



#### Paper SM05

## **An Efficient Way to Combine RTF Files and Create Multi-Level Bookmarks and a Hyperlinked TOC**

Lugang Xie (Larry), Janssen R&D, Raritan, NJ, USA

Lee Armishaw, Janssen R&D, Leiden, Netherlands

### **ABSTRACT**

RTF is a popular format that most sponsors adopt to create tables, listings and figures in the pharmaceutical industry. However, there is an unmet need to concatenate the outputs into a presentable single RTF file independent of the computing system. This paper presents a simple SAS approach to concatenate any SAS generated RTF files, including both tables/listings and figures, and create a multi-level bookmark hierarchy and a hyperlinked TOC, without having to modify those macros/programs that create the individual outputs. It can be executed on any platforms with SAS installed.

### **INTRODUCTION**

#### **BACKGROUND AND APPROACH**

While PDF is the FDA supported format, RTF remains the most popular format used for production of individual reports in the pharmaceutical industry. Standard programs/macros to generate clinical reports in RTF format have been developed for years and as such manipulating RTF files has since been increasingly important. For example, concatenated RTF files with a navigation system (e.g. hyperlinked TOC or bookmarks) added enables switching between tables/graphs easier thereby reducing the time required for internal review. In addition, PDF format with a multi-level bookmark hierarchy is often required for agency submission, e.g. OSI BIMO listings. Some companies re-create these outputs in other formats, e.g. PDF or plain text, and created a bookmarked PDF file based on these files. It will be both time and cost efficient if sponsors take advantage of existing programs/macros and convert these RTF into a deliverable in the required format.

Several methods have been developed to handle this task over the past decade, but each of them has its drawback. For example:

1. Visual Basic approach: it must be executed in windows environment with MS Word installed; it is resource demanding and takes a considerably long time to handle a large number of RTF files.
2. ODS Document approach: it must have the code in the TLF programs to create ODS Document objects. It cannot handle any type of RTF files directly.
3. Other approaches: they either require inserting RTF code in the individual TLF programs or are unable to create a multi-level bookmark hierarchy.

This paper introduces a highly efficient, platform independent approach thought RTF code manipulation to concatenate any type of RTF files generated by SAS and create a multi-level bookmark hierarchy and hyperlinked Table of Contents.

### **METHODOLOGY AND DISCUSSION**

#### **REQUIREMENTS FOR INDIVIDUAL LISTINGS**

Each sponsor has their own specific requirements for RTF reports generated by SAS, but fortunately the basic requirements are actually very similar. The approach this paper provides accommodates the most popular requirements among sponsors:

1. Individual line listings for each site in RTF format are generated by SAS.
2. Titles are printed in the body rather than in the header section of a RTF file.
3. In addition to the listing title, a title that only contains site number is added to each listing. This feature is useful when bookmarks for site numbers (the second level bookmarks) are created. If site number titles are not available, a workaround is provided later in this paper.

### CONCATENATE INDIVIDUAL RTF LISTINGS

RTF files are largely plain text ASCII files, which enables us to read the files in to a SAS dataset and do manipulation for various purposes.

Each SAS generated RTF file includes three parts:

**Part 1:** the opening section, which includes the control words for file attributes (e.g. “{\rtf1}”), file attributes, font table, color table, style sheet, etc.

**Part 2:** the content section, which includes RTF code for header, footer and body. This part contains the content to be displayed. The starting RFT code for this section varies, depending on the way the RTF file is output, RTF template, etc. It usually starts immediately after {\stylesheet}. In the example displayed in Display 2, it starts with either “\paperw15840\paperh12240\landscape” if the page orientation is landscape or “\margl720\margr720” if it is portrait.

**Part 3:** the closing section, which includes a closing curly brace “}”, which may be preceded by one or more control words, e.g. “\par}”.

