

## Paper 133-31

**Converting Multiple SAS® Output Files to Rich Text Format Automatically without Using ODS**

Ling Y. Chen, Rho, Inc., Newton, MA

**ABSTRACT**

A common task facing programmers today is to program in SAS, convert the output to a more commonly recognized file format such as the rich text format, and make some post-processing changes to the file. This task can be easily accomplished with the combined power of SAS macros and Microsoft Word macros. This paper walks you through an example of looping through a series of SAS output files (.lst), converting them to rich text format files (.rtf), and making some post-processing changes automatically using simple SAS codes and an MS Word macro.

**INTRODUCTION**

As a contract research organization doing clinical trials, we are often requested to produce output in the format specified by the client. Most frequently, the output format requested is rich text format. A lot of times, post-processing is needed in the output files before they can be sent to the client. For example, our paging program sometimes produces multiple left parentheses “(((Page x of y)” and “((Page x of y)”. These extra parentheses are removed in MS Word during post-processing. Another example would be header and footer information that need to be added to the output files. SAS ODS provides an easy way to produce .rtf files directly from SAS programs. However, it is not always clear how to customize the appearance of the .rtf files using ODS. The PROC TEMPLATE procedure is not always easy to use and there are other issues such as page breaks, text justification, etc. that have no easy answer when dealing with ODS. On the other hand, using MS Word we can easily accomplish these tasks.

**CONVERTING .LST FILES TO .RTF FILES USING MS WORD**

Converting .lst files to .rtf files is simple. To do that, first open the file from within MS Word. Click on [File]->[Open], double click on the .lst file you want, a pop-up dialog box called “Convert File” will appear, select “Text Only” from the list of options and the file will be read into Word as a text file (Figure 1). Once the file is read in, we can treat it as any Word document and do a number of things to it. For example, we can set paper orientation and margins, format texts to a particular font and size, or underline important passages, etc. When we are ready to save the file, click on [File]->[SaveAs], give the file a name, and choose ‘Rich Text Format (\*.rtf)’ from the drop down list of file extension options, and we are done.

Doing the above series of tasks is easy for one file. But what if there are many .lst files that need to be converted to .rtf files? As is frequently the case, for each project we handle, there are hundreds of .lst files that need to go through the same conversion process to become .rtf files. In our case, we need to do a series of tasks to each .lst file: 1) set top margin to 2 inches, bottom margin to 1 inch, left margin to 1 inch, and right margin to 1 inch, 2) set font to “Courier New” and font size to 9, 3) replace “(((Page” with “(Page”, 4) replace “((Page” with “(Page”, 5) save the file as an .rtf file with the same file name in the same directory where the file comes from, and 6) close the document. If we convert each file using the manual conversion process described in the previous paragraph, it becomes really tedious. Furthermore, we are likely to forget one of the 6 tasks needed and end up with very messy looking files.

To automate these tasks, MS Word macros come to the rescue.

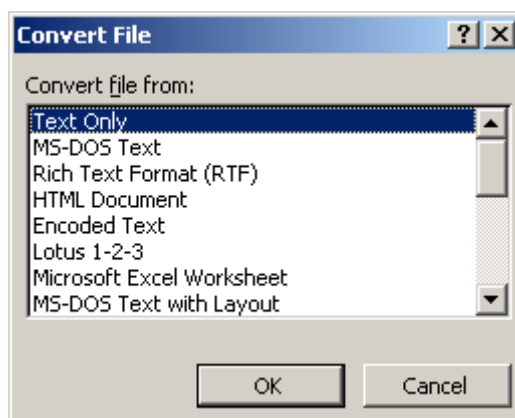


Figure 1

**WORD MACRO**

Most people are familiar with Microsoft Word and can navigate in an MS Word application with ease. However, a lot of people do not realize how powerful and easy to use MS Word macros are and don't fully utilize these macros to their advantage. A

macro is a series of commands grouped together to perform some tasks. For example, if we make a macro with commands for the 6 tasks outlined above, then every time we run this macro, these 6 tasks will be performed automatically in the order specified in the macro. Macros are written with Visual Basic, but if you don't know any Visual Basic, don't worry, you can easily 'record' a macro by using the Macro Recorder.

