



The right type

Tim Phillips reveals the latest development in the world of fonts, eats his words on the subject of grammar checkers, and offers up some tasty macros.

Although it's tempting, I can't continually keep explaining fonts in this column because real life keeps overtaking me. For instance, this month, (to borrow the style of a newspaper headline): ADOBE AND MICROSOFT IN UNIVERSAL-FONT-FORMAT-PLAN SHOCK.

If you remember, the history of software fonts is one of Adobe coming along and giving us PostScript, which defined the territory for scalable vector fonts: those made up of mathematical formulas rather than dots on a screen. Then Microsoft gave us its own set of vector fonts, called TrueType, which is supplied with Windows and Windows applications. Check your PC and you will probably find at least 30 TrueType fonts.

Poor old PostScript hasn't been too prominent in the PC market recently, although for Mac users (and, by implication, most designers and typographers) PostScript is still the dominant vector font format.

The problem Microsoft and Adobe are trying to solve is how to establish

a common font format for the Internet. Web pages have any type style on them at the moment, but to embed fonts in a Web page as you embed them in a word processor document, you need a set of fonts that any operating system can use.

So Microsoft and Adobe plan to merge PostScript and TrueType into a new format, to be called OpenType. To do this the two companies will cross-license each other's technology and make the

OpenType format freely available to anyone who has a vested interest, such as other operating system vendors.

OpenType will be included in all future Windows and Adobe software releases including Acrobat, its portable document software. This paves the way for a common format for embedding fonts in Web documents: miles away from the original aims of the Web, to prioritise content not form. But what the hell, it's one less format to worry about and that's a good thing.

Web publishing

Part 164 of my quest for the best HTML authoring software: Andrew Duke of Southampton University proposes I should use EasyHTML, a shareware application written by... Andrew Duke himself.

He has a point. The problem with the HTML applications supplied by word processor companies is that they never produce true representations of the document you are converting. I may be a bit simple but I, too, have trouble learning them, so I fall back on manual coding.

Fig 1 Spelling checker toggle

```

'Toggles Autospell and displays setting in status Bar
Sub MAIN
Dim spellvals As ToolsOptionsSpelling
GetCurValues spellvals
If spellvals.AutomaticSpellChecking = 1 Then
ToolsOptionsSpelling .AutomaticSpellChecking = 0
Print "Autospelling OFF"
Else
ToolsOptionsSpelling .AutomaticSpellChecking = 1
Print "Autospelling ON"
EndIf
End Sub

```

Statistical Summary

Total Words:	801
Total Sentences:	43
Words in Sentences:	801
Passive Verbs:	6
Average Sentence Length:	19 Excellent
Passive Index:	14 Excellent

for General Writing

Sentence Lengths

Sentence Length (words)	Number of Sentences
0-5	2
5-10	5
10-15	10
15-20	12
20-25	8
25-30	6
30-35	4
35-40	3
40-45	2
45-50	1
50+	1

Writing Task: General Writing Long Sentence Limit: 40

Fig 2 Font display macro

Bob Lauder's short version of a font display macro (the longer version will be on next month's cover-mounted CD-ROM):

```

Sub MAIN
FileNewDefault
StartOfDocument
Dim Fonts$(CountFonts())
For i = 1 To CountFonts()
Fonts$(i) = Font$(i)
Font "courier new"
Insert Fonts$(i)
InsertPara
InsertPara
Font Fonts$(i)
For Alpha = 16 To 255
Insert Chr$(Alpha)
Insert Chr$(32)
Next Alpha
InsertPara
InsertPara
Next i
End Sub

```

EasyHTML is semi-automatic. Run it under Windows 3.1 or 95 in 800 x 600 or 1,024 x 768 resolution and you get a tabbed dialogue box, with six formatting categories. You select the formatting you want for the text using check boxes and radio buttons. The style tags are then generated and you paste these into the document, which is open as plain text

alongside HTML in your word processor. There are pages to help you format images and links, too, but you don't get a WYSIWYG display. You will learn about style tags though, so the program functions as a tutor as well.

Will I be using it? Well, not at the moment. Although it costs only £10, EasyHTML doesn't support frames or tables, which have to be coded manually. I'm looking forward to version 1.1, but I recommend you try the application out for yourself: download it from <http://www.soton.ac.uk/~ajd/easyhtml/>.

Tim's macro club

The debate still rages over whether the autospellcheck function is a good or a bad thing. I reckon it's great, but lots of you hate the wiggly red line that Word puts under letters. Fig 1 shows a timely macro from Gavin Brock at Imperial College, London. It's a simple function that toggles the spelling checker on and off, and puts a display in the status bar. From the same source we have the beginnings of a cross-word solver. The macro in Fig 3 (overleaf) uses the word dictionary to find words by entering a wildcard pattern, using ? for single and * for multiple blanks, or as a single word anagram. "I know its cheating but it helps," says Gavin, and he's right.

Bob Lauder has sent a much improved

Hints & Tips

A bumper crop of hints and tips from Alan O'Brien came my way via CompuServe. Here are three of his ideas that I liked:

● Saving in good order

To keep track of his documents, Alan has a directory for each month of each year on his hard drive. To make sure his documents save in the right directory, he uses the system date in a macro to automatically allocate them, and another macro automatically saves to another drive for backup.

• Alan is keeping his macros to himself, so prizes for anyone who provides a relevant set (or subset) of the required macros for any popular word processor.

