltre ad aggiornare il file WIN.INI, provvede anche a trasferire in un'area di memoria le scelte operate circa l'impostazione di una specifica stampante: si tratta di un *environment* analogo a quello dell'MS-DOS, ma da questo ben distinto; ne possono esistere tante quante sono le porte e possono essere scritti e letti solo mediante le apposite funzioni *SetEnvironment* e *GetEnvironment*. Il file WIN.INI, dal canto suo, può essere agevolmente consultato mediante la funzione *GetProfileString*, nei primi due parametri della quale si possono indica-

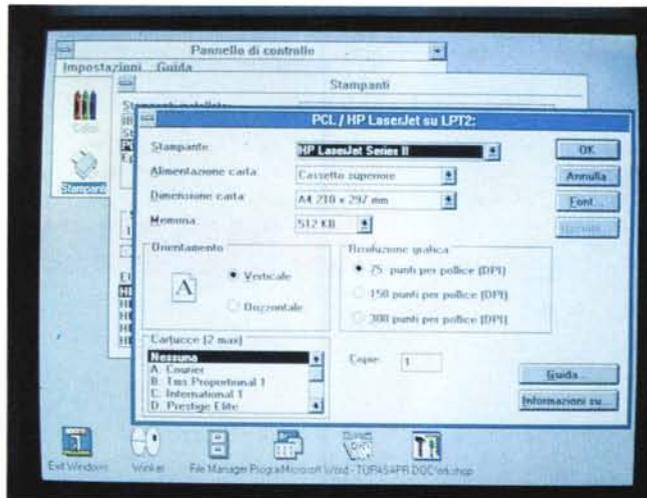


Figura 2 - Impostazione di una stampante mediante la dialog box gestita dal suo driver.

re il titolo della sezione del file e la stringa che definisce una sua riga.

Cercando ora di astrarre dalle specificità di Windows, possiamo sintetizzare i punti fondamentali di un tale impianto: a) è possibile installare e iniziare a usare rapidamente qualsiasi stampante, senza alcuna necessità di cimentarsi nella scrittura di un device driver, in quanto basta una conoscenza non superficiale, ma nemmeno specialistica, delle caratteristiche tecniche della periferica; b) ogni applicazione può accedere rapidamente a informazioni circa il tipo delle stampanti installate e delle relative porte; c) ogni applicazione può sapere in ogni momento quale è la stampante di default.

Scelta e impostazione di una stampante

Quando l'utente sceglie l'opzione *Impostazione stampante* di un'applicazione, gli viene proposta una dialog box contenente l'elenco delle stampanti installate, letto da WIN.INI (alcune applicazioni, come Write o PaintBrush, escludono le stampanti indirizzate su

None). Compare anche un pulsante *Imposta*, con il quale, selezionata una stampante, si può richiamare la dialog box gestita dal suo driver.

Qui bisogna aprire una parentesi. Solo pochi dei driver forniti con Windows 3.0 ne sfruttano tutte le potenzialità, in quanto la maggior parte di loro deriva da versioni precedenti. Tutti comprendono una funzione *DeviceMode*, solo alcuni la funzione *ExtDeviceMode*; non sempre si può fare riferimento ad una struttura *DevMode*, definita nel file DRIVINI.T.H dell'SDK e ribattezzata *TDevMode* nel file WINTYPES.PAS fornito insieme al Turbo Pascal per Windows. Un'applicazione deve capire come funziona il driver con cui desidera stampare, prima di poter colloquiare con la sua dialog box. Non ci curiamo ora di questi problemi; vediamo quindi cosa succede con un driver che comprenda la sola funzione *DeviceMode* (ricordo che i driver di Windows contengono funzioni in quanto, in realtà, sono delle DLL).

