



Imperfect timing

Tim Phillips gives his overview of the new beta version 7 features included in PerfectOffice and WordPerfect.

If you've been waiting patiently for me to review WordPerfect 7, then I'm sorry: you'll have to wait another month as I've only just received my beta copy. But I can fill you in on the new features of the Corel-owned PerfectOffice 7. At this point, I'm not going to tell you how good they are as I'll need longer to work that one out.

WordPerfect 7 has finally woken up to the Internet! There's an Internet publisher supplied as an integral part of the package. It's easy to make HTML documents using the Publisher, which uses a series of pull-down menus to give you the elements of HTML markup. Conversion from HTML to WordPerfect format is in there, too — not revolutionary, granted.

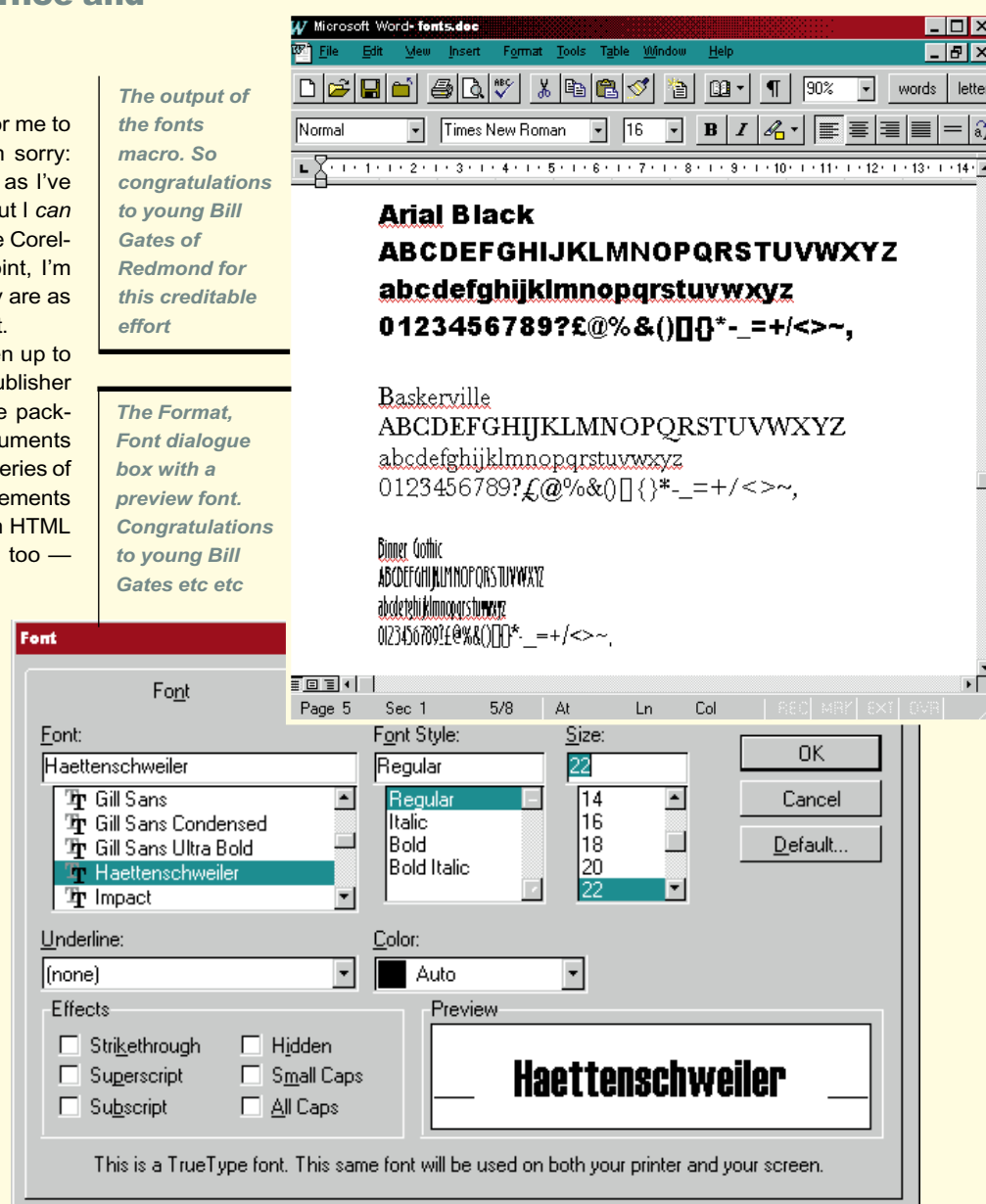
Spell checking, grammar checking and the Thesaurus are now in one dialogue box, so if you use all three it makes the whole thing easier. On the other hand, if you don't use all three (which, judging from my postbag, could apply to most of you) this isn't an improvement.

Perhaps the biggest benefit for users is the amount of common applications in the suite. There's a neat, common address book — good for small businesses but irrelevant for larger ones — that has made the mailmerge much better. Corel claims that PerfectOffice has more common code than any other suite.