● Footer fun

Add the path and filename of the document as a footer. Then the hard copy will show you where the original is kept. In Windows, the field is: {filename \p\ *Caps}

● Over here John... on my header

Automatic numbering of paragraphs in Word causes problems for many of you — the biggest gripe is that it starts numbering again, at one, after any header. Alan instead defines his own styles, such as "MyHeader", which are used instead. This is a kludge but not a bad one, because when you are making a table of contents from a long document (which you should try; it's excellent) you can ask Word to look for those style tags instead of headers. The real problem is that you can't work from an outline defined in the Outliner.

• I'll be returning to outlining, indexing and generating tables of contents in a couple of months, but before then, I'd welcome any tips from you lot at the sharp end.

● A5 booklets for all word processors

We've received more in the mailbag following Alison Walley's need for a macro to make her documents print in A5 booklets. George Lukic wrote from Dar Es Salaam (yes, really) to tell me of a program called ClickBook, which he got for \$20 from BookMaker, 2970 El Camino Real, Palo Alto, California, US. Apparently, the software installs itself as a macro into most Windows word processors. That's all I know, but George swears by it and has no complaints.

• If you want to be happy like George, instead of grumpy like me, I'd suggest you give the company a call. Reports, please.



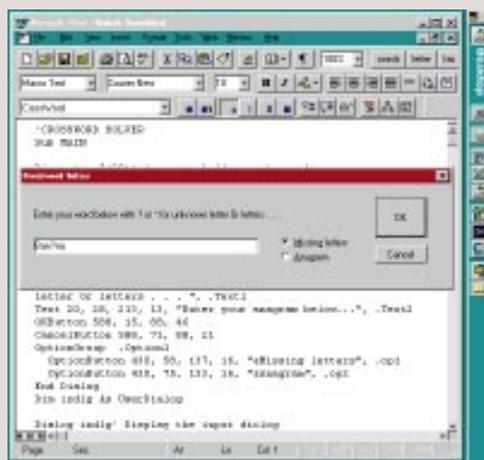
Fig 3 Crossword solver

```

'CROSSWORD SOLVER
Sub MAIN
Dim outarr$(20)' Array to hold search results
On Error Goto Stopnow' In-elegant but efficient way to quit
While (1)' Infinite loop
' Define the input dialog box
Begin Dialog UserDialog 712, 120, "Crossword Solver", .getword
TextBox 20, 63, 374, 18, .TextBox
Text 20, 28, 483, 13, "Enter your word below with ? or * for unknown letter Or letters . . . ", .Text1
Text 20, 28, 213, 13, "Enter your anagram below...", .Text2
OKButton 588, 15, 88, 46
CancelButton 588, 71, 88, 21
OptionGroup .Option1
  OptionButton 430, 58, 137, 16, "&Missing letters", .op1
  OptionButton 430, 75, 133, 16, "&Anagram", .op2
End Dialog
Dim indlg As UserDialog
Dialog indlg' Display the input dialog
Redim outarr$(20)' clear output array
n = ToolsGetSpelling(outarr$(), indlg.TextBox, , indlg.Option1 + 1)
If (n > 0) Then' If any words found
  Begin Dialog UserDialog 294, 302, "Results"
  ListBox 10, 6, 179, 290, outarr$(), .ListBox1
  OKButton 195, 7, 88, 287
  End Dialog
  Dim outdlg As UserDialog
  Dialog(outdlg)'Display results
EndIf
Wend
Stopnow:' Ends the macro
End Sub

Function getword(identifier$, action, supvalue)' Dialog function
  Select Case action
  Case 1 'The dialog box is displayed
    DlgFocus "TextBox"
    DlgVisible "Text2", 0' Default to missing letters
  Case 2 'The user selects a button
    If (identifier$ = "op1") Or (identifier$ = "op2") Then
      DlgVisible "Text1", supvalue - 1
      DlgVisible "Text2", supvalue
      DlgFocus "TextBox"
    EndIf
  Case 3 'Text change (not applicable)
  Case Else
  End Select
End Function

```



*The crossword solver: ? for one blank; * for more than one blank. It finds the possibles from the Word dictionary and has reduced my best time from years to weeks*

style is to certain templates, such as business writing, copywriting, memos and so on. It does this by the rough-and-ready method of counting the words in sentences, the average length of words and the proportion of passive phrases. It also checks your copy according to a standard set of grammar rules.

This version of StyleWriter has some decent Windows connectivity. It will automatically pick up features when installed alongside Word (2 or 6), Ami Pro or WordPerfect (5 or 6), or it will check text on the clipboard. It's a trim little application, weighing in at only 1Mb, and it's fast, too. So even if it doesn't do a great deal, it's an improvement on the grammar checkers provided with many applications. If you are committed to using your computer to improve your prose, it's money well-spent. But don't sack your staff because of it. This is not artificial intelligence by any stretch of the imagination. ■

version of last month's font display routine which will be included on next month's cover-mounted CD-ROM. It's a long macro, but worthwhile. It prints a selection of each font in a box with its name next door. The output is as good a reference sheet as you need. In addition to the long macro, Bob also sent a shorter version, shown in Fig 2 on the previous page.

Mind your language

Not all of you agree with me that grammar checkers are a waste of time, including Editor Software, which sent me the latest

version of StyleWriter. I remember this souped-up grammar checker from the bad old days when I was a cub reporter. The company I worked for suggested sacking all the people who check the features and buying a copy of this package to replace them. I disagreed but StyleWriter won and I was looking for another job.

So when Editor Software called suggesting I review StyleWriter, I was looking for my revenge. I have to admit, however, that it's quite good. StyleWriter goes beyond the bounds of usual grammar checkers and judges how appropriate your

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**