Quando viene chiamata, la funzione *DeviceMode* cerca di ricavare l'impostazione corrente della sua stampante chiamando la funzione *GetEnvironment*

```
[windows]
...
device=IBM Proprinters, PROPRINT, LPT1:
...

[PrinterPorts]
IBM Proprinters=PROPRINT, LPT1:, 15, 45
Stampante PostScript=PSCRIPT, FILE:, 15, 45
PCL / HP LaserJet=HPPCL, LPT2:, 15, 45
Epson 9 pin=EPSON9, None, 15, 45

[devices]
IBM Proprinters=PROPRINT, LPT1:
Stampante PostScript=PSCRIPT, FILE:
PCL / HP LaserJet=HPPCL, LPT2:
Epson 9 pin=EPSON9, None
```

Figura 3 - Le sezioni del file WIN.INI relative alle stampanti, tenute aggiornate dal Pannello di Controllo.

per la porta cui la stampante è associata, oppure, se il tentativo fallisce, leggendo il file WIN.INI (se neanche così vi riesce, usa valori di default). Riempie quindi con i dati così ricavati le varie sezioni della sua dialog box, che propone all'utente perché vi apporti le modifiche che desidera. Quando l'utente preme il pulsante *Ok*, la funzione aggiorna il file WIN.INI e l'environment associato alla porta e invia un messaggio WM_DEVMODECHANGE a tutte le applicazioni, perché queste possano aggiornare le loro conoscenze sull'hardware installato. Le modifiche hanno infatti effetto su tutte le applicazioni, fino a che qualcuna non le cambi a sua volta.

La funzione *ExtDeviceMode*, presente solo nei driver scritti o riscritti espressamente per Windows 3.0, consente di apportare modifiche solo temporanee all'impostazione di una stampante. Prevede infatti un parametro *WMode* che può assumere diversi valori, eventualmente combinati insieme mediante un operatore **or**. Se tra questi valori c'è la costante *dm_Update*, le modifiche hanno la stessa estensione di quelle apportate mediante la funzione *DeviceMode*; se si usa invece il valore *dm_Copy*, le modifiche vengono salvate in una struttura di tipo *TDevMode* che potrà essere usata per le stampe dell'applicazione, ma non si tradurranno in cambiamenti nell'environment o in WIN.INI. Avranno quindi effetto solo per l'applicazione che le ha apportate.

Astraendo ancora dalle specificità di Windows e sintetizzando, rileviamo che un'applicazione non solo può accedere rapidamente all'elenco delle stampanti installate, ma può anche cambiarne l'impostazione senza bisogno di usare altri programmi; le modifiche, inoltre, possono essere permanenti o solo temporanee.

Figura 5 - I codici di escape più comuni, implementati da qualsiasi device driver. L'elenco completo si trova nel primo capitolo della Windows Reference Guide del Turbo Pascal per Windows.

GETPHYSPAGE SIZE	determina le dimensioni del foglio di carta
SETABORTPROC	installa una «procedura di abort»
STARTDOC	informa il device driver che sta per iniziare la stampa di un nuovo documento
NEWFRAME	informa il device driver che la pagina corrente è stata completata
ENDDOC	segnala la fine del documento
QUERYESC SUPPORT	verifica se il device driver implementa un particolare codice di escape

```
[PROPRINTERdrv]
Orient=0
Res=0
Color=0
Mode=0
Width=04
Height=13
Feed=0
Printer=1
NoPageBreak=0

[PostScript,FILE]
feed1=9
feed15=9

[HPPCL,LPT2]
paper=9
duplex=0
FontSummary=C:\W3\FSLPT2.PCL
orient=1
tray=1

[EPSON9drv]
Orient=0
Res=2
Color=0
Mode=0
Width=05
Height=13
Feed=0
Printer=00
NoPageBreak=0
```

Figura 4 - Alcuni esempi delle sezioni del file WIN.INI create e mantenute dai device driver delle stampanti installate.

Stampa di un documento

La scelta e l'impostazione di una stampante possono avere influenza anche su altre scelte dell'utente. Si pensi a programmi che consentono di usare diversi font nel testo che verrà stampato: il numero e il tipo dei font disponibili dipende evidentemente in buona misura dalle caratteristiche della periferica. Ne sono esempio programmi di word processing come Word per Windows che, quando apre la combo box per la scelta dei font, propone elenchi variabili

secondo la stampante correntemente selezionata.

Per poter conoscere tali informazioni, un'applicazione deve per prima cosa impostare un *device context* mediante la funzione *CreateDC*. Vanno usati tre parametri, che specificano il nome del driver e della stampante e la porta cui questa è connessa, seguiti da un quarto parametro contenente l'indirizzo di una struttura *TDevMode*. Se tale ultimo parametro è nullo, viene chiamata la funzione *GetEnvironment*, altrimenti può trattarsi dell'indirizzo di una struttura riempita mediante uso della funzione *ExtDeviceMode*.

