

Business & Decision Life Sciences

How SAS Can Help with RTF Files

Cyrille Cartier

18 SEPTEMBER 2014



Agenda

A vertical line on the left side of the slide contains six light beige circles. Each circle is connected to a horizontal blue bar that contains text. The circles are connected by a thin line that has a small gap at the top and bottom.

Background

Functionality

Generation of SAS program txt file

Generation of SAS program RTF file

Generation of Combined RTF file

Summary and Q&A

Background



- BDLS was contacted by a client, as they were unsure if SAS programs needed to be included within a submission package or not.
- Our initial thoughts were they were not necessary if the define.xml was well constructed, but we also checked the FDA guidance to see what it actually states...

Background



- FDA CDER draft guidance for [Study Data Technical Confirmation Guide](#) contains the statements:
 - 4.1.7.2. *General Considerations (page 12):*

“One of the expected benefits of analysis datasets that conform to ADaM is that they simplify the programming steps necessary for performing an analysis. ADaM datasets should be derived from the data contained in the SDTM datasets. There are features built into the ADaM standard that promote traceability from analysis results to ADaM datasets and from ADaM datasets to SDTM. **Sponsors who provide the software programs used to create ADaM datasets help reviewers to better understand how the datasets were created** (see section 4.1.7.8). Each analysis dataset that is shown in the define.xml file should be described.”

- 4.1.7.8. *Software Programs (page 14)*:

“Any submitted programs (scripts) generated by an analysis tool should be provided as ASCII text files or PDF files and should include sufficient documentation to allow a reviewer to understand the submitted programs. If the programs created by the analysis tool use a file extension other than .txt, the file name should include the native file extension generated by the analysis tool for the ASCII text program files (e.g., adsl_r.txt or adsl_sas.txt).”

Background



- Almost helpful! Whilst the document does not explicitly require the programs, it certainly opens the possibility of them being expected or required. So, to be sure we asked FDA, who responded...

"it is true the programs might not be required, but this question needs to be discussed at Pre-NDA meeting and determined by review division. Sometimes the programs are needed upon request, which means if review division has difficulty reviewing the packages from what sponsors have submitted, they could request programs as well!"

- Our client opted to have the programs included in their submission, both in a combined PDF, and as ASCII text files.

Agenda

A vertical line on the left side of the slide contains six light beige circles. Each circle is connected to a horizontal blue bar that contains text. The circles are connected by a thin line that has a small gap at the top and bottom.

