

An Application of Flexible Operation Page Number Positioning Method for RTF Outputs

WenHsuan Wu, Pharmaceutical Co., Ltd, Taipei, Taiwan

ABSTRACT

In clinical trials, Rich Text Format (RTF) is a general format that is used to create outputs like tables, figures, and listings (TFL). After preparing these outputs, combine multiple RTF outputs into one, then maintain the original content, title, footnote and page number is an arduous challenge for programmers.

However, there are various methods to combine RTF outputs and there are no specific information or details on the operations of the page number. Hence, this paper summarizes an application of page numbering codes in RTF documents, including strength and weaknesses. In the meantime, to overcome all the drawbacks of the existing page numbering method, this paper presents a page number positioning (PNP) method that uses bookmarks to position the page number accurately with editing to RTF outputs, it can keep the operation flexible in various SAS generated RTF outputs. Break through the restrictions on other methods, PNP can maintain the page number without changing the original structure of RTF outputs when combining RTF outputs.

INTRODUCTION

Page numbering issue is encountered in combined the RTF output development, this paper assumes that the display of page number in each output is "PAGE k of n ", k is current page number and n is total page number. When combining multiple outputs by reading the outputs into a SAS data set and manipulating the RTF code¹, the total page numbers were originally intended to be disconnected in each output but continued with combining document, as shown in Table 1, the information of total page number in each output is lost; moreover, keeping the original presentation of output is core value in develop the tool of combine RTF outputs. However, to avoid this scenario, other approach suggests to use section page numbering to retain each individual output's total page number², but the structure of RTF document will be changed to one section per output, resulting in the title and footnote being limited due to it being unique for each section.

Output	Original page number	Combined page number
1 st	1 of 3	1 of 9
	2 of 3	2 of 9
	3 of 3	3 of 9
2 nd	1 of 2	1 of 9
	2 of 2	2 of 9
3 rd	1 of 4	1 of 9
	2 of 4	2 of 9
	3 of 4	3 of 9
	4 of 4	4 of 9

Table 1. Illustrates the display of current and total page number before and after combining. For example, combining three RTF output, first output has three pages, second output has two pages, and third output has four pages.

While several methods have been developed in recent years, SAS diversity of syntax generated various structure of the RTF outputs, including no pagination issue^{3,4} or dynamic title⁵, there is no general method that can be applied to those outputs, especially when the RTF structure is unpredictable. To clarify the applicable situation for each page numbering codes, this paper summarizes the conventional RTF page numbering codes, furthermore, introducing a general method—page number positioning (PNP) method that can keep the operation flexible in various RTF outputs generated via SAS.

SUMMARY OF PAGE NUMBERING CODES

There are four page numbering codes in the RTF documents, which will display as integer, the integer in RTF code is not entity number but dynamic with the current page. Here is a summary of RTF page numbering codes and each of their definitions.

The RTF code	Definition
PAGE	Display the current page number based on the entire RTF document.
NUMPAGES	Display the total page number based on the entire RTF document.
SECTIONPAGES	Display the total number of the current section based on the entire RTF document.
PAGEREF	Display the number of referenced bookmarked pages.

Table 2. Summary of page numbering codes.

Programmers usually use PAGE and NUMPAGES because both are the easiest ways to display page number and total page number for one output.

On the other hand, after combining RTF outputs, ones still need to keep the original page number, the current page number (PAGE) will be restarted to one at first page per output because of “\pgnrestart\pgnstarts1” in the RTF code, but the total page number (NUMPAGES) of per output will change from being based on individual outputs to the entire combined document, as shown in Table 1, so that the original RTF code (PAGE and NUMPAGES) is no longer applicable for the combined RTF document.

In this situation, SECTIONPAGES can directly divide different source outputs into different sections, so that the page number is no longer based on the entire combined document but its’ own section. However, when SAS compiles a new page in the RTF document, it will always create a new section—through the RTF code of “\sect\sectd” to do the section breaks, but now it needs to be modified to “\page\par” to do the page breaks, so that there is only one section per output—the number of sections in the entire combined document will be the same as the number of outputs being combined. This results in not only the page numbering code needing to be modified, but also the RTF document structure, operation detail is in Table 3.

