

Paper 176-2013

Easy Button: A process for generating standardized safety and non-safety related Clinical Trial Reports

Xiangchen (Bob) Cui, Vertex Pharmaceuticals, Cambridge, MA
Mominul Islam, Vertex Pharmaceuticals, Cambridge, MA
Jiannan Hu, Vertex Pharmaceuticals, Cambridge, MA
Yanwei Han, Vertex Pharmaceuticals, Cambridge, MA
Sanjiv Ramalingam, Vertex Pharmaceuticals, Cambridge, MA

ABSTRACT:

Developing a process for standardized reports in terms of Tables, Figures and Listings (TFLs) generations for a clinical study report by developing SAS® macro-based either reporting tool or template SAS programs is an ongoing focus of the health care and life science industry. We have developed SAS macros and template SAS programs (macro calls) based on our Biometrics Standard TFLs shells for safety analysis like Adverse Events (AE), LABs, ECG, Vitals and other non-safety analysis, for example, Baseline Characteristics, Disposition, Concomitant Medication, Drug Exposure, Compliance and Figures for Kaplan-Meier plot. Our newly developed process includes development of reporting macros by utilizing the existing department macros to generate these standard reports. The macros have been developed assuming the CDISC ADaM analysis data set standards which enable us to minimize the number of macro parameters for efficient use of the macros by the user. This process also shortens the development cycle time, and facilitates the adoption from SAS programmers to clinical reporting. During the development of the reporting macros, the macros have been tested using the ongoing clinical trials of different phases. We also developed 'User Manual' and standard 'Template Programs' for easy use for our programmers. The process stipulated in this paper will reduce the report generation time significantly and achieve the quality by design principle.

Keywords: CDISC, TFLs, ADaM, AE, LAB, ECG, VITALS.

INTRODUCTION:

This paper introduces a process which includes several SAS macros and template SAS programs to automate the safety related TFLs generation on the basis of our Biometrics Standard TFLs which was developed by a working group consists of Biostatisticians, Statistical Programming, Medical Monitor and Medical Writing who worked together to develop this document. After approving and releasing this Biometrics standard TFLs within Biometrics, the Statistical Programming group initiated a project to develop macros to generate those TFLs stipulated in the Biometrics Standard TFLs document. Under this project, a process has been developed which consists of the following 4 steps:

- a. Planning
- b. Development
- c. User Acceptance Testing (UAT)
- d. Release of beta version to be used across the therapeutics.

a. Planning:

In the planning phase we developed a working group to work on developing the macros aligned with Biometrics Standard TFLs specification. The working group came up with a modular approach of design architecture. The main principle of this design architecture was to follow the CDISC ADaM standard datasets and variables so that we can have minimum number of macro parameters to gain the efficiency and the modular approach is for the better management of the macros. In this phase we first identified the number of macros will be needed to cover all the standard TFLs stipulated in our Biometrics Standard TFLs specification document and then developed specification for each macro. The macro specification document contains references of macro called programs and department level existing macros to be used, name of the CDISC ADaM data set to be used, a follow chart and an annotation of TFL shell to describe the design architecture. Another principle was to maximize the utilization of our existing department level macros to avoid redundancy in terms of developing.

Module Structure Approach:

- Department Macro Library:
- Macro_1
- Macro_2
- Macro_3
- *
- *
- *
- Macro_n

- **Examples:** Select Population (`%selpop`); Map Treatment (`%trtmapping`); Derive Statistics (`%cntfrqbyclass_sort`); etc.