Background

Functionality

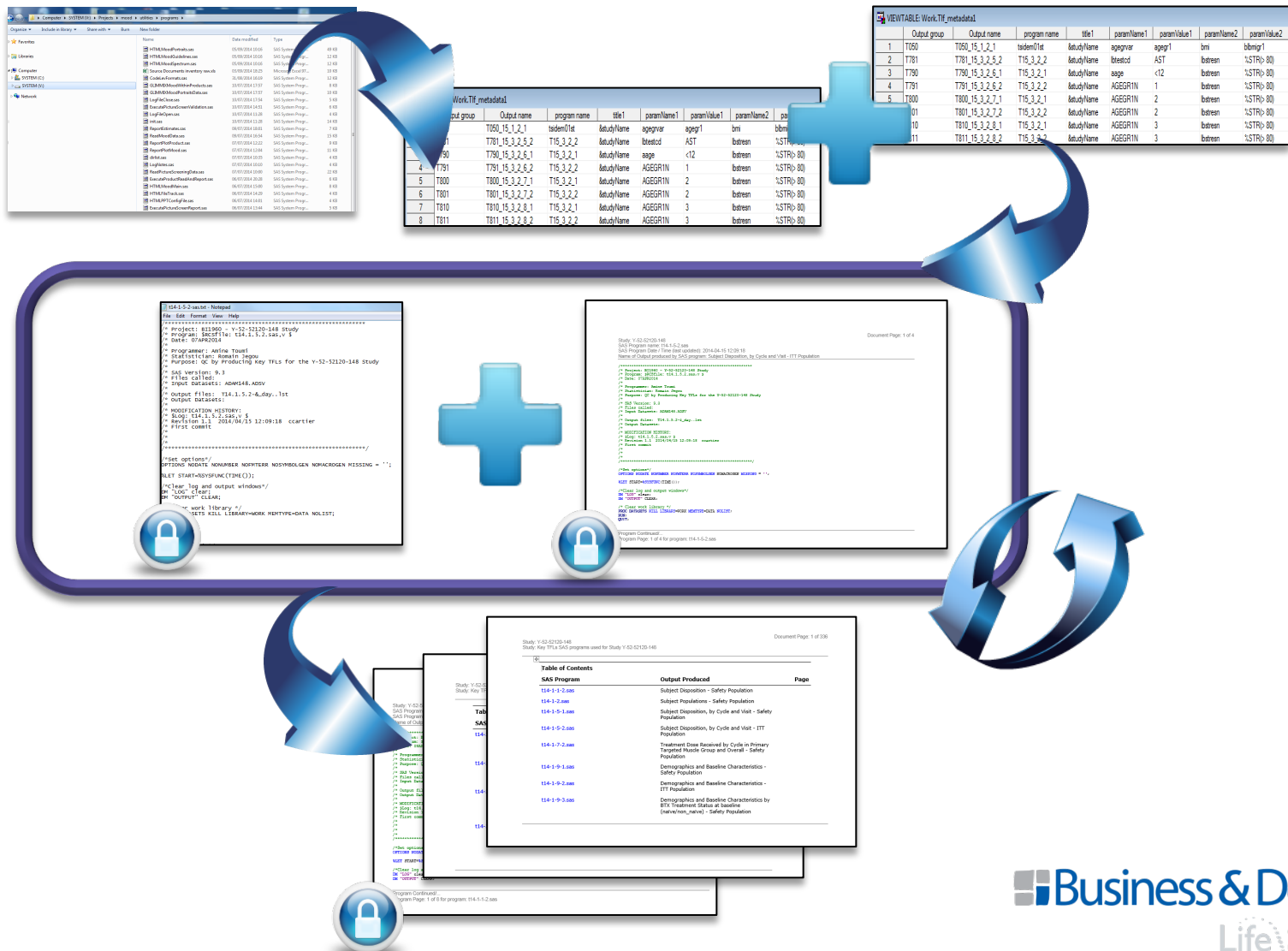
Generation of SAS program txt file

Generation of SAS program RTF file

Generation of Combined RTF file

Summary and Q&A

Functionality



Agenda

A vertical timeline diagram on the left side of the slide. It consists of six light beige circles connected by a thin grey line. The circles are positioned to the left of the agenda items, with the top circle aligned with "Background" and the bottom circle aligned with "Summary and Q&A".

Background

Functionality

Generation of SAS program txt file

Generation of SAS program RTF file

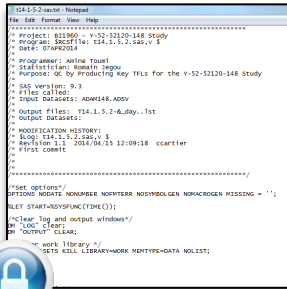
Generation of Combined RTF file

Summary and Q&A

Generation of SAS program txt file



- Code loops round directory listing dataset, and assigns output directory and modified program name* to macro variables: _progrnameRTF, _outdirRTF, _outdirTXT.
- On first loops checks if sub-folder exists, if not creates it, then write out file and set as read-only



```
*** check if sub-folders RTF/TXT folders exist, if not then create them *** ;
%IF %SYSFUNC(FILEEXIST(&_outdirRTF)) = 0 %THEN X MD "&_outdirRTF" ;;
%IF %SYSFUNC(FILEEXIST(&_outdirTXT)) = 0 %THEN X MD "&_outdirTXT" ;;

*** copy SAS program to TXT file *** ;
X COPY "&_progfilename" "%STR(&_outdirTXT.&_progrnameTXT.)" ;

*** host command to set file to read-only *** ;
X ATTRIB +r "%STR(&_outdirTXT.&_progrnameTXT.)" ;
```

- TXT files easy to created, but only black and white and contain no additional information / metadata about the program.