In altri termini, si possono usare sia le impostazioni di sistema che, ove consentito dal driver, quelle temporaneamente stabilite per le operazioni in corso.

Una volta creato il *device context*, si possono chiedere in vario modo molteplici informazioni; la funzione *EnumFonts*, in particolare, consente di elencare sia i font disponibili, sia le diverse dimensioni dei caratteri per ognuno di essi. Altre informazioni, come il formato e l'orientamento della carta, si ottengono con *GetDeviceCaps* e, se il driver è stato scritto o riscritto per Windows 3.0, con *DeviceCapabilities*. Si può inoltre attivare ora uno ora l'altro dei font disponibili, mediante la funzione *SelectObject*.

Quando si giunge alla stampa vera e propria, l'applicazione può usare le stesse funzioni con le quali si scrive sul video, a condizione che siano consentite dalla periferica (può capitare il caso di un plotter non in grado di riprodurre una bitmap); deve tuttavia provvedere ad alcune attività specifiche, quali l'avvio delle operazioni o i cambi pagina. Si dispone per questo di un'unica funzione *Escape*, il cui parametro *nEscape* può

GetMessage	aspetta finché non intercetta un messaggio inviato alla finestra il cui <i>handle</i> le viene passato nel secondo parametro o, se questo è nullo, ad una qualsiasi finestra dell'applicazione.
PeekMessage	verifica la presenza di messaggi inviati alla finestra il cui <i>handle</i> le viene passato nel secondo parametro o, se questo è nullo, ad una qualsiasi finestra dell'applicazione
WaitMessage	sospende l'applicazione per cedere il controllo ad altre fino a che un nuovo messaggio non viene posto nella coda dell'applicazione
Yield	sospende l'applicazione per cedere il controllo ad altre (NB: è preferibile l'uso di un <i>message loop</i> con <i>PeekMessage</i>)

Figura 6 - Le funzioni di Windows mediante le quali un'applicazione può cedere il controllo ad altre.

peraltro assumere ben 58 valori diversi (se li ho contati bene...), detti codici di escape.

La figura 5 riporta i più comuni di tali codici, quelli che si ritrovano in quasi tutte le routine di stampa, implementati da qualsiasi driver. Con STARTDOC si segnala a Windows l'inizio di un nuovo documento, provocando così l'inizializzazione della stampante e il caricamento in memoria di Print Manager (a meno che la riga *Spooler* della sezione *Windows* di WIN.INI non contenga un «no»).

Mediante le funzioni *GetTextMetrics* e *GetTextExtent* si possono misurare l'altezza e la lunghezza di ogni stringa inviata alla stampante; ricordo che la prima delle due funzioni ci dice sia l'altezza massima dei caratteri di un dato font (*tmHeight*), sia di quanto sia opportuno aumentare tale valore per ottenere un'interlinea ottimale (*tmExternalLeading*); ciò consente di determinare l'ampiezza di ogni riga e il numero di righe per pagina e, quindi, di segnalare a Windows al momento giusto il cambio di pagina mediante il codice di escape NEWFRAME.

Il codice ENDDOC viene invece usato per segnalare a Windows la fine del documento.

Anche qui vogliamo sintetizzare astraendo dalle specificità di Windows: una volta scelta una stampante, l'applicazione ha accesso ad informazioni circa le dimensioni della carta e i font disponibili; una volta scelto un font, l'applicazione può costantemente tenere sotto controllo sia la lunghezza delle righe di stampa (in quanto conosce la lunghezza di una stringa composta di caratteri scritti con quel font), sia il numero di righe per pagina (sulla base del valore di interlinea).

Interruzione della stampa

La stampa di un documento con Windows pone problemi del tutto particolari, dipendenti dal suo multitasking cooperativo. Un'applicazione che non sia quella su cui si sta correntemente lavorando, infatti, ha occasione di fare qualcosa solo quando quella corrente cede il controllo; ciò accade di norma piuttosto spesso, in quanto il *message loop*, nascosto nel metodo *TApplication.Run* del Turbo Pascal per Windows, chiama continuamente la funzione *GetMessage*, che è appunto una di quelle che cedono il controllo. Ciò significa, in concreto, che un'applicazione cede il controllo con una frequenza inversamente proporzionale al tempo che impiega nel rispondere ai singoli messaggi.