Template/Macro Assembly:

```

graph TD
    A[code] --> B[Macro_1]
    B --> C[code]
    C --> D[Macro_x]
            
```

"Write once --> Use many times"

Below is an example of `%ae_reporting` macro specification:

References:

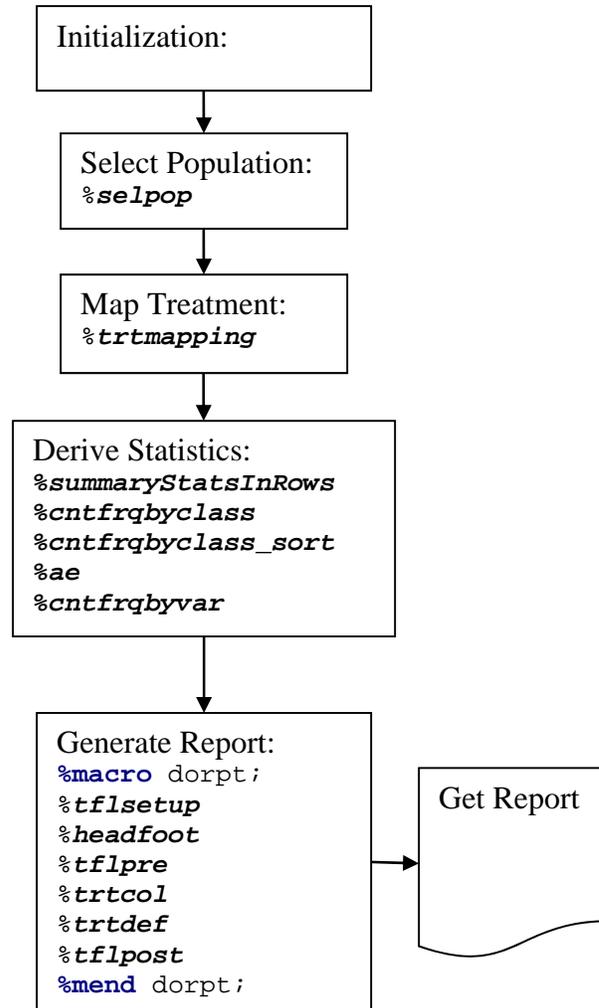
Type	Element	Location
Main Program	ae_reporting.sas	...\ <i>Reporting Event</i> \macrolib
Program(s) Called	t-aept.sas t-aesocpt.sas ----- t-aesummary.sas	...\ <i>Reporting Event</i> \tables
Macro(s)	<i>selpop</i> <i>trtmapping</i> ----- <i>cntfrqbyvar</i>	...\ <i>Reporting Event</i> \macrolib

Data Sources:

Data Set	Description	Required Variables	Required Formats
Adsl	Population	&_usubjid., &_trtcd.	&_trfmt. (&_trtcd.)
Adae	Adverse Events	&_usubjid. aedecod, aebodsys	

Note(s):

1. Variables usubjid and trt01an are the default variable names associated to the macro variables &_usubjid and &_trtcd. The macro variables are created and assigned in the study initialization file (...*Reporting Event*\standard.sas).
2. ADaM datasets located in directory ...*Reporting Event*\data\analysis and SDTM datasets located in directory ...*Reporting Event*\data\sdtm.