The remainder of the improvements are playing catch-up to Word and Word Pro: improving the automatic correction and autoformatting, including a Quick-Finder and a right-click context sensitive menu, adding tabbed dialogue boxes to contain all these functions,

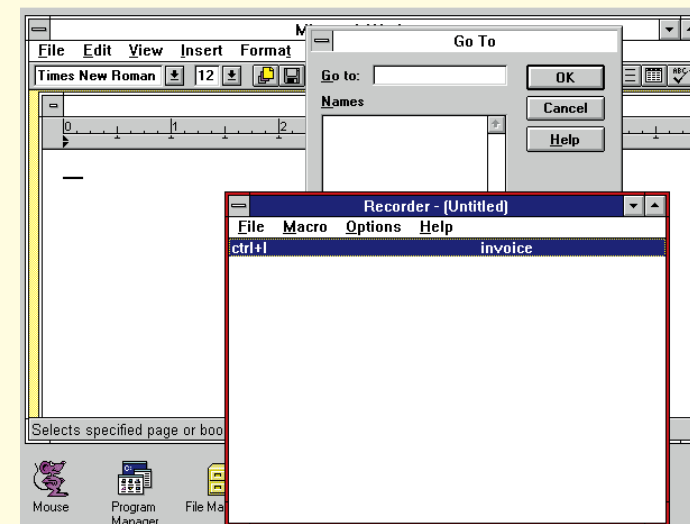
The output of the fonts macro. So congratulations to young Bill Gates of Redmond for this creditable effort

The Format, Font dialogue box with a preview font. Congratulations to young Bill Gates etc etc



and long filename support. There's rotated text in tables, but as you can guess, I'm struggling to find a good reason to buy PerfectOffice 7.

First thoughts are: it's fine if you desperately want to retain your WordPerfect commitment in a 32-bit environment; but it does nothing more than Word can do. I'm



Recording a macro using Recorder, Windows 3.11 and Works. Use this for the invoice numbering solution

TrueType format. It's a good, simple idea and the fact that I didn't think of it first shows that I'm

hardly a cheerleader for Microsoft but I predict that it will not displace many Microsoft Office sales. Like Office, PerfectOffice has traded up in its hardware requirements and we've ended up with word processors that react sluggishly on anything below a Pentium.

Lista' fonts

Several of you have helpfully pointed out that the macro to display all your fonts (*Hands On*, June) has already been supplied by Microsoft. Well done, Microsoft — you don't get a prize, as you're richer than Croesus already. The macro in question (shown in *Fig 1*) is found in the Word 6 template called Macro60.dot.

It takes a few minutes to chug through all your fonts but the result is a document which shows them all, with names. This is only useful if you print it out because Word shows previews of fonts in the Format, Font dialogue box.

True to type

Here's something for TrueType fans: Formula Solutions of Ilford is offering you the chance to get a company logo produced in

just not cut out for this entrepreneurial stuff.

You supply the logo to Formula and in return you get a TrueType font — basically, versions of the logo instead of letters. You choose the logo characters just as you would choose a font in your word processor and the logo is in your document: scalable, with no loss of definition.

This additionally solves the problem of document growth that some of my correspondents experience — Formula estimates that this adds about 30 bytes per logo to a document. Since when did a graphic file occupy only 30 bytes? On the down side, the logo needs to be mono — but the output is exceptional. If you're interested phone Formula on the number shown in our Contacts panel.

Look — no macro!

William Barneby, of Madley, adds a macro-free solution (see page 286) to the invoice number challenge (*Hands On*, April and May) for all you Works for Windows users. This uses Recorder, the much ignored keystroke macro generator that Windows 3.x uses.

Fig 1 Out for the count

```
Sub MAIN
CR$=Chr$(11) + Chr$(9)

FileNew.NewTemplate = 0,.Template="Normal"

MyFonts=CountFonts()
For count=1 To MyFonts
    A$=Font$(count)
    Font A$,16
    Insert A$+CR$
Insert "ABCDEFGHIJKLMNOPQRSTUVWXYZ" + CR$
Insert "abcdefghijklmnopqrstuvwxyz" + CR$
Insert "0123456789?@%&()[]{}*_+</>~," + CR$ + CR$
Next count
End sub
```

If you remember, the challenge is to produce a document which can be used to generate invoices: it automatically assigns a sequential number to the invoices as they are generated.

1. Open a blank word processor document.
2. Embed a two-column spreadsheet on the right-hand side, stretching the length of the page.
3. In cell A1, type '0'.

