PharmaSUG 2023 - Paper SI-235

Secret SAS – You Can Write eCTD Using ODS Word in SAS

Pete Lund, Looking Glass Analytics;
Anusha Minnikanti, Fred Hutchinson Cancer Center;
Calins Alphonse, Fred Hutchinson Cancer Center;
Julie Stofel, Fred Hutchinson Cancer Center

ABSTRACT

The Electronic Common Technical Document (eCTD) standard is the standard format for submitting applications, amendments, supplements, and reports to FDA's Center for Drug Evaluation and Research (CDER) and Center for Biologics Evaluation and Research (CBER) (https://www.fda.gov/drugs/electronic-regulatory-submission-and-review/electronic-common-technical-document-ectd). In a nutshell, the eCTD standard requires linked information in tables of contents, tables and figures. It also requires specific formatting for the sections of the report and the tables of contents. These requirements are typically met with manually inserting report information into a Microsoft® Word document and using the formatting capabilities of Word, a time consuming and error-prone process. However, the SAS® ODS Word destination (pre-production) available in version 9.4 M7 provides functionality to create reports in eCTD format with only a few manual steps, all point-and-click, in Word when complete.

In this paper we present a number of available, but undocumented, ways to use the SAS® ODS Word destination to programmatically produce eCTD-formatted reports. A set of macros was developed to 1) insert headers that can be turned into Table of Contents entries, 2) produce captions for tables or figures that can be turned into Table of Figures or Table of Tables entries, and 3) insert blocks of text into the document. In addition, methods to include dynamic page numbers, bulleted lists, formatting text (bold, italic, hyperlinks, etc.), and inserting empty table templates (for manual update) will be discussed.

The text of this paper was created with SAS using the ODS Word destination. The program that created this document and all the data and macros it uses are available on GitHub. Click here. (https://github.com/GoofyPeteLGAN/SAS-eCTD)

INTRODUCTION

The example program (eCTD_WordCreateDocument) included in the GitHub link above shows the use of a number of macros that were developed to assist in creating eCTD-formatted Microsoft Word documents. The code and macros are internally documented, so the details can be understood by reading the macros themselves to see how they work. As such, the specific details of how the macros work are not discussed in this paper, which is focused on how to use the macros to produce an eCTD document.

SETUP

The ODS Word destination is used to create this document. Like other ODS destinations, there is an ODS sandwich which tells SAS what information is passed to that destination. In this case, it is:

```
ods word file=<path\filename.docx> <options()> <style=>;
     <sas code that produces output>
ods word close;
```

The default ODS Style for this destination is, not surprisingly, Word. If you are using PROC TEMPLATE to create your own style definition, it is best practice to parent based on Styles.Word.

Unlike other style definitions, the Word style has a SCHEME statement that sets up color and font definitions for the Word theme. The scheme defines default light and dark text and background colors, hyperlink colors, and accent colors. In addition, there are definitions for heading and body fonts. If you want to change any of those values, it is best to copy and paste the default SCHEME statement and just tweak the pieces you need to. The PROC TEMPLATE used to define the Word Style for this document is in the setup program (eCTD_WordSetup).

PLACING TEXT IN THE DOCUMENT

There are two macros used to place text in the document: %eCTD_Text and %BulletedList. These macros use PROC ODSTEXT and PROC ODSLIST to create the text for the document.

PLAIN TEXT

To get plain blocks of text, the %eCTD_Text macro is used. The first parameter is the text to be displayed. It is usually best to include this within a macro quoting function, like %nrstr() or %quote(), to mask characters that might interfere with macro processing - like commas, quotes, percent signs and ampersands.

Other parameters control the style of the text, the number of blank lines to be inserted before and after the text and whether to include a header in the line above the text. The current parameters are:

- The first parameter is the text to be displayed.
- Style= parameter is the ODS style element used to format the text. The style element referenced here must be defined in the style template being used. The default is BodyStyle.
- LNL= defines the number of leading new lines to insert before the text (default is 0).
- TNL= defines the number of trailing new lines to insert before the text (default is 1).
- Header= parameter defines the header to display above the text. The default is xxx and is ignored.

Paragraph Header

This is an example of a block of text with a header. All parameters, except for the text and header to be displayed, have default values.

```
%eCTD_Text(%quote(This is an example of a block of text with a header. All
parameters, except for the text and header to be displayed, have default
values.), Header = %quote(Paragraph Header));
```

BULLETED LISTS

Often it is necessary to display a bulleted list of items. The %BulletedList macro helps accomplish that. The parameters include:

- ListText= parameter contains a caret-delimited list of items to be displayed in the list.
- BulletType= parameter gives the type of bullet to display before each item. The default is circle but can be disc or decimal or any type that PROC ODSLIST supports.
- Header= parameter inserts a header line before the list.
- HeaderWeight= parameter sets the header to bold (medium is the default).
- NestAfter= and NestThru= parameters allow for nesting of items:

  - at this point, only a single set of indentation is allowed.
  - the bullet type is the same as the main list.