Here is an example of RTF code of a typical SAS generated RTF table output.

|               |                                                                                                                                                                                                                                                                                                                                                      |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Part 1</b> | <pre>{\rtf1\ansi\deff0\deflang1033 {\fonttbl{\f0\froman Times New Roman;} {\f1\fswiss\fcharset0\fprq2 Arial;}} {\stylesheet {\s0 Normal;} {\s1 \ql\widctlpar heading 1;} {\s2 \ql\widctlpar heading 2;} {\s3 \ql\widctlpar heading 3;}}</pre>                                                                                                        |
| <b>Part 2</b> | <pre>\paperw15840\paperh12240\landscape \margl720\margr720\margt2880\margb1800\headery1800\footery1800 {\header\pard\par} {\footer\pard \qr Page \chpgn of {\field{\*\fldinst NUMPAGES }} \par} \s15\qc\li0\ri0\sb0\sa0\widctlpar\rin0\lin0\itap0\qc\f0\fs24 \qc\f0\fs24\cgrid Table 1: Summary of Vital Signs \par \trowd\trleft0\trqc ... ..</pre> |
| <b>Part 3</b> | <pre>\pard\par}</pre>                                                                                                                                                                                                                                                                                                                                |

**Display 2: An example of RTF code of a typical SAS generated RTF table output**

If we remove part 3 of the first RTF file, part 1 of the last RTF file and both part 1 and part 3 of the rest of the RTF files in the middle, and set them all together with “\sect\sectd” inserted between files in a SAS dataset, and write the SAS dataset to a new RTF file, it will result in a single RTF file with all individual files concatenated. “\sect\sectd” starts a new page in an RTF file.

Display 3 illustrates how RTF files are concatenated.

| RTF File      | RTF Code                                                |
|---------------|---------------------------------------------------------|
| First         | Part 1<br>Part 2                                        |
| Second        | \sect\sectd<br>Part 2                                   |
| Third         | \sect\sectd<br>Part 2                                   |
| more<br>..... | \sect\sectd<br>Part 2<br>\sect\sectd<br>Part 2<br>..... |
| Last          | \sect\sectd<br>Part 2<br>Part 3                         |

**Display 3: RTF code for combined RTF file**

#### **MULTI-LEVEL BOOKMARKS**

A multi-level bookmark hierarchy can be created by adding RTF code around the titles. For example, as per FDA requirements for BIMO listings, three levels of bookmarks need to be created - the first level is study ID, the second level is site number, and the third level is each individual listing titles. This three-level bookmark hierarchy can be created by following two steps:

1. *Identify the title of each table.* The titles are typically surrounded with a consistent pattern of RTF control words, e.g. the title is always preceded by “\qc\fs24\cgrid” and followed by “\par” in the example in Display 2. Again, the surrounding control words vary from RTF file to RTF file, depending on the RTF template used to create these file. As long as they are created by SAS, they will be consistent across different tables.
2. *Insert style control words for the titles.* For example, add \s2 before the site title and \s0 after it, and the site title will appear at the second level in the bookmark hierarchy; add \s2 before the listing title and \s0 after it, and the listing title will appear at the their level in the bookmark hierarchy. This insertion should **ONLY** be done for the first page of each individual RTF file.

A cover page can be created containing study ID “ABC-P123” by adding the following code before Part 2 of the first file:

```
\sect\sectd
\pard\par\pard\plain\qc\fs24\b \s1 ABC-P123 \s0 \b0\par
```

Please note that \s1 and \s0 surround the ABC-P123, enabling the study ID appears at the first level in the bookmark hierarchy.

**Important:** the control words below define the headings and must be included in {\stylesheet} as illustrated in part 1 in Display 2. If the default stylesheet does not include this code, you need to add it.

```
{\s0 Normal;}
{\s1 \ql\widctlpar heading 1;}
{\s2 \ql\widctlpar heading 2;}
{\s3 \ql\widctlpar heading 3;}
```

If the site number titles are not available, there is a workaround that can be considered to adopt – insert into the combined RTF file a site number title before the table titles with font size being 0 and color being white.