Generation of SAS program txt file: Naming convention



- ICH [Electronic Common Technical Document Specification](#) states:
- Name is a token composed of the following characters:
 - Letters "a" to "z" [U+0061 to U+007A].
 - Digits "0" to "9" [U+0030 to U+0039].
 - "-" [HYPHEN-MINUS, U+002D].

Examples of correct names (only the name without the extension):

- part-b
- myfile
- hello

Examples of incorrect names (only the name without the extension):

- part a (' ' ; SPACE is not allowed)
- myfile.xml ('.' ; FULL STOP is not allowed)
- hello:pdf (':' ; COLON is not allowed)
- part_a ('_', **LOW LINE is not allowed**)
- Parta (UPPERCASE is not allowed)

Generation of SAS program txt file: Naming convention

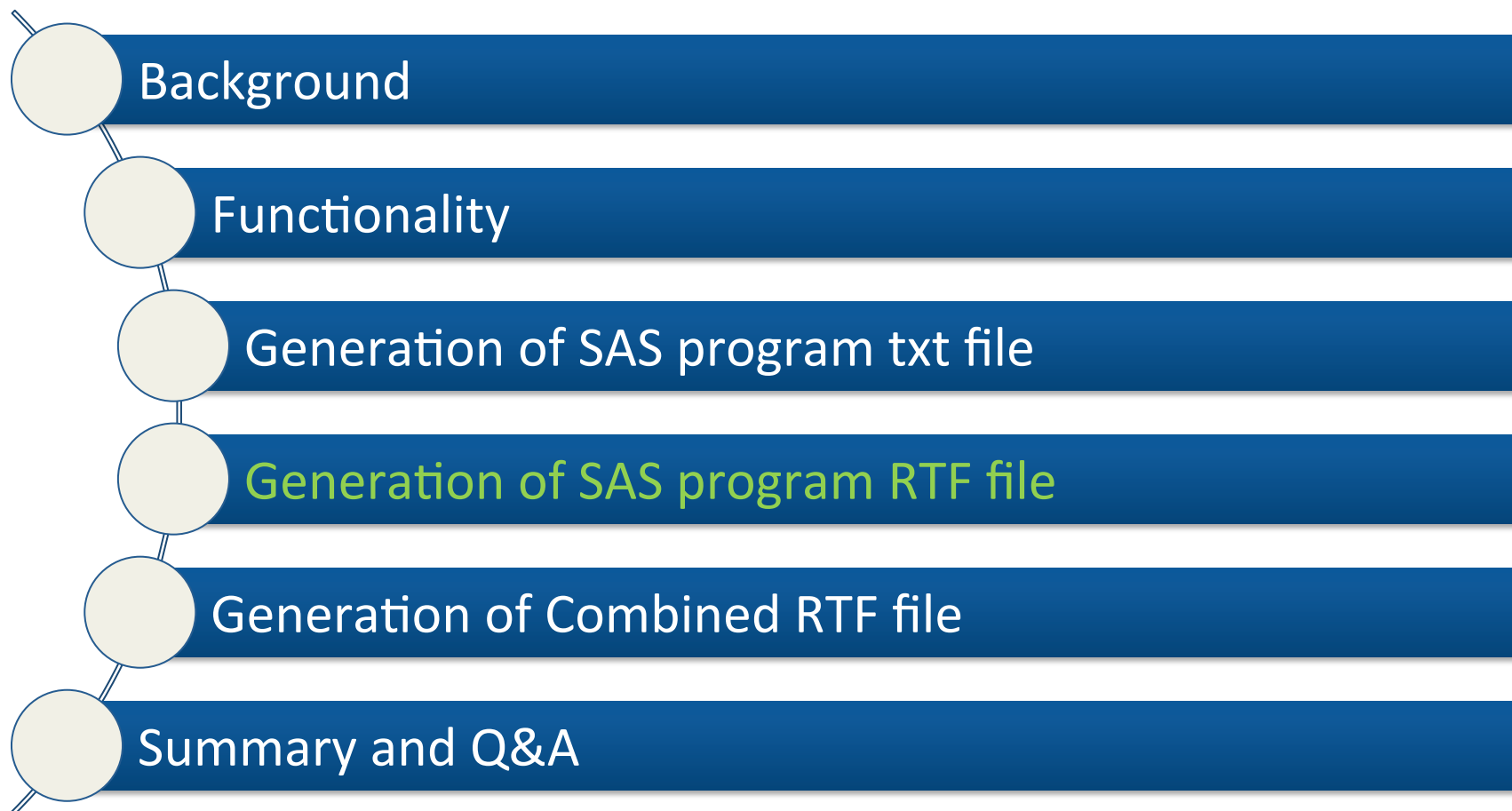


Decision

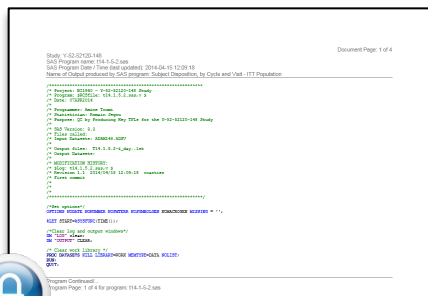
- For submission of SAS programs it was agreed:
 - SAS file names should end '.txt'.
 - *Underscores and blanks spaces should not be present in the file name*