To record a macro that performs the 6 tasks we need, click on [Tools]->[Macro]->[Record New Macro], a pop-up dialog box called "Record Macro" will appear (Figure 2). It will ask for a name for the macro. We will name our macro "tortf". It then asks if we want this macro to be saved in Normal.dot, which means the macro will be available to all the Word documents. We will agree to that option. Notice that at the bottom of the dialog box, there is a description of the macro that contains information about when and who recorded the macro. Click OK and Word starts recording the macro. Then do tasks 1) to 6) like they would normally be done, and our actions are recorded by the Macro Recorder. At the end of the tasks, hit the stop button to stop the recording (Figure 3). The macro "tortf" has now been created. To run this macro, click [Tools]->[Macro]->[Macros], highlight the "tortf" macro and hit [Run] (Figure 4). The 6 tasks previously recorded in this macro will be run automatically.

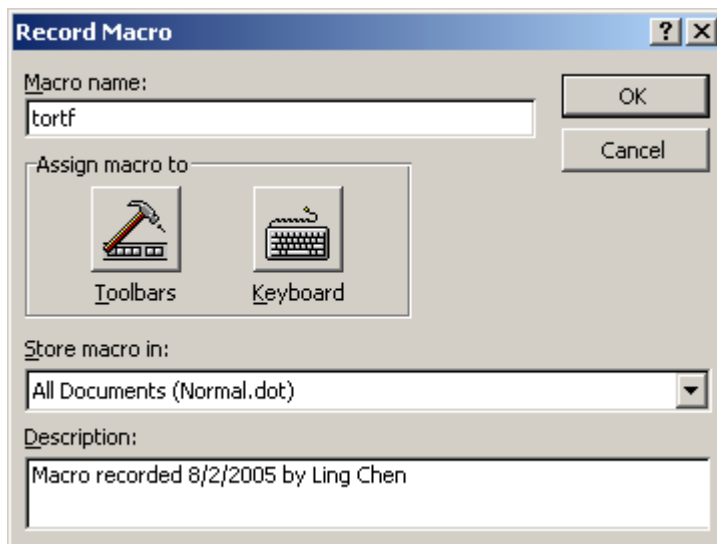


Figure 2



Figure 3

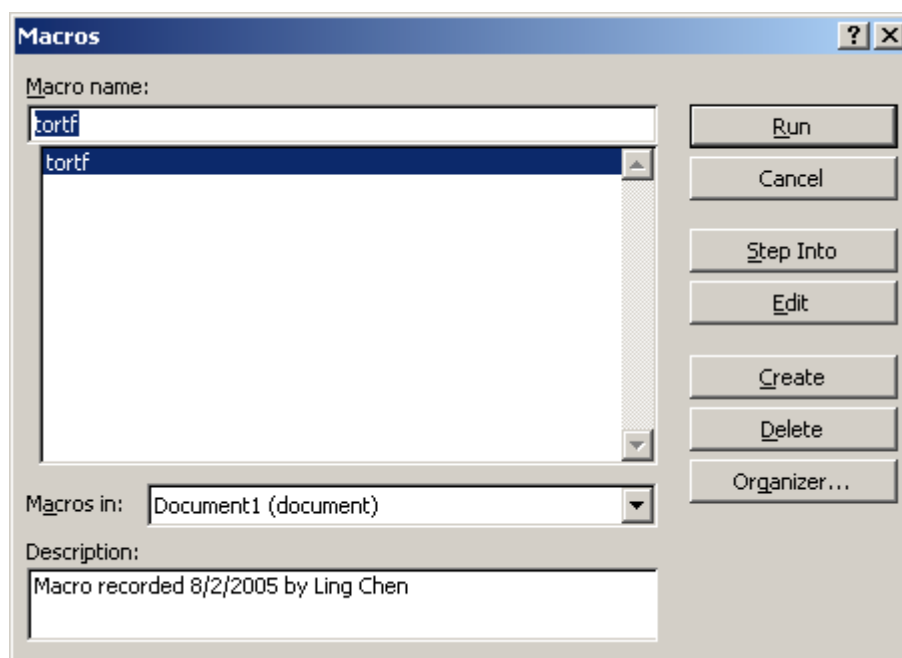


Figure 4

To view the Visual Basic code that was generated by the Macro Recorder, just click on [Tools]->[Macro]->[Macros], highlight the macro "torff", and click on the [Edit] button (Figure 4). The text of the macro will be shown as below. If any modification to the macro is desired, it can be done now. Notice that there are some texts in red, which are the changes I made to the macro in order to save the .lst file under the same name but with an .rtf extension in the same directory where the .lst file comes from.

```
' Macro recorded 8/2/2005 by Ling Chen
'
Dim Str As String
Str = ActiveDocument.Path & "\" & Left(ActiveDocument.Name, (Len(ActiveDocument.Name) - 3))

Selection.WholeStory
With ActiveDocument.PageSetup
.LineNumbering.Active = False
.Orientation = wdOrientLandscape
.TopMargin = InchesToPoints(2)
.BottomMargin = InchesToPoints(1)
.LeftMargin = InchesToPoints(1)
.RightMargin = InchesToPoints(1)
.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
.TwoPagesOnOne = False
.GutterPos = wdGutterPosLeft
End With
Selection.Font.Name = "Courier New"
Selection.Font.Size = 9
Selection.Find.ClearFormatting
Selection.Find.Replacement.ClearFormatting
With Selection.Find
.Text = "(((Page"
.Replacement.Text = "(Page"
.Forward = True
.Wrap = wdFindAsk
.Format = False
.MatchCase = False
.MatchWholeWord = False
.MatchWildcards = False
.MatchSoundsLike = False
.MatchAllWordForms = False
End With
Selection.Find.Execute Replace:=wdReplaceAll
With Selection.Find
.Text = "((Page"
.Replacement.Text = "(Page"
.Forward = True
.Wrap = wdFindContinue
.Format = False
.MatchCase = False
.MatchWholeWord = False
.MatchWildcards = False
.MatchSoundsLike = False
.MatchAllWordForms = False
End With
Selection.Find.Execute Replace:=wdReplaceAll

ActiveDocument.saveas FileName:=Str & ".rtf", FileFormat:=wdFormatRTF _
, LockComments:=False, Password:="", AddToRecentFiles:=True, _
WritePassword:="", ReadOnlyRecommended:=False, EmbedTrueTypeFonts:=False, _
