Creating Visual Basic Macros in MSWord® and MSExcel® to Manage Mainframe Output on a PC

Lynn Palmer, State of California, Berkeley, CA

ABSTRACT

You can record macros in MSWord® that allow you to take a file of your mainframe output and reformat it at the press of a button. This paper will present how to use SAS® PROC PRINTTO and MSVisual Basic® macros to change your 11" by 17" output to 8.5" by 11" and print off your PC printer. It will also cover how to set a macro to load a mainframe text file of your data into MSExcel®.

INTRODUCTION

Nothing is more frustrating than having data in one place and needing it in another. A small part of that problem is having your mainframe output print off a mainframe printer when you would like to fix it up a little and print it on your PC printer or email it in a MSWord file as an attachment. How do you do all those reformat changes with the push of a button? In this presentation, the "how to"s of how to use the MSVisual Basic behind all those MSWindows® applications will be discussed. For those of you who do not have Microsoft© products, the methods presented in this paper can be used with other products. You may have to research how your product sets up macros, but word processing and spread sheet products all have a way to reformat mainframe output with macros.

PROC PRINTTO

SAS PROC PRINTTO allows you to print your output or your log to a file. In our mainframe environment(IMB® VMS), it is necessary to create an empty data set the first time to hold a place(file) for PROC PRINTTO to write the data. The following code will produce an empty print file:

```
//JOB CARD
//STEP1 EXEC SAS
//FLEOT  DD DSN=USERID.MYSASOUT,
//       DISP=(,CATLG,DELETE),
//       UNIT=SYSDA,
//       SPACE=(3600,(100,15),RLSE),
//       DCB=(RECFM=VBA,LRECL=255,
//                  BLKSIZE=3600),
//       LABEL=EXPDT=99365
//SYSIN DD *
DATA _NULL_; FILE FLEOT; OUTPUT; RUN;
/*
//
```

Adjustments may need to be made in the lrecl, blksize, or recfm depending on your mainframe system's requirements for a print file(carriage return characters in the first column).

The syntax for PROC PRINTTO is as follows:

PROC PRINTTO *file* = 'userid.filename' *new*; RUN;

Print or *log* can be used instead of *file*. If you use *log*, your SAS log will be written to the file. Using *new* writes over existing file. If *new* is not specified, your output will be appended to the existing file. The RUN statement is needed with *log* to execute PROC PRINTTO before the next DATA or PROC step or one line of your log will be lost. By using the PROC PRINTTO statement with no options, your output or log will be redirected back to your system defaults. The file created with PROC PRINTTO can be transferred to your PC.

OTHER OPTIONS

There are some options which you can use to format the file when you are running your SAS program. Line size(LS) can be set to 80 for portrait and as long as 130

safely(longer lengths should be tested) for landscape printing. The page length can be controlled with PS to 60 lines for landscape. Also, NOCENTER NOPAGE NODATE can be used in the OPTIONS statement.

OPTIONS LS=80 PS=60 NOCENTER;

If you are using PROC FREQ, the option FORMCHARS (1,2,7) = ' ' (3 blanks) can be used, as well as LIST, NOCOL, NOROW, NOCUM, NOFREQ, NOPERCENT, and OUT=. FORMCHARS gives you control over the lines SAS uses in tables. OUT= allows you to write out a table(file with column headings and row descriptions). There are other options in other SAS procedures which allow you to control how the data in the OUT= or the PROC PRINTTO file is formatted.

TRANSFER TO PC

If you are using an emulation software to connect to your mainframe, the software has instructions on *receiving* files from the host computer(the mainframe). The best transfer protocol is IND$FLE. The file needs to be transferred as text with ANSII and CRLF specified. If the file length is greater than 80, the value of lrecl(logical record length) may need to be set and the record format specified as fixed(recfm=f or recfm(f)). The extension, *txt* is best to use on the file received.

If you have to, you can use MSDOS®. You need to be able to get to the internet or your intranet from your PC and know the IP address of the mainframe. The mainframe has to have an FTP server.