 - *file names should be in lower case,*

e.g. SAS program: T_123-Example.sas would be saved as : t-123-example-sas.txt

Agenda



Generation of SAS program RTF file



- Code loops round directory listing:
- Opens program in SAS, Copies the code to Windows Clipboard, Closes the program using SAS DM commands*

```
*** open and copy program from SAS Enhanced Editor *** ;  
  DM 'whostedit;  
    INCLUDE "&_progfilename";  
    EDCMD selectall;  
    EDCMD copy;  
    EDCMD winclose;  
  
  ;
```

By copying the code from SAS Enhanced Editor, SAS syntax colour coding is preserved when pasted into MS Word document...

* Code snippet originally developed by Dirk Spruck, Accovion, ref: [PhUSE paper](#)



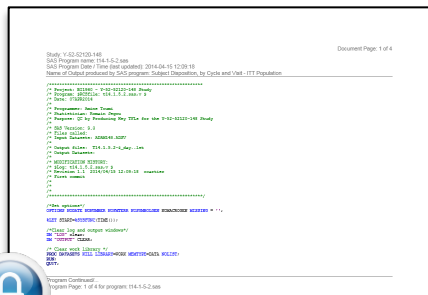
- ```
*** Add book mark *** ;
PUT '[EditBookmark.Name="" "%STR(&_prognameshrt)" "', .Add]';

*** copy the program to Word *** ;
PUT '[EndOfDocument]' ;
PUT '[EditPaste]' ;
```

