# Combining First Page of Multiple RTF Outputs in SAS ® using Bookmark and VBA Macro

Ajay Gupta, PPD Inc, Morrisville, NC

## ABSTRACT

In order to expedite the review process and save programming time, it would be helpful if the programmer could create a document containing the first page of all the tables and listings. This will help both programmer/reviewer to review all tables and listings formats at once, and if required, make necessary updates. Traditionally, the programmer/reviewer has to open each table and listing to check the report formatting which can be tedious and prone to error, especially if there are too many reports. Unfortunately, SAS does not provide a function to get the first page of the rtf report. In my previous paper, I introduced a method to combine multiple rtf outputs. This paper will introduce a method to get the first page of each rtf document using bookmark, and later, combine the multiple documents (all first pages from respective reports) into one document in SAS using word basic commands and VBA macros via DDE (Dynamic Data Exchange). If needed, these functions can be use independently.

## INTRODUCTION

In the Pharmaceutical and CRO industries, to expedite the review process, business users frequently request to combine all tables and listings into one rtf document, particularly the first page. Since the rtf document can be used as a word document, it is easier for them to review and provide comments using the track change option. Having all the comments consolidated in one document will make the programmer's job easier and save time. This paper will introduce a method to get the first page of each rtf document, and later, combine multiple rtf documents in SAS using word basic commands via DDE.

Microsoft Word sets, and automatically updates, a number of reserved bookmarks. You can use these predefined bookmarks just as you use the ones that you place in documents, except that you do not have to set them and they are not listed on the Go To tab in the Find And Replace dialog box (Edit menu). Few predefined bookmarks are listed below.

| Bookmark | Description |
|---|---|
| \Section | Current section, including the break at the end of the section, if any. The current section contains the insertion point or selection. If the selection contains more than one section, the "\Section" bookmark is the first section in the selection. |
| \Page | Current page, including the break at the end of the page, if any. The current page contains the insertion point. If the current selection contains more than one page, the "\Page" bookmark is the first page of the selection. Note that if the insertion point or selection is in the last page of the document, the "\Page" bookmark does not include the final paragraph mark. |

**Table 1. List of Predefined Bookmarks**

This paper introduces a macro which will get the first page from each RTF document and later combine all the rtf documents into one final document. The entire process involves SAS, Word, and VBA macro which is automated by a SAS macro called `%Combine_RTF`.

## TECHNIQUE & MECHANISM

The program is divided into two parts; the first part will describe the process of getting the first page from each rtf document and the second part will describe the process of combining multiple rtf documents.

The general process of getting the first page of each rtf document is as follows:

1.  Get the list of all rtf documents from a given location using pipe command and further create macro variables for each file name.

2.  Count the number of pages and if it is more than one then go to step 3 or else copy report as is in temporary fold.

3. Open the first file and copy the predefined bookmark.

4. Open a new file and paste the predefined bookmark.

5. Execute VBA macros to update the format and remove page break and further save the file in different location.

6. Repeat steps 2, 3, 4 and 5 for all documents using do loop.

The general process of combining multiple rtf documents are as follows:

1. Get the list of all rtf documents from a given location using pipe command and further create macro variables for each file name.

2. Open the first file insert a section break at the end of the document.

3. Insert the second file.

4. Repeat steps 2 and 3 until all documents are combined using do loop.

5. Save the final document as Master.rtf and close.

To automate these steps, the DDE solution is leveraged to build the communication bridge between SAS and Word 2003. WordBasic commands (Microsoft Corporation, 1999) can be sent from SAS via DDE to enable SAS to take control of Word 2003.