#### HYPER-LINKED TABLE OF CONTENTS

Table of Contents can also be created by adding RTF control words. Reviewers may find it useful in various situations. This paper does not go into details on this topic, but the basic idea of how it can be created is briefly explained as follows:

1. Insert the code below between \sect\sectd and part 2 for each file as illustrated in Display 3. Note that the n in IDXn is the order number of individual RTF file. It should be incremental.

```
{\bkmkstart IDXn}{\bmkend IDXn}
```

For example:

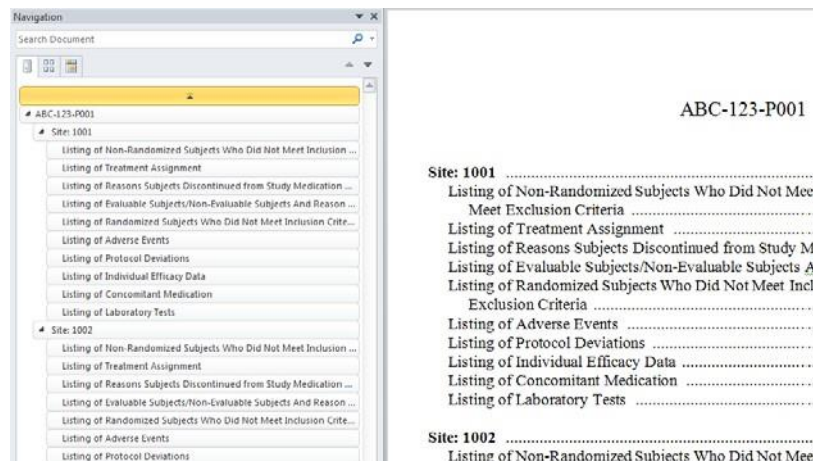
```
{\bkmkstart IDX1}{\bmkend IDX1}
{\bkmkstart IDX2}{\bmkend IDX2}
... .. etc.
```

2. On the first page created earlier, which only contains the study ID, insert a Word field for each title. The titles can be extracted from individual RTF files, given the fact that the surrounding RTF code of the titles should be consistent from file to file. For example:

```
\fi-200\li400{\cf0{\field{\*\fldinst { HYPERLINK \l "IDX1"}}
{\fldrslt {\cf0 Table 1: Summary of Vital Signs }}}
\tab{\field{\*\fldinst { PAGEREF IDX1 \h}}{\fldrslt{NUMPAGES}}}}\par
\pard\plain\ql\li0\ri0\nowidctlpar\fauto\rin0\lin0\itap0\fl\fs20
```

```
\fi-200\li400{\cf0{\field{\*\fldinst { HYPERLINK \l "IDX2"}}
{\fldrslt {\cf0 Table 2: Summary of Adverse Events }}}
\tab{\field{\*\fldinst { PAGEREF IDX2 \h}}{\fldrslt{NUMPAGES}}}}\par
\pard\plain\ql\li0\ri0\nowidctlpar\fauto\rin0\lin0\itap0\fl\fs20
```

... .. etc.



The screenshot shows a Word document with a navigation pane on the left and a table of contents on the right. The navigation pane lists sections for Site 1001 and Site 1002. The table of contents lists sections for Site: 1001 and Site: 1002 with corresponding page numbers.

