PharmaSUG 2021 - Paper DS-130

Untangling the Knot - Implementing Analysis Results Standards (Using SAS®)

Hansjörg Frenzel, PRA Health Sciences

ABSTRACT

In his 2019 PhUSE paper[1] Chris Decker suggested moving from traditional TFL programming to generating analysis results data. Analysis results standards, which define how results data are stored as well as the associated metadata, support the separation of results generation from results presentation. Some of the benefits include avoiding to create results multiple times for multiple outputs, better documentation, traceability to the underlying data, reusability, and potential for automation when creating outputs for multiple stakeholders. This paper attempts to present the metadata for a possible analysis results standard and to show how they could be used for the automated creation of tables.

INTRODUCTION

Analysis Results Standards define how results and associated metadata can be stored. They support the separation of results generation from results presentation. Chris Decker suggested in his 2019 PhUSE paper making this separation and moving from traditional TFL programming to generating analysis results data. But why? He provides some excellent reasons:

- Imbalance of analysis vs. results presentation: Most of the effort that goes into CSR outputs is formatting, not results creation. Because every study is different, it is difficult to use validated programs to generate result displays.
- Reusability of results for scenarios other than CSR generation: Stakeholders other than Biostatisticians and Reviewers, who are interested in the CSR results, must parse the TFL section manually for results they need, instead of accessing results directly and formatting them for their own needs.
- Lack of machine readability: One reason why there is so little automation in TFL generation is that results are not machine readable. Lack of machine readability means that a human reviewer is required in order to understand what a specific number presented in an output means, and thus no automated processes can access results . It also prevents searching for results across multiple trials.
- Transparency: Stakeholders must trust that the producer of the SAP, CSR, and outputs has generated the results as described, and that those results were verified following an appropriate QC process.

When separating results generation from their presentation, an increase in efficiency can be expected. One of the many many reasons is that rather bulky output programs can be split up into modular units for each of the two tasks[2]. Modularization may not be beneficial when every TFL program is a unique piece of code, only generated for one task. However, with the implementation of a standard analysis result structure, complexity is reduced for both steps:

- When creating results, the way results are displayed is unimportant.
- When producing the display, how the results were generated is unimportant.

Because suddenly results generating sub-routines and results displaying sub-routines get more generic, it might prove beneficial to split them further up into modules. These modules in turn will be simpler, thus

[1] Chris Decker: Analysis Results that Saves Trees - #killTFLs, Paper DH12, PhUSE US Connect 2019

[2] In software development this process is known as code refactoring, which refers to the possibility to re-structuring code used in software without changing its functionality, leading to simpler and better readable code, which in turn is easier to maintain and to extend. (Wikipedia: Code Refactoring)

easier to validate, and there is a good chance that result displays can be produced with less effort for both initial programming and quality control.

On the other hand, the current focus on formatting generates most of the effort of output programming and prevents the results from being machine readable. This poses a risk to quality because of the discrepancy between the huge effort of QC and the time and budget restrictions for delivering the output. In addition, the variety of output designs makes it impossible to develop tools to automate this step.

How can we overcome this conundrum? Over the last twenty years standardization of data formats has brought tremendous benefits to the collection and exchange of data. CDISC standards for data collection, tabulation, and analysis have greatly improved transparency and efficiency of the drug approval process, allowing for an increasing amount of automation in the collection and downstream extraction, transformation, and loading of data. TFL production is simply next in line and well ripe for standardization. But instead of standardizing outputs, having a solid standard for results data and metadata can overcome most of the limitations listed above. This paper presents an early proof of concept for generating and using analysis results standards. Because of the limitations of the author, SAS® is used for these first steps. It needs to be stated up-front that it is not a requirement to use SAS® for generating and using of analysis results data and metadata, but it might prove beneficial for the portion of the audience that is heavily SAS® focused.

To prove his point, the author uses his own draft of analysis results metadata and data, as so far, no standard has been developed. The results presented in the example tables stem from manufactured data which were created without any regard for internal consistency. Thus, the individual results might not appear to be meaningful. This does not mean that the intended process or the tools did not work.

SEPARATING CONTENTS FROM PRESENTATION

The concept behind analysis results standards is the idea of separating contents from presentation. Although achieving this is not trivial, it is backed up by a software development principle that has proven incredibly successful for the development of the Internet. As an example, in the clinical submission process the principle is used in storing metadata for data collection and analysis data in XML format. Define.xml files are text files and, although XML is generally considered human readable, are on their own not easy for a human to process. However, when XML is combined with a style sheet it provides well-structured information.