## GETTING LIST OF RTF FILES AND CREATING MACRO VARIABLE FOR EACH FILE:

To get the list of all rtf documents from a given location, the pipe command is used as follows:

```
FILENAME DIRLIST PIPE "DIR ""&PATH"" /B ";

DATA dirlist1;
    INFILE dirlist LENGTH=RECLEN;
    INPUT FILENAME $varying1024. RECLEN;
    IF SCAN(filename,2,'.') IN ('doc' 'rtf');
    path="&path";
    filename1=STRIP(filename);
    filename=STRIP(path)||'\'||STRIP(filename);
RUN;
```

The above data step will get the lists of all the .rtf files to be combined from the given location and a variable filename combining the location and filename is created. Also, variable filename1 has all the individual filename.

Following code will count the total number of documents and also assign a macro variable for each document from the variable filename and filename1.

```
PROC SQL NOPRINT;
        SELECT COUNT(*) INTO:N FROM DIRLIST1;
        SELECT FILENAME INTO:FILE1-:FILE%CMPRES(&N) FROM DIRLIST1;
        SELECT FILENAME1 INTO:FILE21-:FILE2%CMPRES(&N)
        FROM DIRLIST1;
QUIT;
```

The above proc step will assign macro variables for each file and also count the total number of files in a given location.

## RECORD VBA MACRO IN WORD FOR PAGE FORMATTING AND TO DELETE THE PAGEBREAK:

For many, perhaps most SAS developers the easiest and fastest way to write VBA code is to first go into a Microsoft application (in this paper Microsoft Word is used as the application), turn on the macro recorder, perform the steps and functions desired, and then terminate the macro recorder. The end result is a stored VBA program (macro) that can be executed at a later date. This makes it possible for a SAS developer to automate tasks in the Microsoft application, and therefore vastly improve the functionality of an integrated system that takes advantage of the relative strengths of the SAS System and the Microsoft application.

**Steps to record VBA Macro for page formatting are as follows:**

1. Open Word.

2. Select Tools, Macro, and Record New Macro from the pull down menus.

3. Enter the name of macro as 'PAGE'.

4. Select Ok.

5. Select File and GoTo 'Page Setup'.

6. Select the desired attributes for e.g. left margin, right Margin, bottom margin, top margin, orientation, header, and footer from all the three respective tabs Margins, Paper, Layout.

7. Select Ok.

8. Select the stop recording button.

9. Close the Word.

**Steps to record VBA Macro to delete the page break are as follows:**

1. Open Word.

2. Select Tools, Macro, and Record New Macro from the pull down menus.

3. Enter the name of macro as 'PAGEBREAK'.

4. Select Ok.

5. Select the stop recording button.

6. Select Tools, Macro, and Macros from the pull down menus and select 'PAGEBREAK' in edit mode you will see the Visual Basic Editor.

7. Copy the page break macro code given below and save. Further, close the Word.

VBA code for Pagebreak macro is given below:

```
SUB PAGEBREAK()

        SELECTION.MOVEUP UNIT:=WDLINE, COUNT:=1, EXTEND:=WDEXTEND

        SELECTION.DELETE UNIT:=WDCHARACTER, COUNT:=1

END SUB
```

The above code will delete the page break from the document.

## GETTING THE FIRST PAGE FROM EACH RTF OUTPUT:

Prior to executing these statements, there are two system options needed NOXWAIT and NOXSYNC. The NOXWAIT option specifies that the DOS command prompt window disappears without one having to type EXIT when the process is finished, while the NOXSYNC specifies that the process should execute asynchronously. That is, control is returned immediately to the SAS System and the command continues executing without interfering with your SAS session.

In order for a client/server communication link to be established, both SAS and Word must be running. Therefore, for the first iteration, it is necessary to programmatically launch Word from a SAS session. There are several techniques available to launch Word from SAS. The simplest one is the following statement:

```
%LET RC=%SYSFUNC(SYSTEM(START WINWORD));
```

The above command is dependent on the completion of the previous command. Therefore, the SLEEP function can be used frequently in the SAS command/datastep that is dependent on the previous job finishing. These will avoid the error that occurs due to the delay in the execution of previous SAS command/datastep.

The syntax for the SLEEP function is given below:

```
DATA _NULL_;
    X=SLEEP(5);
RUN;
```

The above step will pause SAS session for five seconds.