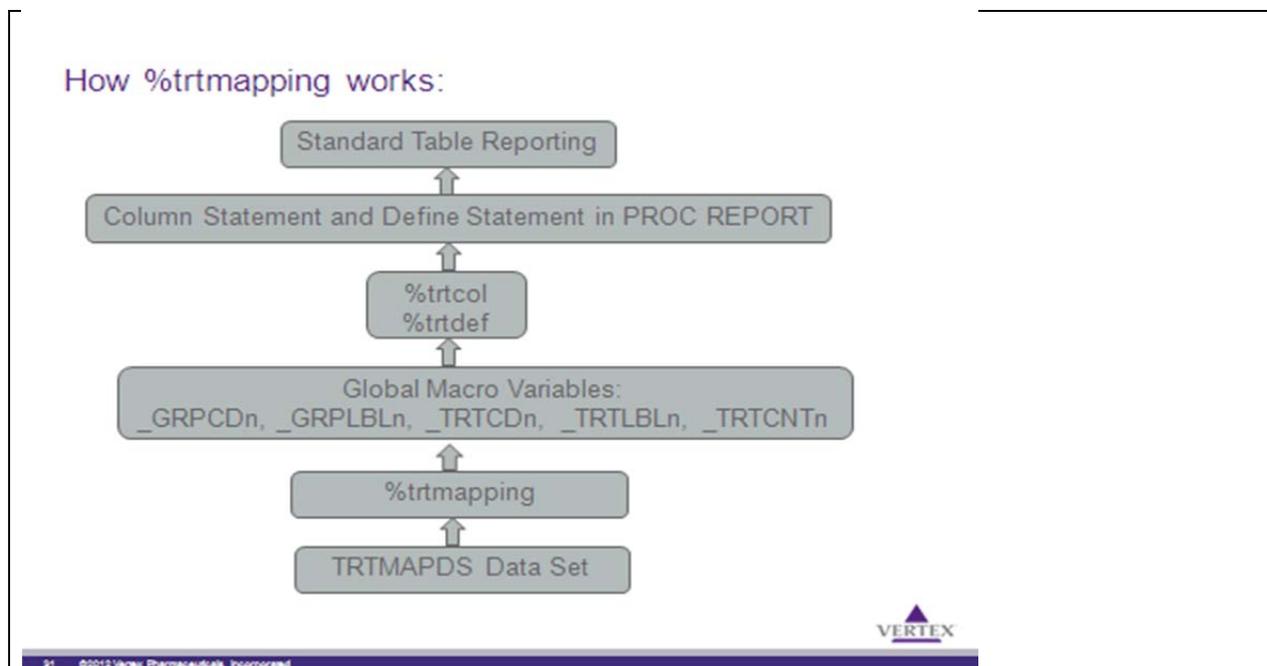
Process Flow Chart:

b. Development:

We developed the macros considering the following assumptions which were implemented across all the macros:

- i. To identify the population of the TFLs (e.g. Safety Set, Full Analysis Set, etc.), the macro %selpop will be used.
- ii. The column header for treatment column in the report will be generated using %trtmapping macro and
- iii. Default Titles and footnotes will come from study specific %headfoot macro.
- iv. Standard macro parameters name and specification across all the macros.

Flow of %trtmapping macro:



Parameters Specifications:

The following are the common parameters used across all 5 the macros we developed. The specified values of these parameters are not case sensitive.

POPDS = SAS-data-set

Identifies the input population SAS data set. The population in this data set determines the population in the report.

As an option for POPDS, after a /, user can specify the sub setting condition for the population to be selected for the analysis. For example, POPDS=%str(ADSL/SAFFL). The default option is FASFL.

It is a required parameter.

CLASSDS = SAS-data-set

Identifies the input analysis SAS data set.

It is a required parameter.

USUBJID

Identifies the input subject variable.

It is a required parameter. Default: &_usubjid.

TRT

Identifies treatment to be analyzed.

It is an optional parameter. Default: TRT01AN.

RPTNAME

Identifies the type of analysis and report table layout, where

DISP

Defines that the subject disposition will be summarized across treatments. It will generate t-disposition report as per standard.

***** Other parameter specification *****

OUTDS = SAS-data-set

Specifies the name of an output SAS data, which contains the same information as in the report if a report is created. If a data set is specified, the macro produces the data set only. If nothing specified, the macro creates a report.

It is an optional parameter. Default: null

DEBUG

Specifies a value of 1 or 0 to determine if debug mode is on (1) or off (0).

It is an optional parameter. Default: 0

In the development phase we developed the following 5 macros to cover our standard safety TFLs:

%nonsafety_reporting:

Macro nonsafety_reporting performs statistical analyses by category for variable that can be classified into continuous or category type. All the parameters in this macro are keyword parameters. The report created by this macro is aligned with the report layout in Biometrics Standard TL Shells V1.0 published. Specifically, this macro produces the following reports stipulated in Standard TL Shells V1.0:

- 1) Subject Baseline Characteristics
- 2) Subject Medical History Conditions (Incidence \geq A% in One or More Treatment Groups)
- 3) Summary of Subject Disposition
- 4) Study Drug Exposure
- 5) Summary of Subject Compliance

Input Datasets:

Macro nonsafety_reporting requires that input data sets comply with CDISC Vertex ADaM Standards. Here is the must-have ADaM variables in input data sets used in macro nonsafety_reporting call. These are the example of parallel design clinical trials:

- **ADSL** data set - USUBJID, TRT01A, TRT01AN, SEXN, AGE, RACEN
- **ADMH** data set – USUBJID, TRT01A, TRT01AN, MHDECOD, MHTERM

There may be other data sets and variables required based on the specific report requirement.