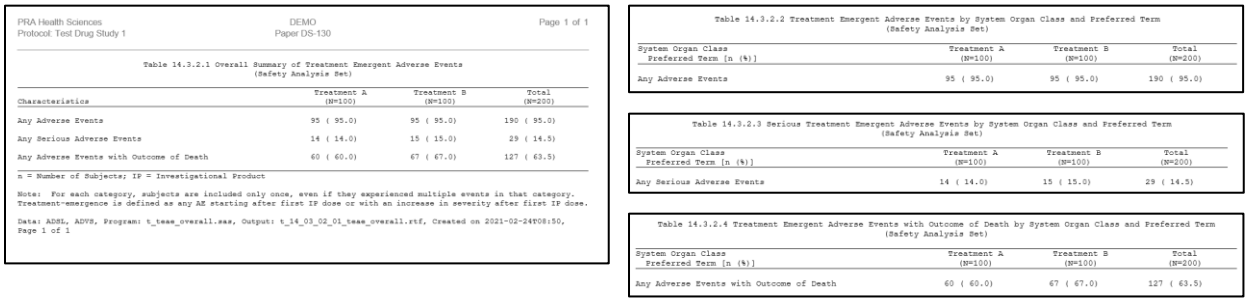
When separating analysis results data from their presentation the following is possible:

- Simplification of data analyses by use of standardized, validated tools.
- Support of analysis results with metadata to increase transparency.
- Using standardized and validated tools for generating specific forms of output, which leads to an increase in reliability and reduction of costs.

The focus is on broadening the use of the results of the analyses for a clinical study, while increasing efficiency, transparency, and reliability. Separating contents from presentation also allows for reusing results data, where required, and abstain from creating the same result multiple times for different tables or figures.

A quite common example can be found with adverse events reporting:

Figure 1: Overall number of subjects experiencing AEs



PRA Health Sciences DEMO Page 1 of 1
Protocol: Test Drug Study 1 Paper DS-130

Table 14.3.2.1 Overall Summary of Treatment Emergent Adverse Events
(Safety Analysis Set)

Characteristics	Treatment A (N=100)	Treatment B (N=100)	Total (N=200)
Any Adverse Events	95 (95.0)	95 (95.0)	190 (95.0)
Any Serious Adverse Events	14 (14.0)	15 (15.0)	29 (14.5)
Any Adverse Events with Outcome of Death	60 (60.0)	67 (67.0)	127 (63.5)

n = Number of Subjects; IP = Investigational Product

Note: For each category, subjects are included only once, even if they experienced multiple events in that category. Treatment-emergence is defined as any AE starting after first IP dose or with an increase in severity after first IP dose.

Data: ADSL, ADVS, Program: t_teae_overall.sas, Output: t_14_03_02_01_teae_overall.rtf, Created on 2021-02-24T08:50, Page 1 of 1

Table 14.3.2.2 Treatment Emergent Adverse Events by System Organ Class and Preferred Term
(Safety Analysis Set)

System Organ Class Preferred Term [n (%)]	Treatment A (N=100)	Treatment B (N=100)	Total (N=200)
Any Adverse Events	95 (95.0)	95 (95.0)	190 (95.0)

Table 14.3.2.3 Serious Treatment Emergent Adverse Events by System Organ Class and Preferred Term
(Safety Analysis Set)

System Organ Class Preferred Term [n (%)]	Treatment A (N=100)	Treatment B (N=100)	Total (N=200)
Any Serious Adverse Events	14 (14.0)	15 (15.0)	29 (14.5)

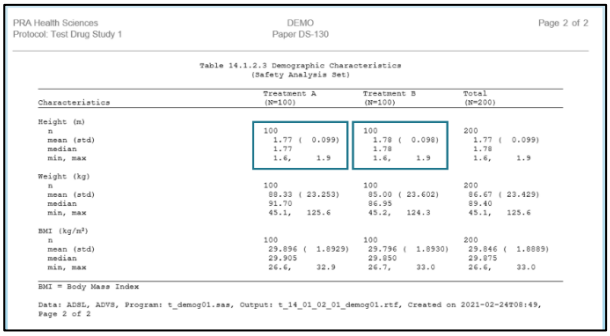
Table 14.3.2.4 Treatment Emergent Adverse Events with Outcome of Death by System Organ Class and Preferred Term
(Safety Analysis Set)

System Organ Class Preferred Term [n (%)]	Treatment A (N=100)	Treatment B (N=100)	Total (N=200)
Any Adverse Events with Outcome of Death	60 (60.0)	67 (67.0)	127 (63.5)

The first line in AE tables counting the overall number of subjects with adverse events is often repeated in an additional table presenting an overview over the different analysis subgroups.

The example below shows how vital signs results are presented not only in the vital signs table, but also in the demography table. Traditionally, this requires producing identical results in two programs, which requires to QC them and in addition to check that they are reported consistently.

Figure 2: Demographics and Vital Signs Results



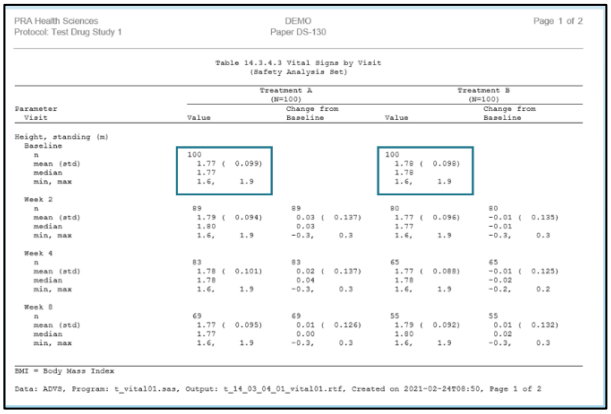
PRA Health Sciences DEMO Page 2 of 2
Protocol: Test Drug Study 1 Paper DS-130