In DOS®, you FTP the IP address(*ftp your.main.frame*) and the mainframe will ask you for your userid(logon) called *user* and then your *password*. When you get the ftp> prompt, you issue the command: *recv filename_on_mainframe path\local_file _name ansii* . If you have problems, be sure the file names are correct. The command, *help* will give the FTP commands available on your system. You maybe able to change the directory from which you are receiving on the mainframe. If the FTP server doesn't put you in your directory, specify the file name in single quotes with your userid ('*user.filename_on_mainframe*'). The extention, *txt* works best on the file name assigned to the PC file. After the transfer is complete, enter the commands, *quit* and *exit* , to leave FTP and DOS and return to Windows.

OPENING FILE ON PC

To use your file with MSWord, go into the application. Click on File, on Open, find your file, and double click. The first thing you may see is lines wrapped around. Although you can quickly fix this problem by changing the font to Courier New at 8 pitch, there are still several other fixes needed for the page to look right. The font change alone requires 5 or so mouse moves and clicks. Bring a file into MSExcel is similar; there are several repeated actions which require time and need to be remembered. These repeated actions can be captured in a Visual Basic program or macro code within PC applications

VISUAL BASIC EDITOR

By clicking on Tools, Macro, and Visual Basic Editor, you can design and compile Visual Basic programs in MSWord and MSExcel. Why would you want to? Any activity on a computer that is repeated can be done by the computer. People are comfortable with their word processor. Work can get done faster, more consistently, and more correct. But before you scream, "I can't learn another programming language!", let me reassure you that you don't have to. Visual Basic macros can be recorded

DEVELOPING MACROS

Recording a macro is like using a video

camera. The first time the results may not be what you thought you were recording. It is best to plan the sequence that you are going to record and even practice, however unlike video taping correcting a part of the code is not impossible.

Open a test document in MSWord. Plan the changes that you would make. Close the test document *without* saving it and open it again. Click on Tools, Macro, and Record New Macro. A menu will open. At this point you can name the macro and choose to have it on the toolbar and/or assigned to a "hot" key on the keyboard. You can do these things later, also. When you click on OK, the menu closes and a little box with an icon for *stop* and an icon for *pause* appears. You then start the sequence of changes to your test document that you have planned. For example, click on Format, Font, select Courier New, 8 pitch, and OK. If you stop the macro recorder now, you will have a macro which changes the font and pitch of your document.

Other things that you might consider changing when downloading from the mainframe are the special symbols, such as vertical and horizontal lines. Also, you will want to identify the carriage control symbol for page breaks and replace it with MSWord's symbol. You click on Edit, and Replace. When the menu comes up, you complete your changes, choose Replace All, and OK.

You may want to go back at this point and name or change your macro. You click on Tools, Macro, Macros, and Edit. In Edit, you can change the code, change the name of the macro, or add to the macro. This is where you can paste code into a macro as well. Be sure to click on Debugg, and Compile Normal when you leave Edit.

On the Macros Menu, you have several options, such as Create, Delete, Step Into, etc. which allow you to manage your macros. Without a button on the toolbar or a hot key, you can submit your macros from this menu by selecting the desired macro and clicking on Run.