# Generation of SAS program RTF file



- Code loops round directory listing:
- Creates new Word document, sets page size, orientation, margin size.



```
%LET _papersize = letter ; *** paper size A4/Letter
*** ;
%LET _orientation = landscape ; *** page orientation
*** ;
%LET _unit = in ; *** units
*** ;
%LET _LeftMargin = 1 ;
%LET _RightMargin = 1 ;
%LET _TopMargin = 1 ;
%LET _BottomMargin = 1 ;
:
:
%IF %UPCASE(&_papersize) = A4 %THEN %DO ;
 %LET _longside = 11.69 ;
 %LET _shortside = 8.27 ;
%END ;
%ELSE %IF %UPCASE(&_papersize) = LETTER %THEN %DO ;
 %LET _longside = 11 ;
 %LET _shortside = 8.5 ;
%END ;
%IF %UPCASE(&_orientation) = LANDSCAPE %THEN %DO ;
 %LET _WordOrient = 1 ;
 %LET _PageWidth = &_longside ;
 %LET _PageHeight = &_shortside ;
%END ;
```



```
%MACRO word_pagesetup () ;
 %PUT start of SAS macro: &sysmacroname ;
 *** format document *** ;
 PUT '[EditSelectAll]' ;
 *** Set page orientation: 1 = landscape, 2 = portrait *** ;
 PUT '[FilePageSetup.Orientation = "&_WordOrient"]' ;
 *** Set page Width *** ;
 PUT '[FilePageSetup.PageWidth = "&_PageWidth" ' + Chr$(34)]' ;
 *** Set page Height *** ;
 PUT '[FilePageSetup.PageHeight = "&_PageHeight" ' + Chr$(34)]' ;
 *** Set top margin *** ;
 PUT '[FilePageSetup.TopMargin = "&_TopMargin" ' + Chr$(34)]' ;
 *** Set bottom of page at 1 inch *** ;
 PUT '[FilePageSetup.BottomMargin = "&_BottomMargin" ' + Chr$(34)]' ;
 *** Set left margin *** ;
 PUT '[FilePageSetup.LeftMargin = "&_LeftMargin" ' + Chr$(34)]' ;
 *** Set right margin *** ;
 PUT '[FilePageSetup.RightMargin = "&_RightMargin" ' + Chr$(34)]' ;
 *** Set Top of Page *** ;
 PUT '[FilePageSetup.HeaderDistance = "1.5" + Chr$(34)]' ;
 %PUT end of SAS macro: &sysmacroname ;
%MEND word_pagesetup ;
```

# Generation of SAS program RTF file



Code loops round directory listing:  
Inserts information into header

Document Page: 1 of 4

Study:  
SAS Program name: t14-1-5-2.sas  
SAS Program Date / Time (last updated): 2014-04-15 12:09:18  
Name of Output produced by SAS program: Subject Disposition, by Cycle and Visit - ITT Population

```
*** Word Header Setup ***;
*** Open Header *** ;
PUT '[ViewHeader]';

*** Put study number at top-left of page *** ;
PUT '[EditSelectAll]' ;
PUT '[EditClear]' ;

%IF %STR(&_pgeanchor3) NE %STR() %THEN PUT '[Insert "' &_pgeanchor3" '"] [InsertPara]' ;;
%IF %STR(&_study) NE %STR() %THEN PUT '[Insert "Study: ' &_study" '"] [InsertPara]' ;;
%IF %QUOTE(&_progrname) NE %QUOTE() %THEN PUT '[Insert "SAS Program name: ' &STR(&_progrname)" '"] [InsertPara]' ;;
%IF %STR(&_prgdate) NE %STR() %THEN PUT '[Insert "SAS Program Date / Time (last updated): ' &_prgdate" ' ' &_prgtime" '"]
[InsertPara]' ;;
%IF "&_progout" NE "" %THEN PUT '[Insert "Name of Output produced by SAS program: ' &_progout" ' '"] [InsertPara]' ;;
PUT '[StartOfDocument] [SelectCurSentence] [RightPara] [EndOfDocument]' ;
PUT '[BorderTop]' ;
PUT '[EditSelectAll]' ;
PUT '[FormatFont .Font = "' &_WordHeadFont" '", .Points = "' &_WordHeadFontSize" '"]' ;

*** Close Header ***;
PUT '[closeViewHeaderFooter]';
```

# Generation of SAS program RTF file



- Code loops round directory listing:
- Inserts bookmarks, information in footer, creates MS Word Field Codes \* using DDE.