Table 14.1.2.3 Demographic Characteristics
(Safety Analysis Set)

Characteristics	Treatment A (N=100)	Treatment B (N=100)	Total (N=200)
Height (m)			
n	100	100	200
mean (std)	1.77 (0.099)	1.78 (0.098)	1.77 (0.099)
median	1.77	1.78	1.78
min, max	1.6, 1.9	1.6, 1.9	1.6, 1.9
Weight (kg)			
n	100	100	200
mean (std)	88.33 (23.253)	85.00 (23.602)	86.67 (23.429)
median	91.70	86.95	89.40
min, max	45.1, 125.6	45.2, 124.3	45.1, 125.6
BMI (kg/m²)			
n	100	100	200
mean (std)	29.896 (1.8929)	29.796 (1.8930)	29.846 (1.8889)
median	29.905	29.850	29.875
min, max	26.6, 32.9	26.7, 33.0	26.6, 33.0

BMI = Body Mass Index

Data: ADSL, ADVS, Program: t_demog01.sas, Output: t_14_01_02_01_demog01.rtf, Created on 2021-02-24T08:49, Page 2 of 2



PRA Health Sciences DEMO Page 1 of 2
Protocol: Test Drug Study 1 Paper DS-130

Table 14.3.4.3 Vital Signs by Visit
(Safety Analysis Set)

Parameter / Visit	Treatment A (N=100) Value	Treatment A (N=100) Change from Baseline	Treatment B (N=100) Value	Treatment B (N=100) Change from Baseline
Height, standing (m)				
Baseline				
n	100		100	
mean (std)	1.77 (0.099)		1.78 (0.098)	
median	1.77		1.78	
min, max	1.6, 1.9		1.6, 1.9	
Week 2				
n	89	89	80	80
mean (std)	1.79 (0.094)	0.03 (0.137)	1.77 (0.096)	-0.01 (0.135)
median	1.80	0.03	1.77	-0.01
min, max	1.6, 1.9	-0.3, 0.3	1.6, 1.9	-0.3, 0.3
Week 4				
n	83	83	65	65
mean (std)	1.78 (0.101)	0.02 (0.137)	1.77 (0.088)	-0.01 (0.125)
median	1.78	0.04	1.78	-0.02
min, max	1.6, 1.9	-0.3, 0.3	1.6, 1.9	-0.2, 0.2
Week 8				
n	69	69	55	55
mean (std)	1.77 (0.095)	0.01 (0.126)	1.79 (0.092)	0.01 (0.132)
median	1.77	0.00	1.80	0.02
min, max	1.6, 1.9	-0.3, 0.3	1.6, 1.9	-0.3, 0.3

BMI = Body Mass Index

Data: ADVS, Program: t_vital01.sas, Output: t_14_03_04_01_vital01.rtf, Created on 2021-02-24T08:50, Page 1 of 2

ANALYSIS RESULTS STANDARDS

The work on CDISC Analysis Results Standards has just started its work and CDISC formed a team mid of 2020. Because no tangible results have been developed yet, for the purpose of this paper all data or metadata definitions presented in this paper are my own development and can deviate from the final standard substantially. For purpose of presentation metadata do also not strictly adhere to the analysis results metadata standard.

Analysis results standards provide information on how analysis results data and their metadata - analysis metadata and output metadata - are structured.

Analysis Results Data contain results together with the necessary key variables and qualifiers to uniquely identify each result stored in the result dataset and link them to metadata.

Analysis Results Metadata describe analysis results and displays:

- Analysis Metadata: Describe data source, data item, selection, and method as well as which additional variables were used for grouping results.
- Display Metadata: Describe the display: Purpose, analyses included, and other associated metadata.

For the analysis results data and metadata structures I purposely used names different from standard

CDISC variable names to avoid confusion.

ANALYSIS RESULTS DATA

Analysis results data contain the contents of tables and some figures in a standard format. Such a format is suggested below. The key concept is the analysis, a collection of results produced by a statistical procedure or sub-routine. A table could consist of multiple analyses, like a demography table consists of results for different variables (age, sex, race, ethnicity, and a variety of baseline characteristics).

Table 1: Analysis Results Data Structure

Variable Name	Label	Variable Type	Description
SPONSOR	Sponsor	Character	Name of the Sponsor
PROTOCOL	Protocol	Character	Title of the Protocol
ANAID	Analysis	Character	Analysis Identifier
TRTVAL	Treatment Value	Character	Treatment Value that represents the treatment arm or a collection of treatment arms used to summarize data
ACATzz	Analysis Category zz Value	Character	Value of Analysis Category zz (01-99). The name of the associated ADaM variable can be found in Analysis Results Metadata, variable ACATzzNM
STATCODE	Statistics Code	Character	Code to identify the statistic that describes the results.
RESULT	Result	Value	Value of the statistic identified in field STATCODE

A macro or sub-routine to generate analysis results should as a minimum consist of:

- One statistical procedure or sub-routine to generate the results.
- Code to transpose the results to the analysis results data structure.
- The metadata required to describe the results.

