What happens when things go wrong?

(Video: youtu.be/AY6B-IkLEN8)

I'll give you a general description of what to do to report an error to me, and then I'll list a few errors that sometimes occur, giving you a suggestion of the possible cause. *Oh, but first...*

A general suggestion

If you get errors with some of the macros – especially all the ...***Alyse*** macros – it may be worth creating a text-only version of your file and then running the macro on that copy.

One way is to use:

Ctrl-A, Ctrl-C, create a new file, Paste as Pure Text

but to be sure that you're getting **all** the text – including what's in the foot/endnotes and text boxes – you can use the macro *CopyTextSimple*. (Better still, especially for large files, use *CopyTextVerySimple*.)

"Word has crashed!" – but has it?

There are times when Word seems to have crashed – but it might not have. Just because it claims that it's "Not responding" it doesn't mean that it has crashed. Try to be patient – go and make a cup of tea, perhaps, especially if you realise that you had *omitted to save the file* (bad idea!) before you ran that macro.

But if it still seems to have crashed (e.g. the red 'X' box at the top right of the Word window is only glowing dull red), you might still be able to regain control of Word (and thus to save that unsaved file!).

One trick you can try (on PCs only, sorry, unless Mac users can tell us your equivalent action) is to open the Windows Task Manager by pressing Ctrl-Alt-Delete. The Task Manager display will show you whether Word is still active. *Whether it is or not*, do NOT click on End Task. Click Cancel, and then **either** you will be able to click in the Word window and you'll have control back, **or** you'll have to click the dull red close box, and Word will do its best to save your unsaved file(s).

How to respond to – and tell us about – an error

Sometimes, when you try to run a macro, it generates an error, and Visual Basic (VBA) asks you what you want to do, offering you:

> **End, Debug, Help.**

Ironically, the least helpful of these is to click 'Help'. Don't bother.

If you just want to give up altogether and ignore the idea of using that macro, you can click on End.

To find out what went wrong – perhaps so that you can report the error to WMT – the first thing to do is to make a note of how VBA describes the error. Here's an example:

> *Runtime error '5174':*
> *This file could not be found.*

MS Word won't let you copy and paste the error message, but you could perhaps go over to your email software or to a text editor, such as NotePad, and start to compose an email to WMT, typing in this error message.

Next, click on Debug. Debugging is a technique that programmers use to try to work out what has gone wrong with a program. This will take you into VBA with one of the lines of the macro highlighted in yellow, maybe looking something like this:

```
    If gottaList = False Then
      Documents.Open dirName & listName
```

```
Else
  listDoc.Activate
End If
```

Make a note of the line so that you can report it to us. However, this time, you *can* do it by selecting a bit of the macro, either side of the yellow line, copying it, and then pasting it into a Word file (or your email), where it will appear as ordinary text. But please explain exactly **which** line was actually highlighted in yellow – this is important if you want us to correct the problem.

Next, you have to stop the debugging process, or 'reset' VBA. You do this by clicking the Reset button on VBA's top tool bar. Look for the set of three icons – as on an AV device: Play, Pause and Stop. The 'Reset' button is the square block, as used for 'Stop' on an AV player.

Send us that information, and we'll see what we can work out.

(If you **don't** stop the debugging process and simply go back into Word, all will **seem** to be OK. However, when you later try to run another macro, it will generate the error: '*Can't execute in break mode*'. You then have to click 'OK', select the VBA window and click the 'Reset' icon, as mentioned above.)

Some possible errors and their possible causes

"Variable not defined" – Search in VBA amongst your macros, and see if there's a line saying `Option Explicit`. If so, put an apostrophe in front of it, to disable it. This won't harm the operation of any of the other macros.

"The Find What text contains a Pattern Match expression which is not valid." – There are various reasons for this. In general, just report it to me, as above.

However, if you're using any of the *XyzAlyse* macros, such as *DocAlyse*, then it's likely to be a problem with what's called the 'list separator' used in the operating system of your computer. This is especially likely if you've got a computer set up for mainland Europe or South Africa. Here are my standard instructions:

The 'list separator' used within Word needs to be a comma, not a semicolon.