To communicate with Word from SAS, the following FILENAME statement is used to establish the linkage between SAS and Word via the DDE triplet:

```
FILENAME WORD DDE 'WINWORD|SYSTEM';
```

Initially to store the first page of each file a folder 'Firstpage' is created using the following x command:

```
X "MKDIR ""&PATH.\FIRSTPAGE"" ";
```

Since, the predefined page bookmark will not work properly for the rtf documents having single page. The second step is to check the total number of pages in the rtf documents and if it is one then copy the rtf report directly to the temporary folder 'Firstpage'. Normally, the rtf documents have the string "PAGE X OF Y". The following code will count the total number of pages using string "NUMPAGES".

```
FILENAME ALLPAGE "&&FILE&I.";

DATA page;
    INFILE ALLPAGE LENGTH=RECLEN;
    INPUT var1 $varying1024. RECLEN;
    IF INDEX(var1,'numpages');
RUN;

PROC SQL NOPRINT;
      SELECT COUNT(*) INTO:PAGECOUNT FROM DIRLIST1;
QUIT;
```

The above code will first read the whole document and further count the number of pages.

If the above count is one then the rtf document is copied to the temporary folder Firstpage as is using the following x command below:

```
X "COPY ""&&FILE&I."" ""&PATH.\FIRSTPAGE""  ";
```

Now, the following data step will get the first page of rtf document. This data step will initially open the first file and copy the predefined bookmark 'Page'. After the bookmark is copied the original rtf document is close. Further, new word blank document is open and VBA macro 'Page' is executed to get the desired attributes and later predefined bookmark is pasted on the blank document. During this process an extra page break is added in the document along with the first page which can be remove using the VBA macro 'Pagebreak' (see code above). Finally, the new word file is saved in temporary folder 'Firstpage' and word session is close. The above process is repeated for all the rtf documents using do loop.

```
DATA _NULL_;
    FILE WORD;
    PUT '[FILEOPEN .NAME = "' "&&FILE&I." '"]';
    PUT '[EDITGOTO.DESTINATION="\PAGE"]';
    PUT '[EDITCOPY]';
    PUT '[FILECLOSE]';
    PUT '[FILENEWDEFAULT]';
    PUT '[PAGE( )]';
    PUT '[EDITPASTE]';
    PUT '[PAGEBREAK( )]';
    PUT '[FILESAVEAS .NAME = "' "&PATH\FIRSTPAGE\FILE2&I" '" ]';
    PUT '[FILESAVE]';
    PUT '[FILECLOSEALL]';
    PUT '[FILEEXIT]';
RUN;
```

In the above datastep `Fileopen` command will open the first file and predefined bookmark page is copied using `Editgoto.destination` and `Editcopy`. The new blank document is created using the `Filenewdefault` command and later the bookmark is copied using the `Editpaste` command. It is necessary to programmatically save and close the document with the `FileSave` and `FileClose` command. Further, `FileExit` command will exit from the Microsoft Word.

## COMBINING ALL RTF DOCUMENTS USING WORD BASIC COMMAND:

The same approach given above is used to start a new Word session:

```
%LET RC=%SYSFUNC(SYSTEM(START WINWORD));
```

To communicate with Word from SAS, the following FILENAME statement is used to establish the linkage between SAS and Word via the DDE triplet:

```
FILENAME WORD DDE 'WINWORD|SYSTEM';
```

The following data step will combine all the rtf documents located in 'Firstpage' folder into one file. This data step will initially open the first file and add a section break at the end of document. Once the section break is added, the next file will be inserted using word basic command. This process will be continuing until all files are combined. Once combined, the macro will save the final rtf file as Master.rtf (the location will be same as original rtf documents) and close the winword session.

```
DATA _NULL_;
    FILE WORD;
    PUT '[FILEOPEN .NAME = "' "&PATH\FIRSTPAGE\&FILE21." '"]';
    %DO I=2 %TO &N;
        PUT '[ENDOFDOCUMENT]';
        PUT '[INSERTBREAK .TYPE = 2]';
        PUT '[INSERTFILE .NAME ="' "&PATH\FIRSTPAGE\&&FILE2&I." '" ]';
    %END;
    PUT '[FILESAVEAS .NAME ="' "&PATH\&IN." '"]';
    PUT '[FILESAVE]';
    PUT '[FILECLOSEALL]';
    PUT '[FILEEXIT]';
RUN;
```