The tool should make all statistics calculated by the procedure or sub-routine available as analysis results. Especially for more complex analyses this allows the recipient of the data to independently vet whether the results can be used or not.

ANALYSIS METADATA

Analysis metadata describe the different analyses sufficiently for a stakeholder to enable the independent verification of the results, thus increasing transparency.

Table 2 Analysis Metadata Structure

Variable Name	Label	Variable Type	Description
SPONSOR	Sponsor	Character	Sponsor name
PROTOCOL	Protocol	Character	Protocol title
DATA	Dataset	Character	ADaM dataset name
ANAID	Analysis	Character	Analysis ID
POPUL	Population	Character	Population name
WCLAUSE	Filter	Character	Analysis where clause
AVAR	Variable	Character	Analysis variable name (provide if required)
POPVAR	Population Variable	Character	Population variable name
TRTVAR	Treatment Variable	Character	Treatment variable name
TRTGPzz	Treatment Group zz	Character	Treatment group zz (01-99).Treatment group code(s) as stored in treatment variable. Comma-separate in case more than one treatment group is used for analyis, e.g. for a sub-total or a total ("'TRT A'", "'TRT B'", "'TRT A', 'TRT B'").
TRTVALzz	Treatment Value zz	Character	Treatment value zz (01-99).Treatment value code used in the analysis (e.g. "TRT A", "TRT B", "TOTAL")

Variable Name	Label	Variable Type	Description
ACATzzNM	Analysis Category zz Name	Character	Name of analysis by-group or class variable zz.
METHOD	Analysis method	Character	Analysis Method used to generate results (e.g. PROC SUMMARY, MIXED MODEL, LINEAR REGRESSION ANALYSIS, T-TEST)
DESCR	Description	Character	Description of the analysis
OUTNAME	Output Dataset Name	Character	Name of the dataset the analysis results are stored in.

Most of the information in the analysis metadata would conveniently be provided by the sub-routine or the macro generating the analysis results. The information could be extended with the quality control status, associated dates, and the programmers and qc programmers names.

DISPLAY METADATA

Display metadata allow stakeholders as a minimum to find a specific display and to identify the analyses presented therein.

Table 3 Display Metadata Structure

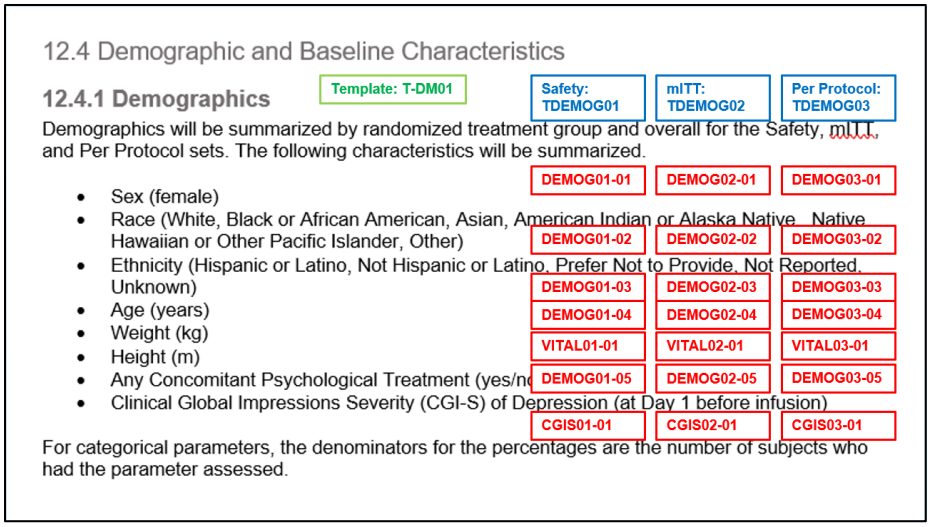
Variable Name	Label	Variable Type	Description
SPONSOR	Sponsor	Character	Sponsor name
PROTOCOL	Protocol	Character	Protocol title
DISPLAID	Display ID	Character	Display ID
ANAzz	Analysis zz	Character	Analysis ID(s) presented in the display, in the order of appearance
PROGNAME	Program Name	Character	Name of the program which generated the display
OUTFNAME	Output File Name	Character	Name of the display file
TITLEzz	Title zz	Character	Title zz (01-99)
FNOTEzz	Footnote zz	Character	Footnote zz (01-99)
DESCR	Description	Character	Description of the display