Section Breaks (Original RTF structure)	Page Breaks
<pre>{\rtf1\ansi ... \sect...\pgnrestart\pgnstarts1... ... \sect\sectd ... \sect\sectd ... \pard}</pre>	<pre>{\rtf1\ansi ... \sect...\pgnrestart\pgnstarts1... ... \page\par ... \page\par ... \pard}</pre>

Table 3. The structure of RTF output in section breaks and page breaks. For example, a three pages RTF output in three sections and one section. Besides, “\pgnrestart\pgnstarts1” always exist at first page per output in the RTF code.

If the RTF structure has been changed, expand one section per page into one section per output, the title and footnote will be limited due to it being unique for each section in RTF specification, so the dynamic titles and footnotes as Figure 1 will be limited as Figure 2.

Table 14.3.4.1 Summary Statistics for Continuous Laboratory Values (Page 1 of 3)
 Analysis Set: Safety Analysis Set
 Parameter: ALANINE AMINOTRANSFERASE

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	17.6 (9.22)		82	18.0 (8.72)	
Week 1	83	18.0 (12.53)	0.2 (7.90)	80	20.9 (10.55)	2.8 (8.18)
Week 2	79	18.7 (12.91)	0.7 (8.66)	72	17.5 (7.66)	-0.7 (5.08)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

Table 14.3.4.1 Summary Statistics for Continuous Laboratory Values (Page 2 of 3)
 Analysis Set: Safety Analysis Set
 Parameter: ALBUMIN

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	39.8 (2.81)		82	39.8 (2.56)	
Week 1	83	38.9 (3.11)	-1.0 (2.49)	80	38.7 (3.17)	-1.1 (2.71)
Week 2	79	38.8 (3.29)	-1.0 (2.69)	72	38.6 (2.80)	-1.2 (2.66)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

Table 14.3.4.1 Summary Statistics for Continuous Laboratory Values (Page 3 of 3)
 Analysis Set: Safety Analysis Set
 Parameter: ALKALINE PHOSPHATASE

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	77.7 (58.11)		81	73.3 (20.72)	
Week 1	84	77.7 (69.50)	0.1 (15.51)	80	73.0 (21.87)	0.3 (10.25)
Week 2	82	78.0 (69.43)	0.4 (14.16)	72	73.2 (23.17)	-0.3 (11.12)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

Figure 1. The presentation of dynamic titles, using the section breaks to create a new page. There are three sections so there can has three titles.

Table 14.3.4.1 Summary Statistics for Continuous Laboratory Values (Page 1 of 3)
 Analysis Set: Safety Analysis Set
 Parameter: ALKALINE PHOSPHATASE

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	17.6 (9.22)		82	18.0 (8.72)	
Week 1	83	18.0 (12.53)	0.2 (7.90)	80	20.9 (10.55)	2.8 (8.18)
Week 2	79	18.7 (12.91)	0.7 (8.66)	72	17.5 (7.66)	-0.7 (5.08)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

PAGE: 1 Section: 1 Page 1 of 3 154 words English (United States)

Table 14.3.4.1 Summary Statistics for Continuous Laboratory Values (Page 2 of 3)
 Analysis Set: Safety Analysis Set
 Parameter: ALKALINE PHOSPHATASE

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	39.8 (2.81)		82	39.8 (2.56)	
Week 1	83	38.9 (3.11)	-1.0 (2.49)	80	38.7 (3.17)	-1.1 (2.71)
Week 2	79	38.8 (3.29)	-1.0 (2.69)	72	38.6 (2.80)	-1.2 (2.66)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

PAGE: 2 Section: 1 Page 2 of 3 154 words English (United States)

Table 14.3.4.1 Summary Statistics for Continuous Laboratory Values (Page 3 of 3)
 Analysis Set: Safety Analysis Set
 Parameter: ALKALINE PHOSPHATASE

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	77.7 (58.11)		81	73.3 (20.72)	
Week 1	84	77.7 (69.50)	0.1 (15.51)	80	73.0 (21.87)	0.3 (10.25)
Week 2	82	78.0 (69.43)	0.4 (14.16)	72	73.2 (23.17)	-0.3 (11.12)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

PAGE: 3 Section: 1 Page 3 of 3 154 words English (United States)

Figure 2. The presentation of limited dynamic title, using the page breaks to create a new page will limit the presentation of dynamic titles and footnotes. There is only one section so there has only one title, and it will always shows the last one.

