

Creating new user experiences?

Via user interfaces, user journeys and narrative



Communicator

The Institute of Scientific and Technical Communicators
Summer 2017

localisation e-learning
technical API
content CSS UX personas
ethics Email project UI
editing system
writer compliance user
XML DITA ISTC single-source
translation PDF UA
communicator CCMS software

Discover ST4 and
Paligo software

Promote yourself by
creating a portfolio

Tailor your documentation
for the European market

Read all about it!
From hot-metal to digital

Using the Quick Access Toolbar

Your first steps towards a simple GUI by [Mike Mee](#).

My excuses first

This issue's article will be smaller than previous ones, for which I apologise.

I have no real excuse other than breaking my ankle in the first week of March and finding that 'sitting comfortably' and 'broken ankle' are not happy bedfellows.

Weeks went by as I put writing this article off and off and it was only when asked by Katherine if I had an article to submit, and that I had less than a week in which to complete it, that I made a start!

Introduction

This article will be the forerunner to me taking your VBA code to the next level where you will be adding a simple graphical user interface (GUI) to your macros.

As this subject can be a large and unwieldy one, I will cover the easier side of creating a GUI first by showing you how to attach your macros to the Quick Access Toolbar. I am using Word 2016, so some of the menu items or options may be different for other versions.

Quick Access Toolbar

The what? Some of you may be aware of it, some might not, but the Quick Access Toolbar (QAT) is usually above your menu bar.

Word allows you to customise what icons appear on it and, you might find that you already have some icons up there already. The more recent versions of Word have Undo/Redo options available as standard.

Customising your QAT

Customising the QAT is very simple: you can add shortcuts to most of Word's built-in functionality as well as your own macros. You then do not need to remember which ribbon has which function in it.

For my own personal use, I add two shortcuts: **Quick Print** and **Print Preview and Print**. These options just help me when I am working on a document.

The QAT is also 'always there', regardless of whichever ribbon options

you are currently using too. You can move it above or below the ribbon.

Along with your own shortcut preferences, you can also add macros to the QAT.

Adding functions and macros

First a quick refresher for those who have not tweaked their QAT or an introduction to those who have never used it before.

Click the QAT dropdown icon at the end of the row of icons, will show you a menu of what you currently have on there. See Figure 1 below for an example of my menu options.

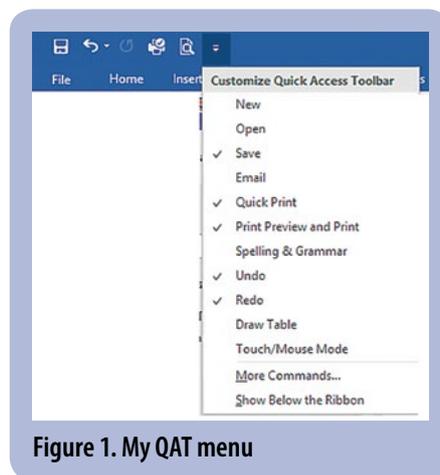


Figure 1. My QAT menu

At the bottom of the list is the option **More Commands**. Click this option to start customising your QAT.

When the **Word Options** window appears, you will see that there are two panes: the left-hand pane shows what is available for you to add to the left-hand pane which lists the options currently available on the QAT.

For standard Word functions, you scroll down through the items in the left-hand window, click on them once, click the **Add** button that is located between the two windows. The selected Word function will then appear in the right-hand window.

However, you cannot select multiple items and transfer them across in bulk, only one at a time.

Once a function is in the right-hand window, you can select it, then click on

the **Modify** button below and change the icon's appearance and associated text that appears on the QAT. This latter option will be useful as you can add a proper description instead of your macro's actual VBA name.

My 'RefreshEverything' function

In my Word Toolbox, the most popular function is the 'Refresh' button. Clicking it will go through your entire document, including the headers and footers, and refresh all of the fields, tables of contents, tables of figures and tables of authorities.