Program Continued/  
Program Page: 1 of 4 for program: t14-1-5-2.sas

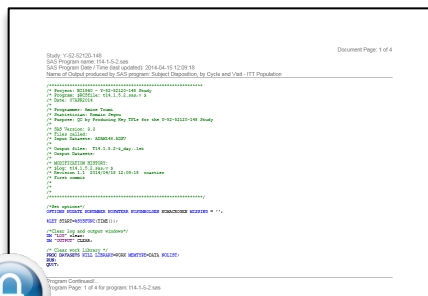
```
PUT '[ViewFooter]';
PUT '[EditSelectAll]' ;
PUT '[EditClear]' ;
PUT '[BorderTop]' ;

%IF %STR(&_pgeanchor1) NE %STR() %THEN PUT '[Insert "' &_pgeanchor1 '"] [InsertPara]' ;;
%IF %STR(&_pgeanchor2) NE %STR() %THEN PUT '[Insert "' &_pgeanchor2 '"] [InsertPara]' ;;
PUT '[EditSelectAll]' ;
PUT '[FormatFont .Font = "' &_WordHeadFont '", .Points = "' &_WordHeadFontSize '"]' ;
PUT '[CloseViewHeaderFooter]';

*** Add book mark *** ;
PUT '[EditBookmark.Name="' &_progrnameshrt '"]', .Add]';
```

\* As RTF files are flat, text files and the MS Field Codes required were complicated with conditional logic, dummy strings were entered `&_pgeanchor1` and `&_pgeanchor2`. These could then be read and replaced.

# Generation of SAS program RTF file



- Code loops round directory listing:
- Read in RTF file that has just been created and replace dummy strings with required MS Field Code text.\*

Program Continued/...  
Program Page: 1 of 8 for program: t14-1-1-2.sas

- Program End -  
Program Page: 8 of 8 for program: t14-1-1-2.sas

{IF={PAGE}-{PAGEREF T14\_1\_1\_2}-1}<{SECTIONPAGES}"Program Continued/..."- Program End -"}  
Program Page: {={PAGE}-{PAGEREF t14\_1\_1\_2}-1} of {SECTIONPAGES} for program: t14-1-1-2.sas

\*\*\* continued / end of program field codes in word \*\*\* ;