Tim's Macro Club

Teleprinting

Two letters in a matter of days from the enthusiastic Graham Brown of Brent Cross about his macro for Word 6 called Teleprinter. It's one of those simple but useful ideas that I never have: the macro simply types your document back to you, with the speed being dependent on what value you use in the loop "For y=1 to 25:Next". More than 25, it's slower. Less than 25, it's faster. The idea, Graham says, is to help with proof-reading, or to make a demo screen.

```
Sub MAIN
EditSelectAll
a$ = Selection$()
FileNewDefault
For x = 1 To Len(a$)
Insert Mid$(a$, x, 1)
For y = 1 To 25 : Next
Next
End Sub
```

Now, I'm not answering any email from users who say they can't get it to work... but, hardly had I read Graham's first letter than another one hit my desk — with a revision. The revised version contains pauses for punctuation. This month's prize goes to Graham. I've found this macro extremely useful already, as I edit a lot of copy and reading it on screen can be a nightmare.

```
Sub MAIN
EditSelectAll
a$ = Selection$()
FileNewDefault
For x = 1 To Len(a$)
Insert Mid$(a$, x, 1)
If Mid$(a$, x, 1) = "," Then For y = 1 To 300 : Next
If Mid$(a$, x, 1) = "." Then For y = 1 To 600 : Next
If Mid$(a$, x, 1) = ";" Then For y = 1 To 500 : Next
If Mid$(a$, x, 1) = ":" Then For y = 1 To 400 : Next
For y = 1 To 10 : Next
Next
End Sub
```

The next challenge in developing this is to make it stop for changes. There needs to be an interrupt command to allow you to stop the macro executing so that you can edit the document. At the moment, you need to run it from the Tools menu to get the macro controls and press Escape to stop the macro, which is a bit clunky. More work on this one, please — and versions for other platforms. It's not a taxing concept, so I'm expecting a good response.

More on A5 booklets

To recap: Alison Walley wanted to print word processor output as an A5 booklet. I've had a letter from Jean Elliott, who actually does this to produce booklets for Word users who she trains. Warning! Exhaustive process follows:

- 1. Make a landscape template called "Bookpage" which has wide margins so that the text area is 4.35ins wide.
- 2. When you finish each page, make that text into a piece of AutoText and call the AutoText by its page number. Remember — to make AutoText you highlight the text and select Edit, AutoText.
- 3. Use the template organiser to transfer the AutoText items to another landscape template, which is set up with columns 4.35ins wide. Start a new document and type the page numbers in printing order, with page and section breaks to match. This is a bit subtle, as Jean shows. For a 24-page booklet, the order would be:

```
11<CR><CB>12<CR><PB>13<CR><CB>10<CR><ES>
09<CR><CB>14<CR><PB>15<CR><CB>08<CR><ES>
07<CR><CB>16<CR><PB>17<CR><CB>06<CR><ES>
05<CR><CB>18<CR><PB>19<CR><CB>04<CR><ES>
03<CR><CB>20<CR><PB>21<CR><CB>02<CR><ES>
01<CR><CB>22<CR><PB>23<CR><CB>00<CR>
```

- where <CR> is a carriage return, <CB> a column break, <PB> a page break, <ES> the end of a section, and page 00 the contents page.
- 4. Move to each number and press F3. The AutoText item is automatically inserted.
 - 5. Page numbering: set up a one-row, three-column table as a footer, using odd and even page footers and section breaks.
 - 6. Print the first page of each section on one side of A4; the second page on the other side.
 - 7. Lie down in a darkened room.

So what if it's not the most elegant solution? It gets the job done. I'm happy to hear of any more elegant methods but I'm not sure there are any.

WordPerfect, A5 booklets and Initial caps

Three of my most favourite things in one! I'd better get on with it or I'll wet myself with excitement. Following on from Jean-Luc Addams' excellent WordPerfect 5.2 A5 booklet macro last month, he has written a follow-up macro to allow you to use drop caps in a document. I have put it on this month's free, cover-mounted CD-ROM — I've left the explanation in, but when retyping you can leave it out.

4. In cell B1, type '1'.

5. Reduce the width of column B to zero.

6. Select any cell in column B.

7. Save the document as a template.

8. Record the macro (Fig 2) in Recorder. If you haven't done this before, simply start
- up Recorder and start recording. It will record your keystrokes, which should be as shown in Fig 2.

9. Customise your invoice template and save it under the same name.

To use the invoice, open the document

Complete the invoice and then save as a separate document in an invoice directory.

That's it for this time. Note my new CIS number, shown in the Contacts panel below. I won't be answering mail on the old address, so please use the new one. I much prefer my Internet address anyway!

Fig 2 A macro-free solution	
<F5>	(go to)
B1	(cell containing seed number)
<ctrl-C>	(copy)
<left arrow>	(moves to A1)
<alt-E>	(opens edit window)
S	(paste special)
A	(add to value in A1)
<enter>	(confirm and close menu)
<ctrl-S>	(saves template with new number)

and the last invoice number will be in A1. Double-click on the spreadsheet column and run the macro, which will add one to the number and save the template for next time.

PCW Contacts

Contact **Tim Phillips** by surface or airmail to PCW, otherwise email him at his new CIS address
CompuServe 104047,2750
or **wong@cix.compulink.co.uk**
Formula Solutions 0181 252 4444