Ne possono derivare problemi per le stampe, in quanto, generalmente, i documenti da stampare vengono provvisoriamente parcheggiati in file temporanei; se si verifica una situazione di disco pieno durante una stampa, Print Manager non ha la possibilità di scaricare quei file su disco in quanto l'applicazione, impegnata ad inviargli una pagina dopo l'altra, non gli cede il controllo.

Facile immaginare le conseguenze se non si fosse prevista una via d'uscita. Salvo il caso del ricorso a tecniche particolari (il *banding*, ovvero la scomposizione di una pagina in diverse «strisce»), quei file temporanei vengono creati e poi inviati alla stampante pagina per pagina: si completa il file relativo ad una pagina, lo si manda alla stampante, si passa al file per la pagina successiva, e così via. Il momento critico è quindi quello dell'invio di un codice di escape NEWFRAME; questo viene processato da GDI.EXE (una delle tre DLL che costituiscono il «cuore» di Windows), che,

mentre vi provvede, si preoccupa di chiamare frequentemente una cosiddetta «procedura di abort» (in realtà una funzione), che l'applicazione deve curarsi di installare mediante il codice di espace SETABORTPROC prima di inviare il codice STARTDOC. La funzione deve contenere un vero e proprio *message loop*, compresa una funzione *PeekMessage* con la quale cedere il controllo. Di ciò può approfittare Print Manager per liberare il disco di quei file temporanei. La funzione installata con SETABORTPROC può ritornare TRUE o FALSE, per segnalare nel primo caso che la stampa può proseguire regolarmente, nel secondo che va interrotta. Si usa quindi aprire una dialog box non modale subito prima di inviare il codice SETABORTPROC, con la quale proporre all'utente la pressione di un pulsante per interrompere la stampa; nel caso l'utente si decida in tal senso, la funzione ritornerà FALSE.

Prescindiamo, naturalmente, dalle peculiarità dell'interazione tra applicazione, GDI e Print Manager, per limitarci a prendere nota dell'utilità di un meccanismo che consenta di interrompere una stampa quando lo si desidera.

Voglio solo accennare, infine, ad un altro tipo di possibili interruzioni: il verificarsi di un errore critico (stampante assente, spenta o senza carta). In questi casi Windows apre una dialog box che propone di ritentare (ovviamente dopo aver eliminato la causa del problema) o di interrompere la stampa. Una gestione sicuramente più pulita di quella propria del DOS, in quanto non compare nemmeno quell'opzione «Annulla» che provoca la brusca e incontrollata

138

Sergio Polini è raggiungibile tramite MC-link alla casella MC1166.



TAIWAN E' IN LINEA



OTC - Via Maestri del Lavoro, 48 - 48010 Fornace Zarattini (Ra) - Tel. 0544 / 500608 / 500715 - Fax 0544 / 500695

Distributori regionali:

LOMBARDIA

Sig. Luca Recigno
Pulsar s.n.c.
Via S. A. Merici, 6
25123 Brescia (BS)
Tel. 030 / 3760774
Fax 030 / 3760168

TRENTINO ALTO ADIGE

SIG. Luciano Ercolani
Gread Elettronica s.r.l.
Via Zeni, 8
38068 Rovereto (TN)
Tel. 0464 / 443235
Fax 0464 / 443237

VENETO

Sig. Maurizio Andretta
OTC Veneto s.r.l.
Via Europa, 2
35015 Gagliera Veneta (PD)
Tel. 049 / 9400492
Fax 049 / 9400492

ROMAGNA

Sig. Giorgio Del Bene
C. S. Computers s.r.l.
Via Indipendenza, 4
47033 Cattolica (FO)
Tel. 0541 / 963801
Fax 0541 / 953847

MARCHE - ABRUZZO

Sig. Marcello Caselunghe
El - TEL s.r.l.
Via Leopardi, 14
65015 Falconara Marittima (AN)
Tel. 071 / 911334
Fax 071 / 9173816