In the above macro, the `Fileopen` command will open the first file and then the cursor will move to the end of file by the `EndofDocument` command. Further, `InsertBreak` and `Insertfile` command will insert a section break and insert the file. It is necessary to programmatically save and close the document with the `FileSave` and `FileClose` commands. Further, the `FileExit` command will exit from Microsoft Word.

Further the temporary folder 'Firstpage' and its content are deleted using the following x commands below.

```
X "DEL /Q ""&PATH.\FIRSTPAGE"" ";
```

The above command will delete all the files from folder 'Firstpage'.

```
X "RMDIR ""&PATH.\FIRSTPAGE"" ";
```

The above command will delete the folder 'Firstpage'.

### %Combine_RTF

To facilitate and automate the above discussed steps from combining the first page of RTF files a SAS macro called `%Combine_RTF` was developed for SAS v9.1.3 or above (see Appendix for details). The user can easily extend the macro to fit other SAS versions.

There are only two keyword parameters:

**In**: Define the path and the name of the input Word file, e.g., `C:Master.rtf.`
**Path**: Production Location, for e.g., `C:\Demo`.

Below is the simple macro call to % Combine_RTF.

```
% Combine_RTF (in= Master.rtf, path=C:\Demo);
```

## CONCLUSION

Above solution demonstrates an innovative approach of combining first page of multiple RTF outputs without any formatting changes in the document. This solution requires minimal amount of coding and can be further modified to take advantage of other Word functions to enhance the RTF output.

## REFERENCES

Gupta Ajay, 2012. Watermarking and Combining Multiple RTF Outputs in SAS ®. Proceedings of the PharmaSUG 2012 Conference, paper CC06.

Luo Haiping, 2004. Let SAS® Tell Microsoft Word® to Collate. Proceedings of the Twenty-ninth Annual SAS Users Group International Conference, paper 34.

Roper, C.A. 2000. Using SAS and DDE to execute VBA macros in Microsoft Excel. Proceedings of the Twenty-Fifth Annual SAS Users Group International Conference, paper 98.

Bandi Ranganath, Kunduru Harini, 2011. Better Ways to Speak to Your System Using SAS: Automate Routine Tasks by using X, SYSTASK & FILENAME. Proceedings of the PharmaSUG 2011 Conference, paper AD06.

Microsoft Corporation. 2000. Word 95 WordBasic Help File.
http://www.microsoft.com/downloads/details.aspx?familyid=1a24b2a7-31ae-4b7c-a377-45a8e2c70ab2&displaylang=en

Office 2003 Editions: Access VBA Language Reference.
http://www.microsoft.com/downloads/details.aspx?displaylang=en&FamilyID=0447c5a0-5e58-4e69-b90e-c42ec7dbf887

http://msdn.microsoft.com/en-us/library/office/aa223084(v=office.11).aspx

http://msdn.microsoft.com/en-us/library/office/aa217319(v=office.10).aspx

http://msdn.microsoft.com/en-us/library/office/aa160407(v=office.10).aspx

## ACKNOWLEDGMENTS

## CONTACT INFORMATION

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

Name: Ajay Gupta, M.S
Enterprise: PPD Inc.
Address: 3900 Paramount Parkway
City, State ZIP: Morrisville, NC-27560
Work Phone: (919) -456-6461
E-mail: Ajay.gupta@ppdi.com, Ajaykailasgupta@aol.com