ANNOTATING THE SAP WITH ANALYSIS AND DISPLAY INFORMATION

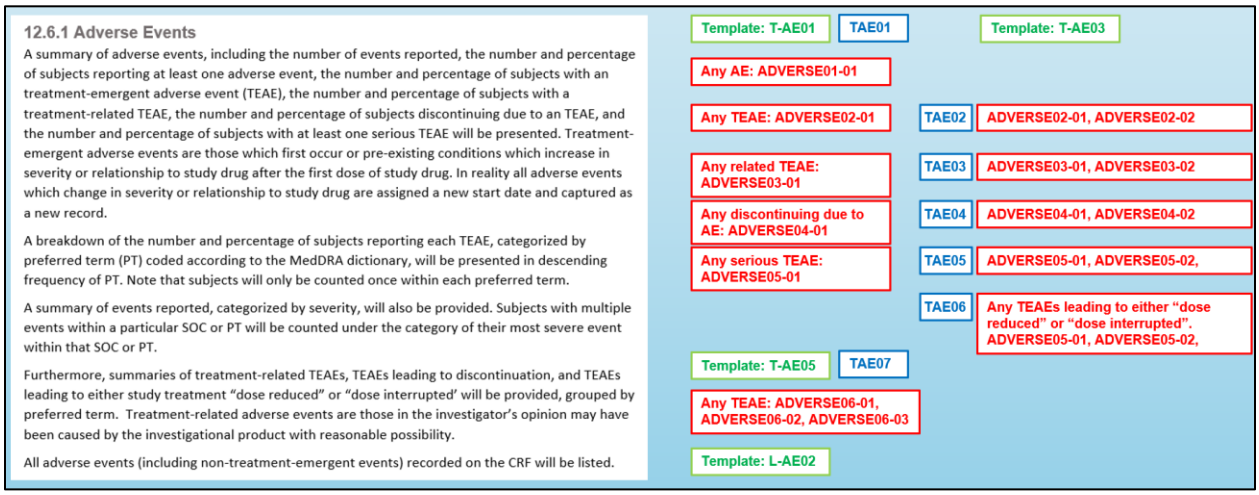
An interesting insight while writing this paper was that with analysis result data and metadata it might be possible to shift the annotations from TFL shells to the Statistical Analysis Plan (SAP), because it already describes analyses in sufficient detail. TFL shells can be viewed as more detailed instructions for programmers (and as illustrations for sponsor statisticians and medical writers) on how to create specific displays. But most of the information codified in shells is already present in the SAP. Below some examples:

> 12.0 **Statistical Methods**
>
> Unless otherwise indicated, continuous variables will be summarized with the following descriptive statistics: n (number of observations), (arithmetic) mean, standard deviation (SD), first and third quartiles (Q1 and Q3), minimum (min) value, median, and maximum (max) value. The median, minimum and maximum values will be displayed to the same level of precision as the raw data, the mean to a further decimal place and the SD to two additional decimal places – up to 4 decimal places total.
>
> Categorical data will be summarized with frequencies and percentages. Percentages by categories will be based on the number of subjects exposed within a treatment. Percentages will be rounded to one decimal place, except 100% will be displayed without any decimal places and percentages will not be displayed for zero counts.
>
> All analyses will use SAS version 9.4 or higher. No inferential analyses (ie, hypothesis testing, confidence intervals) will be performed for any variable. For each summary table, corresponding supportive listing(s) will be provided.

The statistical methods section of the SAP describes in detail which descriptive statistics for both continuous and categorical data are to be presented in descriptive tables.



Similarly, the section describing the demographics table include sufficient detail about which data shall be presented in the table, and in which categories and in which order they shall be presented.



Finally, the section about adverse events describes the number of tables, their contents, and which data must be presented in which table.

The only missing part is a visual instruction on how to organize the output. Many sponsor organizations and some CROs developed output standards, which could be utilized in this case. The annotations in the green boxes point a developer to templates that are to be used as guidance.

Here lie great opportunities for automation: By selecting templates instead of laboriously generating TFL shells in a word processing program, the statistician can already provide metadata for the generation of TFL shells and result displays. Templates can be associated with programs for creating both TFL shells and statistical outputs. By adding study specific titles and footnotes, the statistician only adds to the metadata required for creating a shell or a display while writing the SAP. This relieves programmers from manually entering the same information in a manually written program.

UTILIZING RESULT DATA AND METADATA FOR TFL PRODUCTION

As discussed in the previous section, the first step would be to parse the SAP for displays and analyses, annotating them in the SAP's text. If an application for generating TFLs is available, these would be configured at this point to generate TFL Shells. The configurations provide the metadata that software would pick up and translate into analysis and display metadata, as well as picking the right sub-routine to prepare the analysis results. When using SAS®, macros would be an optimal candidate to fulfill this purpose.

```
data ana_meta01 ;
    length item $ 40 value $ 1000 ;

    %_items( Analysis, DEMOG01-01 ) ;
    %_items( Data, ADSL ) ;
    %_items( Description, %str(Counts and percentage for sex.) ) ;
    %_items( Population, Safety Set) ;
    %_items( Population Variable, SAFFL ) ;
    %_items( Where Clause, saffl eq 'Y') ;
    %_items( Treatment Variable, TRT01A) ;
    %_items( Analysis Bygroup, SEXN SEX ) ;
    %_items( Analysis Statistics , n pct ) ;
    %_items( Analysis Type , CATEGORICAL ) ;
    %_items( Treatment Group 01, 'TRT A' ) ;
    %_items( Treatment Group 02, 'TRT B' ) ;
    %_items( Treatment Group 03, 'TRT A' 'TRT B' ) ;
    %_items( Treatment Value 01, 'TRT A' ) ;
    %_items( Treatment Value 02, 'TRT B' ) ;
    %_items( Treatment Value 03, 'TOTAL' ) ;
run ;
```