SaveNativePictureFormat:=False, SaveFormsData:=False, SaveAsAOCELetter:= _
False
ActiveDocument.Close

End Sub
```

Define file name (without the "lst" extension)

Set margins

Set Font and size

Replace "(((Page" with "(Page"

Replace "((Page" with "(Page"

Save document with rtf extension

Close the .rtf document

## SAS MACRO FOR LOOPING THROUGH ALL THE .LST FILES

With the "tortf" Word macro, we can perform all the post-processing we want to an .lst file with one click of a button. However, for the hundreds of .lst files we want to process, we still need to open each file, find the "tortf" macro, and then click to run the macro. It is still a lot of work to process all the .lst files. So we write a SAS macro to loop through the files automatically.

First we prepare a text document that contains the paths and the names of the files, such as this one:

```
C:\CHEN\table1.lst
C:\CHEN\table2.lst
C:\CHEN\table3.lst
C:\CHEN\table4.lst
```

We will call it lstname.txt. Then we extract paths and names information from this file and store them in macro variables using SAS:

```
filename lstname "&pgmpath\lstname.txt";

data lstname;
  length var1 $200;
  infile lstname;
  input var1;
run;

data lstname2;
  set lstname;
  pathname=substr(var1, 1, 8);
  outname=substr(var1, 9);
run;

data _null_;
  set lstname2 nobs=nobs;
  call symput('out' || left(_n_), outname);
  call symput('path' || left(_n_), pathname);
  call symput('num', nobs);
run;
```

The result is a series of macro variables containing the .lst files' paths and names:

Macro Name	Macro Value
Path1	C:\CHEN\
Out1	Table1.lst
Path2	C:\CHEN\
Out2	Table2.lst
Etc.	

We can now use this information to connect to MS Word and run the "tortf" macro within Word.

```
%macro doit;

  %do i=1 %to &num;
    %let docname=&&out&i;
    %let path=&&path&i;
    %let wdmacro=tortf;

    filename cmnds dde 'winword|system';

    data _null_;
      x=sleep(2);
      RUN;

    data _null_;
      file cmnds;
      put '[FileOpen.Name=" ' "&path&docname" '"]';

      x=sleep(3);
      put '[ToolsMacro .Name = " ' "&wdmacro" ' ", .Run]';

    %end;

%mend doit;
```

We now have everything we need to convert as many .lst files as we want to .rtf files without having to open each file up, do 6 individual tasks to it, and close it. These tasks will be taken care of automatically for us. And no ODS code is required!

## CONCLUSION

Many tasks are repetitive and are best handled with automation. Utilizing the power of both the SAS macro language and MS Word macros, these tasks can be completed consistently, in less time, and with minimal amount of errors. In this example, we accomplished such a task, that is, converting a bunch of .lst files to .rtf files with some post-processing. This was accomplished without having to learn complicated ODS codes.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Please contact the author at:

Ling Yun Chen  
Rho, Inc.  
199 Wells Ave, Suite 302  
Newton, MA 02459  
617-965-8000 x26  
Fax: 617-965-8014  
[lchen@rhoworld.com](mailto:lchen@rhoworld.com)