MSWORD MACRO

The following code is a macro for converting mainframe SAS output to a landscape document(8.5" by 11").

```
Sub Landscape()
' Original Macro by Joe Sousa for Word Perfect
' Macro to reformat HWDC output as landscape
' Version 3
   Selection.HomeKey Unit:=wdStory
   Selection.WholeStory
   With Selection.Font
      .Name = "Courier New"
      .Size = 8
      .Bold = False
      .Italic = False
      .Underline = wdUnderlineNone
      .StrikeThrough = False
      .DoubleStrikeThrough = False
      .Outline = False
      .Emboss = False
      .Shadow = False
      .Hidden = False
      .SmallCaps = False
      .AllCaps = False
      .ColorIndex = wdAuto
      .Engrave = False
      .Superscript = False
      .Subscript = False
      .Spacing = 0
      .Scaling = 100
      .Position = 0
      .Kerning = 0
      .Animation = wdAnimationNone
   End With
   Selection.HomeKey Unit:=wdStory,
Extend:=wdExtend
   Selection.Find.ClearFormatting
   Selection.Find.Replacement.ClearFormatting
   With Selection.Find
      .Text = "^p1"
      .Replacement.Text = "^m "
      .Forward = True
      .Wrap = wdFindContinue
      .Format = False
      .MatchCase = False
      .MatchWholeWord = False
      .MatchWildcards = False
      .MatchSoundsLike = False
      .MatchAllWordForms = False
   End With
```

```
    Selection.Find.Execute
Replace:=wdReplaceAll
    Selection.Find.ClearFormatting
    Selection.Find.Replacement.ClearFormatting
    With Selection.Find
        .Text = "]"
        .Replacement.Text = "|"
        .Forward = True
        .Wrap = wdFindContinue
        .Format = False
        .MatchCase = False
        .MatchWholeWord = False
        .MatchWildcards = False
        .MatchSoundsLike = False
        .MatchAllWordForms = False
    End With
    Selection.Find.Execute
Replace:=wdReplaceAll
    With ActiveDocument.PageSetup
        .LineNumbering.Active = False
        .Orientation = wdOrientLandscape
        .TopMargin = InchesToPoints(0.3)
        .BottomMargin = InchesToPoints(0.3)
        .LeftMargin = InchesToPoints(1#)
        .RightMargin = InchesToPoints(0.5)
        .Gutter = InchesToPoints(0)
        .HeaderDistance = InchesToPoints(0.5)
        .FooterDistance = InchesToPoints(0.5)
        .PageWidth = InchesToPoints(11)
        .PageHeight = InchesToPoints(8.5)
        .FirstPageTray = wdPrinterDefaultBin
        .OtherPagesTray = wdPrinterDefaultBin
        .SectionStart = wdSectionNewPage
        .OddAndEvenPagesHeaderFooter = False
        .DifferentFirstPageHeaderFooter = False
        .VerticalAlignment = wdAlignVerticalTop
        .SuppressEndnotes = False
        .MirrorMargins = False
    End With
    Selection.Delete Unit:=wdCharacter,
Count:=1
    Selection.TypeText Text:=" "
End Sub
```

By clicking on Tools, Macro, Macros, entering a name(Landscape), and clicking on Create, you will be able to type this code into a macro. Be sure that you only have one "sub" and one "end sub" statement in the macro.

BUTTON ON TOOL BAR

To get a button on the toolbar after a macro is created, you move the cursor to the toolbar and right click. This action will open a menu; at the bottom, click on Customize... The Customize window will open. Then pick Commands from the tabs. Next pick Macros from the list. All your macros will be listed in the window on the right. Highlight the one that you want on the toolbar. You grab the box and drag it to the toolbar

CONCLUSION

Macros allow us to do those things which are repetitious. Learning enough Visual Basic to be able to use MSWord and MSExcel to manage your mainframe output can remove some of the more tedious tasks of editing and reformatting. You will be able to do more work faster.

REFERENCES

SAS® Language, Reference, Version 6, First Edition, SAS Institute Inc., Cary, NC, USA, 1990.

SAS® Procedures Guide, Version 6, Third Edition, SAS Institute Inc., Cary, NC, USA, 1990.

TRADEMARKS

SAS is a registered trademark or trademark of SAS Institute Inc. in the USA and other countries. ®indicates USA registration.

MSWindows®, MSDOS®, MSVisual Basic®, MSWord® and MSExcel® are products of the Microsoft Corporation. Copyright © 1983-1997 Microsoft Corporation. All rights reserved.

AUTHOR

Lynn Palmer
Email: lpalmer@dhs.ca.gov,
Lpalmer363@aol.com,
Lipalmer@compuserve.com
Phone: (510) 540-3296