Future enhancements will allow for more list control.

CREATING A TABLE OF CONTENTS

The eCTD standard requires a table of contents (TOC) that includes page numbers, headers, and is hyperlinked to the appropriate section in the document. The %eCTD_Heading macro creates that.

The %eCTD_Heading macro is set up to work for a particular style of headers: a section number (e.g., 4.2) and the section header text. Indentation and font size are determined from the value of the section number. The concept could easily be adapted for other header styles. Word TOCs are built from text that is defined as "headers" such as H1, H2, H3, etc. and the macro uses PROC ODSTEXT to create the header lines.

eCTD_Heading Parameters:

- HeadingText1= contains the section number. The font size and header type (H1, H2, etc.) are determined by the number of periods in this value. For example, HeadingText1= 4. and HeadingText1= 4.1 will differ in font size and header type.
- HeadingText2= contains the section header text.
- HFS= can contain a font size to override the size determined by the section number value.
- Tabs= the number of tab characters to place between the section number and the section header text (default is 1).
- Indent= number of spaces to indent from the left edge of the page. This will override the value determined by the section number.
- TrailingLine= number of blank lines to insert after the header (default is 1).

AUTOMATED TOC

There are several options that can be used on the ODS WORD statement and in the Style template that will create a Table of Contents automatically. However, there is an ODS WORD bug, in all releases of 9.4, that often makes this not the best option, especially in a dynamic situation where you are not sure as to what the TOC entries will be. If a header text spans multiple lines, it appears correctly in the body of the document, but only the last line is displayed in the TOC.

The SAS Global Forum paper (Kelley, 2020) covers these options and issues in detail. Due to the problems with automated TOC, we have opted for manual TOC.

MANUAL TOC

The multiple-line header text issue is not really an issue if you just leave the TOC-related options off the ODS WORD statement. Still set up the headers just as you would have, with the %eCTD_Heading macro, and they can be picked up by Word to build a table of contents.

INSERTING THE TOC IN WORD

Once the document has been created, open it in Microsoft Word and place the cursor where you want the table of contents to be inserted. Click on References Tab on the Microsoft Word menu and then on the arrow under Table of Contents on the far left of the ribbon. Click on Automatic Table 1 or Automatic Table 2 and the TOC will appear. You can then change the contents header and insert page breaks, if needed.

If manual changes are made to the document, go back to the TOC and click near the top. An Update Table option will appear. Clicking that will help you automatically update the page numbering in the TOC.

CAPTIONS FOR TABLES AND FIGURES

The %eCTD_Caption macro is used to add captions below tables or figures. Like the TOC entries created with the %eCTD_Heading macro, hidden field codes are placed in the document that can be converted to Word captions and added to a Table of Tables (and/or Figures).

eCTD_Caption Parameters:

- CaptionText parameter contains the text of the caption. Do not include "Table" or "Figure" or a number, like "Table 1", as these will be automatically generated by Word.
- CaptionType= either Table (the default) or Figure can be chosen.
- LeadingLines= the number of lines to insert between the table or figure and the caption (default is 1).

Below is the block of code used to add Captions for Tables and Figures. All parameters, except for “CaptionText” have default values.

```
%eCTD_Caption(List of Girls in the Class,CaptionType=Table);

%eCTD_Caption(League and Division in SASHELP.Baseball,CaptionType=Figure);
```

TABLE CAPTIONS

Use CaptionType=Table or leave this parameter out as it is the default, to create a table caption. The word Table and the sequential number will be inserted automatically.

Here are a couple of examples:

Table 1

Name	Sex	Age	Height	Weight
Alice	F	13	56.5	84.0
Barbara	F	13	65.3	98.0
Carol	F	14	62.8	102.5
Jane	F	12	59.8	84.5
Janet	F	15	62.5	112.5
Joyce	F	11	51.3	50.5
Judy	F	14	64.3	90.0
Louise	F	12	56.3	77.0
Mary	F	15	66.5	112.0

Table 1 : List of Girls in the Class

Table 2

Name	Sex	Age	Height	Weight
Alfred	M	14	69.0	112.5
Henry	M	14	63.5	102.5
James	M	12	57.3	83.0
Jeffrey	M	13	62.5	84.0

Name	Sex	Age	Height	Weight
John	M	12	59.0	99.5
Philip	M	16	72.0	150.0
Robert	M	12	64.8	128.0
Ronald	M	15	67.0	133.0
Thomas	M	11	57.5	85.0
William	M	15	66.5	112.0

Table 2 : List of Boys in the Class

FIGURE CAPTIONS

Use CaptionType=Figure to insert a figure caption. The word Figure and the sequential number will be inserted automatically.