Compared to SECTIONPAGES, PAGeref is a more flexible way to adjust page numbers, because it has independent page number, does not need to depend on current sections, only needs to create bookmarks to mark the position in advance, then calls out the page numbers of referenced bookmarked pages.

Therefore, without changing the original structure of the RTF output, PNP method uses bookmarks to position the page number accurately by applying PAGeref, even though there is too much content in a page, pagination issue occurs in the RTF document^{3,4}, but PAGeref can still positioning the correct page number. This is more compatible way to maintain the page numbering when combining the RTF outputs.

PAGE NUMBER POSITIONING (PNP) METHOD

The RTF code of PAGeref can use bookmarks to position the page number, it does not need to depend on the structure of the RTF document, merely call out the page number in the bookmarked page, so the collocation of bookmark and PAGeref is page number positioning (PNP) method.

PNP does not affect the RTF structure while retaining the flexibility of various structure of RTF outputs. The following instruction is the application of PNP for combined RTF documents step-by-step:

STEP1. CREATE BOOKMARKS

Bookmark is used to mark the place which the page number will be referenced. In the first step, mark bookmarks after the RTF code of “\sectd” at first page of each individual output as following RTF code:

```
{\bkmkstart BookmarkName}{\bkmkend BookmarkName}
```

For the convenience of explanation, the bookmark name of the first output is FILE1, the second output is FILE2, etc. In addition, the bookmark name can be changed but must be unique in the entire RTF document.

STEP2. DO NOT RESTART PAGE NUMBER AT THE FIRST PAGE PER OUTPUT

To position the current page number base on entire document, the page number need to be unique and continuous, so do not restart the page number of each output. In the second step, delete the RTF code of “\pgnrestart\pgnstarts1” in the row starting with “\sectd” at first page of each individual output, if there is not, omit this step.

STEP3. POSITIONING AND CALCULATING THE PAGE NUMBER

This paper assumes that there has n of outputs being combined (n is the total number of combined outputs) and use PAGEREF to position the page number in bookmark of FILE k (k is current sequence of outputs, $k = 1, \dots, n$).

To maintain original page number for each output after the combining, using the gap of page number between each output, three elements will be used:

1. PAGE: The current page number based on the entire RTF document.
2. FILE k : The current page number in first page of k output (i.e., the position of bookmark made in step one).
3. NUMPAGES: The total page number based on the entire RTF document.

For the display of current page number, use PAGE minus FILE k plus one; for the display of total page number, use FILE $k+1$ minus FILE k , but there is no output that follows for the last output, thus use NUMPAGES minus FILE k plus one. The illustration as Table 4.

Output	Current Page Number	Total Page Number
1 st	{= {PAGE} - {PAGEREF FILE1} + 1}	{= {PAGEREF FILE2} - {PAGEREF FILE1}}
2 nd	{= {PAGE} - {PAGEREF FILE2} + 1}	{= {PAGEREF FILE3} - {PAGEREF FILE2}}
3 rd	{= {PAGE} - {PAGEREF FILE3} + 1}	{= {NUMPAGES} - {PAGEREF FILE3} + 1}

Table 4. The RTF code of current and total page number for combining three outputs ($n = 3$)

APPLICATION

The bookmarks need to use PAGEREF to get referenced page number as Table 4, it shows that each calculation consist of control words or control symbols, start with equal sign, enclosed in braces. To add braces of RTF control symbols, use Ctrl+F9 in Word document or insert with following RTF code:

```
{\field{\*\fldinst{ ... }}}
```

In Word document, use Alt+F9 to toggle the field between function variable code and result as shown in Figure 3 and Figure 4.

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	17.6 (9.22)		82	18.0 (8.72)	
Week 1	83	18.0 (12.53)	0.2 (7.90)	80	20.9 (10.55)	2.8 (8.18)
Week 2	79	18.7 (12.91)	0.7 (8.66)	72	17.5 (7.66)	-0.7 (5.08)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

Figure 3. Function variable code.