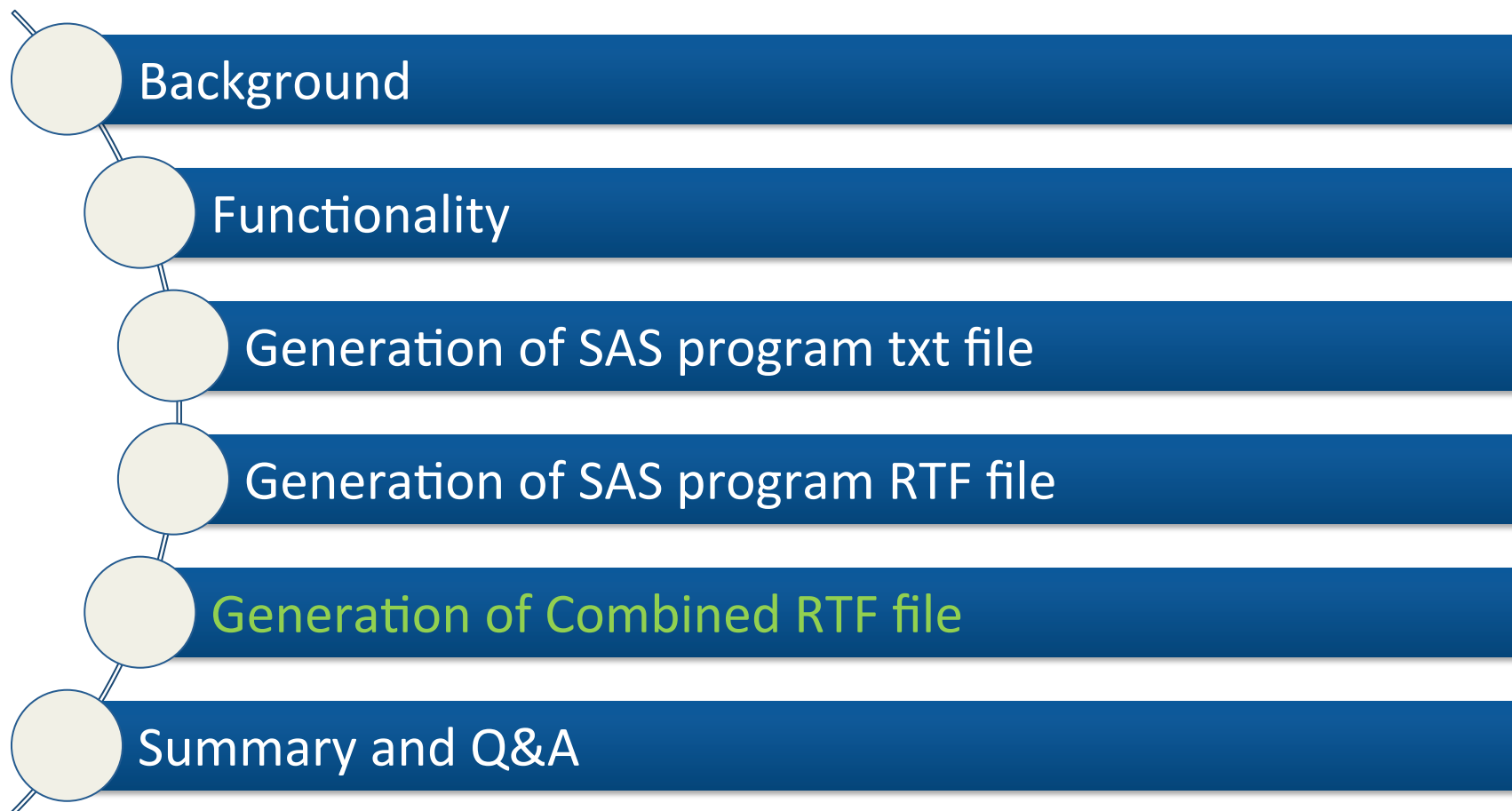
```
datavar1 = TRANWRD(TRANWRD(datavar1,
 "&_pgeanchor1",
 "{\field{*\fldinst{IF}{\field{*\fldinst{={\field{*\fldinst{PAGE}}}{-}{\field{*\fldinst{={\field{*\fldinst{PAGEREF %UPCASE(%STR(&_prognameshrt))}}{-1}}}}}<{\field{*\fldinst {SECTIONPAGES}}}{ 'Program Continued/...' '- Program End -'}}}"),",",'"') ;
```

\*\*\* page X of Y for section field codes in word \*\*\* ;

```
datavar1 = TRANWRD(TRANWRD(datavar1,
 "&_pgeanchor2",
 "{Program Page: }{\field{*\fldinst{={\field{*\fldinst{PAGE}}}{-}{\field{*\fldinst{={\field{*\fldinst{PAGEREF %LOWCASE(%STR(&_prognameshrt))}}{-1}}}}}} of }{\field{*\fldinst {SECTIONPAGES}}}{ for program: %LOWCASE(%STR(&_prognameshrt))}"),",",'"') ;
```

\*MS Word does not have a Field Code for Page 1 within a new Section of a document

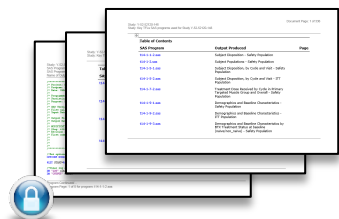
# Agenda



# Generation of Combined RTF file



- Creation of ToC using PROC REPORT with a custom style, document properties, and variables defined in input SAS dataset with MS Word Field codes to support hyperlinking within document.