Here are a couple of examples:

Figure 1

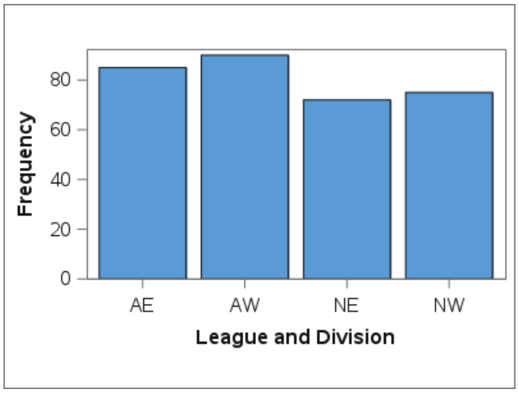

Figure 1 : League and Division in SASHELP.Baseball

Figure 2

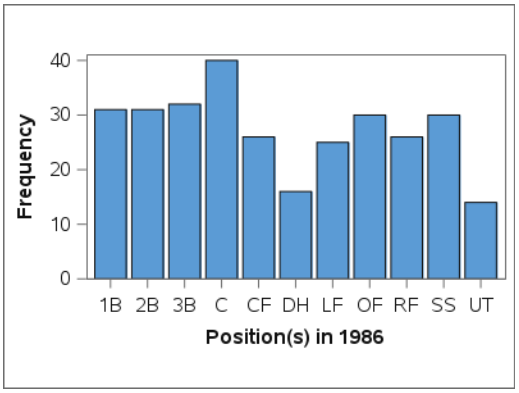

Figure 2 : Player Position in SASHELP.Baseball

GETTING A TABLE OF TABLES/CAPTIONS TO APPEAR IN WORD

When you first open the document in Word, you will notice that the words "Table" and/or "Figure" appear in the captions without any numbers. Below are a few quick manual steps to be followed to complete the caption process.

For the caption numbers to appear:

1. Hit CTRL+A to select the entire document.
2. Hit F9 and the table and figure numbers will appear.

Additional steps to create a Table of Tables or Table of Figures:

1. Position the cursor where you want the table to appear.
2. Click on References.
3. Click on Insert Table of Figures in the Captions portion of the ribbon.
4. On the Table of Figures tab, select either Figure or Table in the Caption label dropdown.
5. Make sure Include label and number is checked.
6. Click OK.

USING MICROSOFT WORD FIELD CODES

Field codes in a Microsoft Word document are placeholders for items that might change, like page or caption numbers. When the document is viewed, the field codes are not seen, but are resolved to the meaningful text they represent. Most often these are inserted into the document using the point-and-click options in Word, so you never really see them. However, you can place them into the document programmatically from SAS as well.

To insert a field code with SAS, the syntax is {run {<field code>} <\switches>}. Note that it is an ODS escape sequence, hence must be preceded by the ODS ESCAPECHAR that has been set (here we are using a tilde, ~). A couple of field codes have been used so far in creating this document. For example, to insert page numbers in the footer, the following FOOTNOTE statement uses two field codes:

```
footnote1 "~{run {PAGE}} of ~{run {NUMPAGES}}";
```

Also, the caption numbers for the tables and figures use the sequence field code in a P statement in PROC ODSTEXT:

```
p "~{run Table {SEQ Table \* ARABIC}: List of Girls in the Class}";
```

Note that the table number is not specified; it is calculated by Word based on how many tables were captioned prior to the current table. If you manually enter another table before the current table, the number on this table would change appropriately.

The list of available field codes can be accessed here: List of field codes in Word - Microsoft Support (https://support.microsoft.com/en-us/office/list-of-field-codes-in-word-1ad6d91a-55a7-4a8d-b535-cf7888659a51).

TEXT FORMATTING

You can use ODS style attributes in text sent to the Word destination. For example, the string

```
"This is ~{style [fontweight=bold]bold} and ~{style
[textdecoration=underline]underlined}."
```

would show up in the document as

This is **bold** and underlined.

Adding the style attributes involves a lot of typing, so we sometimes use a SAS macro that scans text and converts simple HTML-like field codes to the appropriate syntax. For example, the string above would be "This is <b>bold</b> and <u>underlined</u>." and the macro would convert the field codes to the STYLE attributes and the text would be rendered in Word as shown above. The macro, %CleanText, is included in the GitHub repository.