For this paper, a simple SAS® dataset serves as crude user interface. Macro %_items simply assigns the contents of the two positional parameters to two dataset variables (*item* and *value*) and outputs a new record, which results in a SAS dataset with two variables containing item and value pairs.

```
/* generate metadata */

%_genMeta(sponsor=PRA,protocol=Testdrug Study 01, confds=ana_meta01, outlib=anares, outds=adsl_ana,debug=N) ;
```

Macro %_genMeta uses dataset *ana_meta01* as input to generate the analysis metadata, and creates the first row of the analysis metadata dataset ADSL_ANA:

Table 4 Analysis Metadata

SPONSOR	PROTOCOL	ANAID	DATA	POPUL	WCLAUSE	AVAR	POPVAR	TRTVAR	TRTGP01	TRTGP02	TRTGP03	TRTVAL01	TRTVAL02	TRTVAL03
PRA	Testdrug Study 01	DEMOG01-01	ADSL	Safety Set	saffl eq 'Y'		SAFFL	TRT01A	'TRT A'	'TRT B'	'TRT A' 'TRT B'	'TRT A'	'TRT B'	'TOTAL'
PRA	Testdrug Study 01	DEMOG01-02	ADSL	Safety Set	saffl eq 'Y'		SAFFL	TRT01A	'TRT A'	'TRT B'	'TRT A' 'TRT B'	'TRT A'	'TRT B'	'TOTAL'
PRA	Testdrug Study 01	DEMOG01-03	ADSL	Safety Set	saffl eq 'Y'		SAFFL	TRT01A	'TRT A'	'TRT B'	'TRT A' 'TRT B'	'TRT A'	'TRT B'	'TOTAL'
PRA	Testdrug Study 01	DEMOG01-04	ADSL	Safety Set	saffl eq 'Y'	AGE	SAFFL	TRT01A	'TRT A'	'TRT B'	'TRT A' 'TRT B'	'TRT A'	'TRT B'	'TOTAL'

SPONSOR	PROTOCOL	ANAID	ACAT01NM	ACAT02NM	METHOD	DESCR	OUTDATA
PRA	Testdrug Study 01	DEMOG01-01	SEXN	SEX	SQL COUNT	Counts and percentage for sex.	ADSL_ADR
PRA	Testdrug Study 01	DEMOG01-02	RACEN	RACE	SQL COUNT	Counts and percentage for race.	ADSL_ADR
PRA	Testdrug Study 01	DEMOG01-03	ETHNICN	ETHNIC	SQL COUNT	Counts and percentage for ethnicity.	ADSL_ADR
PRA	Testdrug Study 01	DEMOG01-04			PROC SUMMARY	Summary statistics n, mean, std dev, median, min, max of age data by treatment group	ADSL_ADR

The same configurations are used to generate the results:

```
/*generate result data*/

%_genSumStats(sponsor=PRA, protocol=Testdrug Study 1, confds=ana_meta01,lib=analysis,outlib=anares,outds=adsl_adr) ;
```

%_genSumStats is chosen to generate summary statistics for continuous and categorical data (for variable *age* in this case). Although in this example the macro was created to generate both statistics for continuous and categorial data, in a real application there might be two separate macros. In any way, in an application these macros would be called because the configuration for "Analysis Type" was "CONTINUOUS" or "CATEGORICAL". For other analysis types different macros would be called instead.

Table 5 Analysis Result Data for DEMOG01-01

Sponsor	Protocol	ANAID	TRTVAL	ACAT01	ACAT02	STATCODE	RESULT
PRA	Testdrug Study 1	DEMOG01-01	TOTAL	1	F	COUNT	101
PRA	Testdrug Study 1	DEMOG01-01	TOTAL	1	F	PERCENTAGE	50.5
PRA	Testdrug Study 1	DEMOG01-01	TOTAL	1	F	DENOMINATOR	200
PRA	Testdrug Study 1	DEMOG01-01	TOTAL	2	M	COUNT	99
PRA	Testdrug Study 1	DEMOG01-01	TOTAL	2	M	PERCENTAGE	49.5
PRA	Testdrug Study 1	DEMOG01-01	TOTAL	2	M	DENOMINATOR	200
PRA	Testdrug Study 1	DEMOG01-01	TRT A	1	F	COUNT	45
PRA	Testdrug Study 1	DEMOG01-01	TRT A	1	F	PERCENTAGE	45
PRA	Testdrug Study 1	DEMOG01-01	TRT A	1	F	DENOMINATOR	100
PRA	Testdrug Study 1	DEMOG01-01	TRT A	2	M	COUNT	55
PRA	Testdrug Study 1	DEMOG01-01	TRT A	2	M	PERCENTAGE	55
PRA	Testdrug Study 1	DEMOG01-01	TRT A	2	M	DENOMINATOR	100
PRA	Testdrug Study 1	DEMOG01-01	TRT B	1	F	COUNT	56
PRA	Testdrug Study 1	DEMOG01-01	TRT B	1	F	PERCENTAGE	56
PRA	Testdrug Study 1	DEMOG01-01	TRT B	1	F	DENOMINATOR	100
PRA	Testdrug Study 1	DEMOG01-01	TRT B	2	M	COUNT	44
PRA	Testdrug Study 1	DEMOG01-01	TRT B	2	M	PERCENTAGE	44
PRA	Testdrug Study 1	DEMOG01-01	TRT B	2	M	DENOMINATOR	100

For analysis DEMOG01-01, metadata specify that variables SEXN and SEX from dataset ADSL were utilized to subset the data for counts in each combination of these variables. The treatment value "TOTAL" wass created from data within both treatment groups "TRT A" and "TRT B" stored in variable TRT01A. The denominator stems from the number of subjects in the safety population and a specific (combination of) treatment group(s).