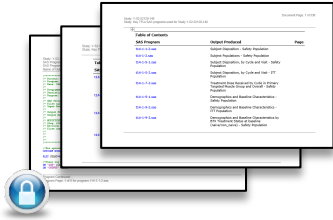
```
DATA work.sasprg ;
 SET work.sasprg ;
:
:
*** ToC hyperlinks: local hyperlinks within the same document as
ToC *** ;
ltoc1 = '{\field {*\fldinst {HYPERLINK \l1 " '
 || TRIM(LEFT(_pronameshrt)) || '"}}'
 || '{\fldrslt {\cf2 ' || TRIM(LEFT(_proname)) || '"}}}'
;
:
:
pageno1 = '{\field {*\fldinst PAGeref '
 || TRIM(LEFT(_pronameshrt))
 || '\h}}'
;
RUN ;
```



```
*** create a ToC *** ;
ODS NORESULTS ;
ODS RTF FILE = "&_outdirRTF.&_combinedRTF"
 STYLE = custom
 ANCHOR = "ToC"
 AUTHOR = "BDLS"
 TITLE = "&_doctitle"
;
ODS NOPROCTITLE ;
ODS PROCLABEL " " ;

PROC REPORT DATA = work.sasprg NOWD
 STYLE = {protectspecialchars=off rules=groups frame=hside} ;
 COLUMN ltoc1 title3 pageno1 ;
 DEFINE ltoc1 / display 'SAS Program' LEFT
 STYLE = {cellwidth=3.5in just=left} ;
 DEFINE title3 / display 'Output Produced' LEFT
 STYLE = {cellwidth=3.5in just=left} ;
 DEFINE pageno1 / display 'Page'
 STYLE = {cellwidth=1in just=right} ;
RUN ;
ODS RTF CLOSE ;
```

# Generation of Combined RTF file



- Loops round and appends individual RTF files together
- At the end of each Inserted RTF files, adds a hyperlink to support navigation to Table of Contents.
- Saves Document as Combined RTF file

```
:
:
*** append RTF documents *** ;
PUT '[EndOfDocument]' ;
PUT '[InsertBreak .Type = 2]' ;
PUT '[InsertFile .Name = "' &STR(&_outdirRTF.&_progrnameRTF.)" '"]' ;

*** Add go to link to top of document *** ;
PUT '[EndOfDocument]' ;
PUT '[FormatFont .Font = "' &_WordHeadFont" '", .Points = "' &_WordHeadFontSize" '"]' ;
PUT '[Italic]' ;
PUT '[Insert "' &_hypanchor1" '"]' ;
RUN ;
:
:
*** save document as RTF *** ;
PUT "[FileSaveAs " " .Name=" "' &_outdirRTF.&_combinedRTF" "'
 ", " " .Format=6]" ;
PUT '[FileSave]' ;
PUT '[FileClose 2]' ;
```

# Summary



- **Pros**

- Allows generation of TXT files and RTF files using directory listings and existing study information
- RTF files are rich in file metadata
- Relatively straightforward to convert RTF files into PDF files.
- Been used across several projects with minimal changes required.
- DDE, although an old technology, it still works.

- **Cons**

- Microsoft does not actively support DDE.
- DDE documentation is limited, and hard to find (Microsoft has removed the DDE manual from their website, but DDE commands are reference in Microsoft Developer Website: [Visual Basic Equivalents I](#))

Thank you





**Cyrille Cartier** | Senior Statistical Programmer | +32 2 774 11 00 | [cyrille.cartier@businessdecision.com](mailto:cyrille.cartier@businessdecision.com)

**Business & Decision Life Sciences**  
Sint-Lambertusstraat 141 rue Saint-Lambert  
B-1200 Brussels  
T: +32 2 774 11 00  
F: +32 2 774 11 99  
[lifesciences@businessdecision.com](mailto:lifesciences@businessdecision.com)  
<http://www.businessdecision-lifesciences.com/>