EMPTY TABLES

In the CAPTIONS FOR TABLES AND FIGURES Section, we learnt that SAS procedure output can be placed in a Word document (the class lists used PROC PRINT and the baseball graphs used PROC SGPLOT). Often it is useful to place an empty table in the document that will be manually updated with status or other information that is not in data but is maintained elsewhere.

One easy way to do this is with the DATA step Report Writing Interface (RWI). This is an object-oriented component of the DATA step language that allows for, among other things, creating tables. Unlike procedures, which need data input, the RWI tables can be created with no data at all. The following sample table was generated from a DATA _NULL_ step, in the %MakeBlankStatusTable macro. Refer to the macro for further details.

Table 3

Version Number	Version Date	Program Manager

Table 3 : RWI Blank Table

More information on using the RWI can be found here. (https://support.sas.com/resources/papers/proceedings16/2240-2016.pdf).

DATA DRIVEN USE OF THE MACROS

For the most part, the SAS program that generated this paper uses text that is hard coded in the calls to the macros referenced above. Obviously, this is not the easiest way to maintain and modify a document. In practice, we use the eCTD (and other) macros to generate Word documents where most of the text is in a standardized format spreadsheet. Versions of the spreadsheet can be maintained by different projects and then the same SAS program can be used to create different documents in the same format across projects.

Any method that you choose could be used to generate the calls to the eCTD macros from a data source containing the text for the document. For example, in a DATA step with CALL EXECUTE statements that contain macro calls or using CALL SYMPUT or PROC SQL SELECT INTO, to create macro variables that can be used in the macro calls.

In this paper, the %LoadTitleAndAbstract macro programmatically reads the paper title, author information and the text of the abstract from a spreadsheet. Also, the %DataDrivenSectionText macro loaded the text for this section of the paper, and the next, from the same spreadsheet. Refer to that code for some examples of creating document text from data.

FINALIZING THE PAPER

The code provided in the GitHub link and discussed here will produce a complete eCTD document with minimal effort. However, a few manual adjustments will be required to finalize the placement and appearance of table of contents, figures and tables.

CONCLUSION

Whether you need to create an eCTD-formatted document or not, the ODS Word destination is an extremely useful tool for the reporting toolbox. Since it is pre-production as of SAS 9.4 M7, the official documentation is sparse. We hope that the tidbits presented here will help make it work better for you!

Please let us know if you make any additional discoveries!

REFERENCES

Huntley, Scott, ODS: It's Not Just for Tables Anymore. Using Formatted Text and Lists in Your Reports, Proceedings of the SAS Global Forum 2020 Conference. Cary, NC: SAS Institute Inc. Available at https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2020/4526-2020.pdf

Kelley, David, An Introduction to the ODS Destination for Word, Proceedings of the SAS Global Forum 2019 Conference. Cary, NC: SAS Institute Inc. Available at https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2019/3235-2019.pdf

Kelley, David, Customize Your Table of Contents with the ODS Destination for Word, Proceedings of the SAS Global Forum 2020 Conference. Cary, NC: SAS Institute Inc. Available at https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2020/4265-2020.pdf

Lund, Pete, Have it Your Way: Creating Reports with the Data Step Report Writing Interface, Proceedings of the SAS Global Forum 2016 Conference. Cary, NC: SAS Institute Inc. Available at https://support.sas.com/resources/papers/proceedings16/2240-2016.pdf

Microsoft, List of Field Codes in Word, Available at https://support.microsoft.com/en-us/office/list-of-field-codes-in-word-1ad6d91a-55a7-4a8d-b535-cf7888659a51

SAS Institute Inc, ODS Word Statement (preproduction), Available at https://go.documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/odsug/p10mxeb6wxqfjgn1p5u0w4t8qf20.htm

SAS Institute Inc, Report Writing Interface Documentation, Available at https://documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/odsadvug/n1egaqwnkqh19wn1gfxbyx0hsqof.htm

ACKNOWLEDGEMENTS

The authors would like to say thanks to David Kelley of SAS for his unending willingness to answer questions about the ODS Word destination.

CONTACT INFORMATION

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

Pete Lund
Looking Glass Analytics
Olympia, WA
pete.lund@lgan.com

Anusha Minnikanti
Fred Hutchinson Cancer Center
Seattle, WA
aminnika@fredhutch.org

Calins Alphonse
Fred Hutchinson Cancer Center
Seattle, WA
calphons@fredhutch.org

Julie Stofel
Fred Hutchinson Cancer Center
Seattle, WA
julie@fredhutch.org