However, this is not a Word option, rather it's an **operating system option**.

So, on Windows 7, 8.1 and 10, it is in the Control Panel under 'Clock Language and Region' and then 'Region' and then 'Additional settings' (which is a button near the bottom of the Region window). In 'Additional settings', the fourth from the bottom is 'List separator'. Change it to a comma and click OK.

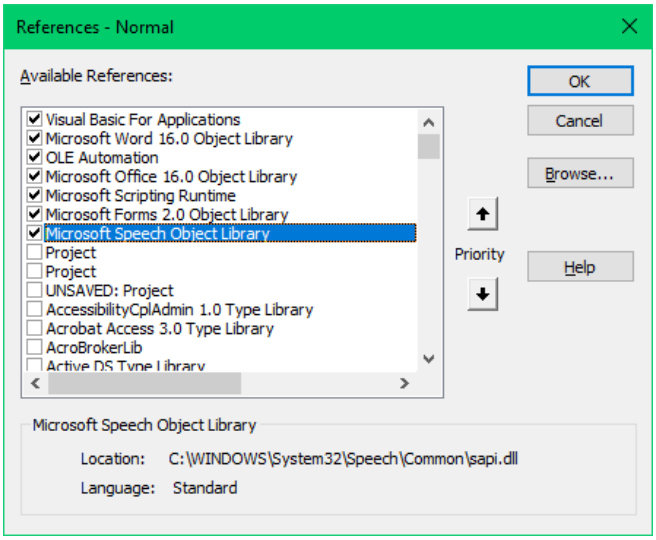
(If the above instructions don't relate to your computer, please email us and we'll try to help; that way we can add to these instructions, then other people won't suffer the setback you just had. Thanks.)

"Compile error: ambiguous name detected: BlahBlah" (where 'BlahBlah' is the name of a macro) – This means that, in pasting an extra macro into VBA, you've ended up with *two* copies of the macro *BlahBlah*. So the solution is to delete one of them. How? You do it in Visual Basic, and you do it very carefully, making sure to delete a whole macro from 'Sub' to 'End Sub', inclusive.

Note: You can select a whole macro automatically, by double-clicking. However, you need to know *where* to double click: you do so in the 2 mm-wide white margin, between the actual words of the macro and the light-grey vertical strip that looks like (but isn't) a vertical scroll bar (which is the vertical bar on the right, as with all application windows).

"User-defined type not defined" – If you get this error message, try the following: go into Visual Basic and click on: Tools – References.

It **should** look something like this:



In particular, check that you've got both Microsoft Scripting Runtime and Microsoft Forms 2.0 Object Library. Or maybe others are missing from the list of ticked items?

If not, look down through the list of unticked items and see if you can find either or both of them. If so, tick them, click OK, close VBA and then [b]close and re-open Word[/b].

Does that do the trick? (If not, try a complete restart of the computer.)

"Compile error: procedure too long" – This normally only occurs if you're trying to use DocAlyse. And only on a Mac. So if you get this error, try using the macro, *DocAlyseForMac*. If even that gives the same error, there is a solution: try *DocAlyseForThinMacs* and if that fails, there's finally a *DocAlyseForVeryThinMacs*.

Problems with losing or gaining spaces – If when you run a macro you and uop with words joinedtogether or extra spaces appearing (as per the previousbit of this sentence) then the culprit might be Word's 'helpful' Smart Cut and Paste option. **PC users** can look in my Appendix 9 *Word 365 Options*, and search for 'My advice: switch this off!', where there are four options that I think are (a) unnecessary and unhelpful for an editor and (b) in some circumstances cause macros to malfunction.

For Mac users, here are some instructions for removing the worst of the four (Smart Cut and Paste):

1. Click on '*Word*' in the upper left of your Word window.
2. From the drop-down menu, click on *Preferences*.
3. A new window will appear. Click on *Edit* in the *Authoring and Proofing Tools* section.
4. In the new window, uncheck *Use smart cut and paste*.