The code below is from my Word Toolbox, but with a few bits removed so that it is smaller. The code is not as scary as it may look as I have covered most of these types of VBA functionality in my previous articles.

Copy the code in Figure 2 into your Normal.dot or Normal.dotx file so that it is available in all of the new documents that you create. Remember to keep a regular backup of the macros that you put into these templates, by placing them in a text file for example. That way you can recover them should anything get corrupted or altered.

Adding macros to the QAT

Once a macro is associated with a button on the QAT, you will no longer have to go via the **Developer > Macros** menu on the ribbon to find your macro(s) that you have created.

Adding a macro to your QAT is just as simple as adding a Word function.

Using the same method as before, open up the Word Options dialog. On the left, there is a drop-down box at the top of the display labelled '**Choose commands from**'. Click the drop-down box below and change it to **Macros**.

The left now shows all of the macros that Word can currently find. This includes the current open document(s) and the current template.

If, whilst following this series, you have created your own library of macros that you use on a regular basis, they will all appear in the left-hand window.

```

Sub RefreshEverything()
Dim rngStory As Range
Dim objShp As Object
Dim toc As TableOfContents
Dim tof As TableOfFigures
Dim toa As TableOfAuthorities
Dim intRunCode As Integer

' Loops through and fixes all fields in a document (including headers
and footers)
For Each rngStory In ActiveDocument.StoryRanges
    Do
        On Error Resume Next
        rngStory.Fields.Update

        Select Case rngStory.StoryType

            ' Look in certain areas of document only
            Case wdEvenPagesHeaderStory, wdEvenPagesFooterStory,
wdPrimaryHeaderStory, wdPrimaryFooterStory,
wdFirstPageHeaderStory, wdFirstPageFooterStory

            ' If the header or footer StoryRange has one or more shapes then
the TextRange has to be checked individually.
                If rngStory.ShapeRange.Count > 0 Then
                    For Each objShp In rngStory.ShapeRange
                        If objShp.TextFrame.HasText Then
                            objShp.TextFrame.TextRange.Fields.Update
                        End If
                    Next objShp
                End If
            End Select

            On Error GoTo 0
            Set rngStory = rngStory.NextStoryRange
        Loop Until rngStory Is Nothing
    Next rngStory
    Set rngStory = Nothing

' Table of Contents, Figures and Authorities are updated here
With ActiveDocument
    If .TablesOfContents.Count >= 1 Then
        For Each toc In .TablesOfContents
            toc.Update
        Next toc
    End If

    If .TablesOfFigures.Count >= 1 Then
        For Each tof In .TablesOfFigures
            tof.Update
        Next tof
    End If

    If .TablesOfAuthorities.Count >= 1 Then
        For Each toa In .TablesOfAuthorities
            toa.Update
        Next toa
    End If
End With

End Sub

```

If you have only added the code included in this article, then it will just show RefreshEverything as the only selection available.

Select and move each of your macros that you want to see in your QAT across to the right-hand side window. If you want to keep them apart from other Word functions, you can select and insert a separator item.

Remember that for each item you place in the right-hand window, you can modify the icon displayed, along with the text shown.

For example, the macro accompanying this article is titled RefreshEverything. Maybe it would be better called Refresh? I will leave that decision up to you.

VBA form design

That will be the subject of the next few articles for *Communicator*.

For those itching to jump ahead, I will create a post on the ISTC forum and provide links to third party tutorials for those who want to get 'stuck in'. 



Mike Mee MISTC Currently looking for my next technical authoring opportunity in the software industry.
E: mugukmail@gmail.com

T: @Mug_UK

W: www.mikestoolbox.co.uk – my toolkit for Word 2007-2016

Safety-critical instructions must be easy to read

1. Use specification ASD-STE100 (ASD Simplified Technical English).
2. Use a checker to make sure that the text conforms to ASD-STE100.

Free trial of the TechScribe checker:
www.simplified-english.co.uk

Techscribe[®]

Figure 2. A macro to update all fields in a document