The analysis results data for the TOTAL treatment group of the continuous data of analysis DEMOG01-04 is presented below:

Table 6 Analysis Results Data for DEMOG01-04

Sponsor	Protocol	ANAID	TRTVAL	ACAT01	ACAT02	STATCODE	RESULT
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			COUNT	200
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			MEAN	45.72
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			STDDEV	18.63081146
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			MEDIAN	43
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			1ST QUARTILE	29
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			3RD QUARTILE	62
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			MINIMUM	18
PRA	Testdrug Study 1	DEMOG01-04	TOTAL			MAXIMUM	77
PRA	Testdrug Study 1	DEMOG01-04	TRT A			COUNT	100

Here, the associated metadata provide the user with the information that variable AGE was summarized.

Similar to analysis results, displays are to be configured by parsing the SAP for suitable information. Again, a crude SAS® dataset is used in this paper to present the configuration interface:

```
*** report the table *** ;
data rpt_demog01 ;
    length item $ 40 value $ 1000 ;

    %_items( Output ID, TDEMOG01 ) ;
    %_items( Analyses, DEMOG01-01 DEMOG01-02 DEMOG01-03 DEMOG01-04 VITAL01-01 DEMOG01-05 CGIS01-01) ;
    %_items( Program File , t_demog01.sas) ;
    %_items( Output File , t_14_01_02_01_demog01.rtf) ;
    %_items( Population , Safety Analysis Set ) ;
    %_items( Title 01, Table 14.1.2.3 Demographic Characteristics ) ;
    %_items( Title 02, (Safety Analysis Set)) ;
    %_items( Footnote 01, %str(BMI = Body Mass Index) ) ;
    %_items( Description , %str(Summary statstics for demographic characteristics Age, Sex, Race,
                          Ethnicity and baseline vital signs parameters weight, height, and BMI. )
           ) ;
run ;
```

And similar to generating analysis results data and metadata, a macro is called to generate display metadata:

```
%_reportOutput(sponsor=PRA,protocol=Testdrug Study 01,confds=rpt_demog01,outlib=anares,outds=displays) ;
```

Macro %_reportOutput generates the display metadata for specific outputs, requiring one call for each output.

Table 7 Display Metadata

sponsor	protocol	DISPLAYID	DATA	PROGNAME	OUTFNAME	ANA01	ANA02	ANA03	ANA04	ANA05
PRA	Testdrug Study 01	TDEMOG01	ADSL, ADVS, ADQSCGSI	t_demog_saf.sas	t_14_01_02_01_demog_saf.rtf	DEMOG01-01	DEMOG01-02	DEMOG01-03	DEMOG01-04	VITAL01-01
PRA	Testdrug Study 01	TDEMOG02	ADSL, ADVS, ADQSCGSI	t_demog_mitt.sas	t_14_01_02_01_demog_mitt.rtf	DEMOG02-01	DEMOG02-02	DEMOG02-03	DEMOG02-04	VITAL02-01
PRA	Testdrug Study 01	TDEMOG03	ADSL, ADVS, ADQSCGSI	t_demog_pp.sas	t_14_01_02_01_demog_pp.rtf	DEMOG03-01	DEMOG03-02	DEMOG03-03	DEMOG03-04	VITAL03-01

sponsor	protocol	DISPLAYID	ANA06	ANA07	TITLE01	TITLE02	FNOTE01
PRA	Testdrug Study 01	TDEMOG01	DEMOG01-05	CGSI01-01	Table 14.1.2.3 Demographic Characteristics	(Safety Analysis Set)	BMI = Body Mass Index
PRA	Testdrug Study 01	TDEMOG02	DEMOG02-05	CGSI02-01	Table 14.1.2.4 Demographic Characteristics	(modified Intent-to-Treat Set)	BMI = Body Mass Index
PRA	Testdrug Study 01	TDEMOG03	DEMOG03-05	CGSI03-01	Table 14.1.2.5 Demographic Characteristics	(Per-Protocol Set)	BMI = Body Mass Index