| Section                                                             | Page |
|---------------------------------------------------------------------|------|
| Site: 1001                                                          | 1    |
| Listing of Non-Randomized Subjects Who Did Not Meet Inclusion ...   | 1    |
| Listing of Treatment Assignment                                     | 1    |
| Listing of Reasons Subjects Discontinued from Study Medication ...  | 1    |
| Listing of Evaluable Subjects/Non-Evaluable Subjects And Reason ... | 1    |
| Listing of Randomized Subjects Who Did Not Meet Inclusion Crite...  | 1    |
| Listing of Adverse Events                                           | 1    |
| Listing of Protocol Deviations                                      | 1    |
| Listing of Individual Efficacy Data                                 | 1    |
| Listing of Concomitant Medication                                   | 1    |
| Listing of Laboratory Tests                                         | 1    |
| Site: 1002                                                          | 1    |
| Listing of Non-Randomized Subjects Who Did Not Meet Inclusion ...   | 1    |
| Listing of Treatment Assignment                                     | 1    |
| Listing of Reasons Subjects Discontinued from Study Medication ...  | 1    |
| Listing of Evaluable Subjects/Non-Evaluable Subjects And Reason ... | 1    |
| Listing of Randomized Subjects Who Did Not Meet Inclusion Crite...  | 1    |
| Listing of Adverse Events                                           | 1    |
| Listing of Protocol Deviations                                      | 1    |

Display 4: An example of combined RTF file with a three-level bookmark hierarchy and a TOC

## CONCLUSION

This approach is especially useful if sponsors want to take advantage of the existing SAS macros/programs already available to create RTF outputs and want to avoid duplicate efforts without extra effort or having to invest in extra software.

While other options are also available, there are distinct advantages for this SAS approach to combine RTF files:

1. Efficiency. It is more efficient than the VBA approach. It takes considerably less time to create a combined RTF file using SAS than using VBA scripts.
2. Low maintenance. As it is implemented using SAS, every SAS programmer with basic RTF knowledge should be able to update the macro. This is especially important when customization is required.
3. Fully automated process to create combined RTF files. No manual process is involved. It can be added to the end of the batch job for report generation. Combined files can be automatically generated after all the individual outputs are created.

It is worth noting that there is a file size limit of 500 MB for any MS Word document. In other words, if an RTF file is larger than 500 MB in size, it cannot be opened by MS Word. However, if you create a few smaller combined RTF files and convert them into multiple PDF files, it would get around the issue. It should not be a concern for internal review, and FDA accepted this approach for recent submissions.

## REFERENCES

1. Meeker-O'Connell, Ann and Kassim, Sean. "Overview of Information Requested by CDER's Office of Scientific Investigation (OSI) for NDA and BLA Submissions. October 2012". Available at <https://collaboration.fda.gov/p44198603/>
2. "Portable Document Format (Pdf) Specifications". September, 2014. Available at <http://www.fda.gov/downloads/Drugs/DevelopmentApprovalProcess/FormsSubmissionRequirements/ElectronicSubmissions/UCM163565.pdf>
3. "Field codes in Word". Available at <http://www.fda.gov/downloads/Drugs/DevelopmentApprovalProcess/FormsSubmissionRequirements/ElectronicSubmissions/UCM163565.pdf>
4. "Script to convert Word document to PDF file (PowerShell)". Available at <https://gallery.technet.microsoft.com/office/Script-to-convert-Word-f702844d>
5. "Rich Text Format (RTF) Specification Version 1.9.1.". Available at <https://www.microsoft.com/en-us/download/confirmation.aspx?id=10725>
6. Xie, Lugang (Larry) and Yan, Zhiping. "A Fully Automated Approach to Concatenate RTF outputs and Create TOC". PharmaSUG China 2014. Available at <http://www.lexjansen.com/pharmasug-cn/2015/DV/PharmaSUG-China-2015-DV28.pdf>

## CONTACT INFORMATION

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

Author Name: Lugang Xie (Larry)  
Company: Janssen R&D  
Address: 920 US Highway 202  
City, Postcode: Raritan, NJ 08869, USA  
Work Phone: +1 (908) 927-7415  
E-mail: [lxie9@its.jnj.com](mailto:lxie9@its.jnj.com)

Author Name: Lee Armishaw  
Company: Janssen R&D  
Address: Einsteinweg 101  
City, Postcode: 2333 CB Leiden, Netherlands  
Work Phone: +31 713053952  
E-mail: [LArmisha@ITS.JNJ.com](mailto:LArmisha@ITS.JNJ.com)

Brand and product names are trademarks of their respective companies.