Visit	Treatment A			Treatment B		
	N	Mean (SD)	CFB (SD)	N	Mean (SD)	CFB (SD)
Baseline	86	17.6 (9.22)		82	18.0 (8.72)	
Week 1	83	18.0 (12.53)	0.2 (7.90)	80	20.9 (10.55)	2.8 (8.18)
Week 2	79	18.7 (12.91)	0.7 (8.66)	72	17.5 (7.66)	-0.7 (5.08)

CFB = Change from baseline.
 Program Name: l-lbl.sas Program Run: 2021-03-24 9:13 Data Cut-off: 2021-03-22

Figure 4. Result of function variable code.

To display of page number as “PAGE *k* of *n*” in each output, here is an example of combining three outputs (*n* = 3), the RTF code of page number for the first output (*k* = 1) in combined RTF document by using PNP:

Page

```
{\field{\*\fldinst { ={\field{\*\fldinst { PAGE }}}-{\field{\*\fldinst { PAGEREF FILE1 }}}+1 }}}
```

of

```
{\field{\*\fldinst { ={\field{\*\fldinst { PAGEREF FILE2 }}}}-{\field{\*\fldinst { PAGEREF FILE1 }}} }}}
```

And then, for the second output (*k* = 2):

Page

```
{\field{\*\fldinst { ={\field{\*\fldinst { PAGE }}}-{\field{\*\fldinst { PAGEREF FILE2 }}}+1 }}}
```

of

```
{\field{\*\fldinst { ={\field{\*\fldinst { PAGEREF FILE3 }}}}-{\field{\*\fldinst { PAGEREF FILE2 }}} }}}
```

Using the gap of page number from current and next output to calculate total page number per output, but there are no follows for the third output (*k* = 3), hence substitute the total page number information of next output to NUMPAGES.

Page

```
{\field{\*\fldinst { ={\field{\*\fldinst { PAGE }}}}-{\field{\*\fldinst  
{ PAGEREF FILE3 }}}+1 }}
```

of

```
{\field{\*\fldinst { ={\field{\*\fldinst { NUMPAGES }}}}-{\field{\*\fldinst  
{ PAGEREF FILE3 }}}+1 }}
```

CONCLUSION

This paper introducing various page numbering codes and applicable status, it helps programmers clarify different definition to achieve the desired effects and results.

For widely applicable on diversity RTF outputs, PNP is more compatible way to position page number while combining RTF outputs. This approach is extraordinary with others, because of following reason:

1. Flexibility and maintaining original RTF structure, PNP apply PAGEREF and bookmark to calculate the page number is more flexible than others.
2. Accurate positioning page number to maintain original page number even though pagination issue occurs when too much content in a page.
3. Simply apply to diversity RTF outputs.

To summarize, various page numbering methods can be applied in developing RTF output, understanding how the RTF structure works and one's suitable method, make development easier to achieve.

REFERENCES

[1] Xie, Lugang (Larry), Ma, Jundong and Wang, Jie (Jerry). 2019. "SAS Utility to Combine RTF Outputs and Create a Multi-Level Bookmark Hierarchy and a Hyperlinked TOC" PharmaSUG China 2019. Available at <https://www.pharmasug.org/proceedings/2019/AD/PharmaSUG-2019-AD-104.pdf>

[2] Worley Jim. 2007. "Combining RTFs While Retaining Each Original Document's Page Numbering" PharmaSUG 2007. Available at <https://www.lexjansen.com/pharmasug/2007/cc/CC21.pdf>

[3] Yong Cao and Xuan Sun. 2017. "A convenient tool to check the pagination issue in TLF" PharmaSUG China 2017. Available at <https://www.lexjansen.com/pharmasug-cn/2017/AD/PharmaSUG-China-2017-AD06.pdf>

[4] Michael Yang and Vera Zhu. 2019. "Large batch processing of TLF pagination auto-inspection, 10 secs is enough" PharmaSUG China 2019. Available at <https://www.lexjansen.com/pharmasug-cn/2019/AD/Pharmasug-China-2019-AD30.pdf>

[5] Carpenter, Arthur L. 1998. "Better Titles: Using The #BYVAR and #BYVAL Title Options" The 23rd Annual SAS Users Group International Conference. Available at <https://support.sas.com/resources/papers/proceedings/proceedings/sugi23/Coders/p75.pdf>

ACKNOWLEDGMENTS

Thanks to KuenHung Lin for the comment and inspired me to write this paper.

CONTACT INFORMATION

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

Name: WenHsuan Wu (Sherry)
E-mail: s91046123@gmail.com

Any brand and product names are trademarks of their respective companies.