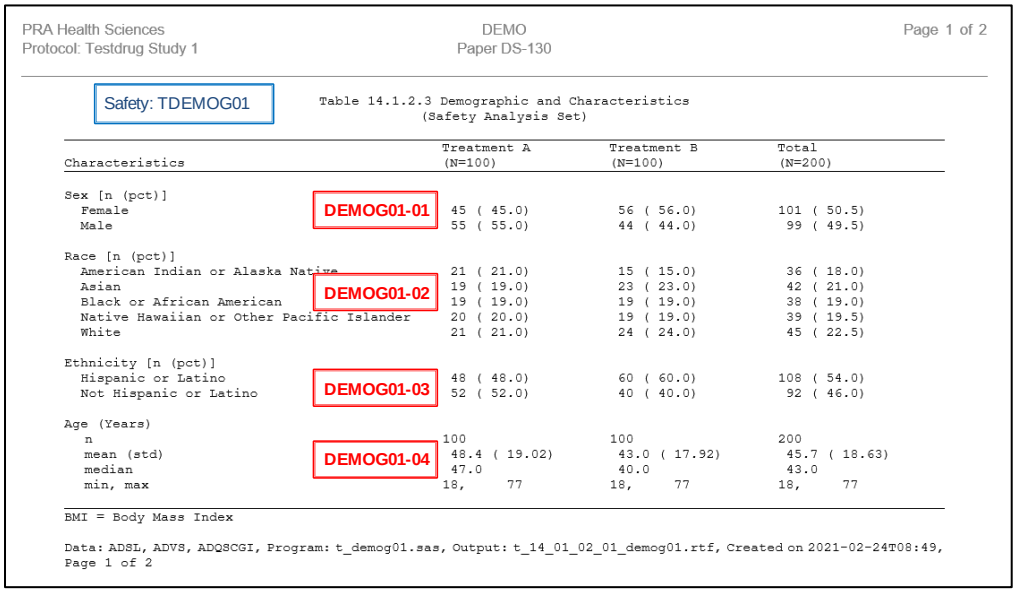
sponsor	protocol	DISPLAYID	FNOTE02	FNOTE03	descr
PRA	Testdrug Study 01	TDEMOG01			Summary statstics for demographic characteristics Age, Sex, Race, Ethnicity and baseline vital signs parameters weight, height, and BMI.
PRA	Testdrug Study 01	TDEMOG02			Summary statstics for demographic characteristics Age, Sex, Race, Ethnicity and baseline vital signs parameters weight, height, and BMI.
PRA	Testdrug Study 01	TDEMOG03			Summary statstics for demographic characteristics Age, Sex, Race, Ethnicity and baseline vital signs parameters weight, height, and BMI.

A call to a macro generating the desired output would be required but was not developed for the purpose of this paper. Such display macros would very likely be organized alongside templates, and would need to have components for the following tasks:

- Loading the analysis data from the analysis results dataset(s).
- For tables:
  - Format results according to the associated template.
  - Transpose the formatted results to the final output structure.
  - Generate the output using PROC REPORT.

- For figures, the required graphical procedure would be called.

The final output generated by the above-mentioned macros and a custom table program can be seen below:



PRA Health Sciences
Protocol: Testdrug Study 1

DEMO
Paper DS-130

Page 1 of 2

Safety: TDEMOG01

Table 14.1.2.3 Demographic and Characteristics
(Safety Analysis Set)

Characteristics		Treatment A (N=100)	Treatment B (N=100)	Total (N=200)
Sex [n (pct)]				
Female	DEMOG01-01	45 (45.0)	56 (56.0)	101 (50.5)
Male		55 (55.0)	44 (44.0)	99 (49.5)
Race [n (pct)]				
American Indian or Alaska Native		21 (21.0)	15 (15.0)	36 (18.0)
Asian	DEMOG01-02	19 (19.0)	23 (23.0)	42 (21.0)
Black or African American		19 (19.0)	19 (19.0)	38 (19.0)
Native Hawaiian or Other Pacific Islander		20 (20.0)	19 (19.0)	39 (19.5)
White		21 (21.0)	24 (24.0)	45 (22.5)
Ethnicity [n (pct)]				
Hispanic or Latino	DEMOG01-03	48 (48.0)	60 (60.0)	108 (54.0)
Not Hispanic or Latino		52 (52.0)	40 (40.0)	92 (46.0)
Age (Years)				
n		100	100	200
mean (std)	DEMOG01-04	48.4 (19.02)	43.0 (17.92)	45.7 (18.63)
median		47.0	40.0	43.0
min, max		18, 77	18, 77	18, 77

BMI = Body Mass Index

Data: ADSL, ADVS, ADQSCGI, Program: t_demog01.sas, Output: t_14_01_02_01_demog01.rtf, Created on 2021-02-24T08:49,
Page 1 of 2

CONCLUSION

Standardizing analysis results data would provide significant benefits to the industry. It would help with transparency and traceability to the results presentation, which is otherwise only available from the SAP and the code of the program that created the result display. It would add machine readability to the presentation of results, allowing for code-based searches of results across multiple studies. By supporting the separation of results generation from results display, one tool can be developed for results generation and another for results display, thus mitigating the imbalance of analysis vs results presentation. Additionally, it supports the reuse of results for scenarios other than CRS generation. Furthermore, it allows for quicker turn-around time of additional analysis requests because the tools are already available, and the project team does not need to adapt or alter already existing programs.

REFERENCES

Decker, Chris. 2019. "Analysis Results that Saves Trees - #killTFLs", Paper DH12, PhUSE US Connect 2019

Code Refactoring. Wikipedia. Available at: https://en.wikipedia.org/wiki/Code_refactoring

CONTACT INFORMATION

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

Hansjörg Frenzel
PRA Health Sciences
frenzelhansjoerg@prahs.com