## APPENDIX :

```
%MACRO Combine_RTF(in=,path=);

        OPTIONS NOXWAIT NOXSYNC;

        X "MKDIR ""&PATH.\FIRSTPAGE"" ";

        FILENAME DIRLIST PIPE "DIR  ""&PATH"" /B ";

        DATA dirlist1;
            INFILE dirlist LENGTH=RECLEN;
            INPUT FILENAME $varying1024. RECLEN;
            IF SCAN(filename,2,'.') IN ('doc' 'rtf');
            path="&path";
            filename1=STRIP(filename);
            filename=STRIP(path)||'\'||STRIP(filename);
        RUN;

        PROC SQL NOPRINT;
            SELECT COUNT(*) INTO:N FROM DIRLIST1;
            SELECT FILENAME INTO:FILE1-:FILE%CMPRES(&N) FROM DIRLIST1;
            SELECT FILENAME1 INTO:FILE21-:FILE2%CMPRES(&N)
                FROM DIRLIST1;
        QUIT;

        %DO I=1 %TO &N;

        FILENAME ALLPAGE "&&FILE&I.";

        DATA page;
            INFILE ALLPAGE LENGTH=RECLEN;
            INPUT var1 $varying1024. RECLEN;
            IF INDEX(var1,'numpages');
        RUN;

        PROC SQL NOPRINT;
                SELECT COUNT(*) INTO:PAGECOUNT FROM ALLPAGE;
        QUIT;

        %IF &PAGECOUNT=1 %THEN %DO;

        X "COPY ""&&FILE&I."" ""&PATH.\FIRSTPAGE""  ";

        %END;

        %ELSE %DO;

        DATA _NULL_;
                X=SLEEP(15);
        RUN;

        %LET RC=%SYSFUNC(SYSTEM(START WINWORD));

        FILENAME WORD_F DDE 'WINWORD|SYSTEM';

        DATA _NULL_;
                X=SLEEP(15);
        RUN;

        DATA _NULL_;
            FILE WORD_F;
            PUT '[FILEOPEN .NAME = "' "&&FILE&I." '"]';
```

```
                    PUT '[EDITGOTO.DESTINATION="\PAGE"]';
                    PUT '[EDITCOPY]';
                    PUT '[FILECLOSE]';
                    PUT '[FILENEWDEFAULT]';
                    PUT '[PAGE( )]';
                    PUT '[EDITPASTE]';
                    PUT '[PAGEBREAK( )]';
                    PUT '[FILESAVEAS .NAME = "' "&PATH\FIRSTPAGE\&&FILE2&I." '" ]';
                    PUT '[FILESAVE]';
                    PUT '[FILECLOSEALL]';
                    PUT '[FILEEXIT]';
            RUN;

            %END;
            %END;

            DATA _NULL_;
                    X=SLEEP(5);
            RUN;

            %LET RC=%SYSFUNC(SYSTEM(START WINWORD));

            FILENAME WORD DDE 'WINWORD|SYSTEM';

            DATA _NULL_;
                    X=SLEEP(5);
            RUN;

            DATA _NULL_;
                FILE WORD;
                PUT '[FILEOPEN .NAME = "' "&PATH\FIRSTPAGE\&FILE21." '"]';
                %DO I=2 %TO &N;
                    PUT '[ENDOFDOCUMENT]';
                    PUT '[INSERTBREAK .TYPE = 2]';
                    PUT '[INSERTFILE .NAME ="' "&PATH\FIRSTPAGE\&&FILE2&I." '" ]';
                %END;
                PUT '[FILESAVEAS .NAME ="' "&PATH\&IN." '"]';
                PUT '[FILESAVE]';
                PUT '[FILECLOSEALL]';
                PUT '[FILEEXIT]';
            RUN;

            DATA _NULL_;
                X=SLEEP(5);
            RUN;

            X "DEL /Q ""&PATH.\FIRSTPAGE"" ";

            DATA _NULL_;
                X=SLEEP(5);
            RUN;

            X "RMDIR  ""&PATH.\FIRSTPAGE"" ";

        %MEND Combine_RTF;
```