Table 1 t_disposition.doc: Summary of Subject Disposition

```

/* Initialization */

%include "..\standard.sas";
%let tblname=%bquote(%scan(&PROGNAME, 1, .));

/* Get the data */

proc datasets library=work nolist nodetails;
  copy in=ad out=work;
  select adsl;
  quit;
run;

/* Process data */

data adsl;
  set adsl;
  label randfl = 'Randomized';
  label nomedfl = 'No Study Medication';
  label fasfl = 'Included in Full Analysis Set';
run;

proc format;
  value streasn
    1 = 'Adverse Event'
    2 = 'Death'
    -----
    6 = 'Other'
  ;
run;

/* Call macro to generate report */

%nonsafety_reporting(popds = adsl,
                    classds = adsl,
                    rptname = DISP,
                    binaryvar = randfl nomedfl fasfl,
                    varord = randfl nomedfl fasfl,
                    format = streasn streasn.);

```

%ae_reporting:

Macro ae_reporting performs statistical analyses for variable that can be classified into several levels, such as PTERM, SOCTERM, etc. All the parameters in this macro are keyword parameters. The report created by this macro is aligned with the report layout in Biometrics Standard TL Shells V1.0 published. Specifically, this macro produces the following reports stipulated in Standard TL Shells V1.0:

- 1) Summary of Adverse Events
- 2) Summary of Adverse Events by System Organ Class and Preferred Term (PT Incidence \geq A% in One or More Treatment Groups)
- 3) Summary of Subjects With Adverse Events by Relationship to Study Drug, System Organ Class and Preferred Term
- 4) Summary of Subjects With Adverse Events by Relationship to Study Drug, System Organ Class and Preferred Term (Relationship presented in a column)
- 5) Summary of Adverse Events by Preferred Term (PT Incidence \geq A% in One or More Treatment Groups)

Input Datasets:

Here is the must-have ADaM variables in input data sets used in macro ae_reporting call:

- **ADSL** data set - USUBJID, TRT01A, TRT01AN
- **ADAE** data set – USUBJID, TRT01A, TRT01AN, AEDECOD, AETERM, AEBODSYS, AREL, ARELN, ATOXGR, ATOXGRN

Table 2 t_ae_soc_pt.doc: Summary of Adverse Events by System Organ Class and Preferred Term

```

/* Initialization */

%include "..\standard.sas";
%let tblname=%bquote(%scan(&PROGNAME, 1, .));

/* Get the data */

proc datasets library=work nolist nodetails;
  copy in=ad out=work;
  select adsl adae;
quit;
run;

/* Process data */

proc sort data=adae;
  by &_usubjid.;
  where aeecod ne "" and upcase(aetvpbfl)='Y';
run;

%ae_reporting( popds    = adsl,
               classds  = adae,
               rptname   = SOCPT,
               missing   = Yes,
               sortcols  = 10);

```

%summarybyvisit_reporting:

Macro summarybyvisit_reporting performs statistical analyses by visit or by category for variable that can be classified into continuous or category type. All the parameters in this macro are keyword parameters. The report created by this macro is aligned with the report layout in Biometrics Standard TL Shells V1.0 published. Specifically, this macro produces the following reports stipulated in Standard TL Shells V1.0:

- 1) Summary of Laboratory Results and Change from Baseline
- 2) Summary Statistics for Standard Digital ECG by Treatment Group and Visit
- 3) Summary of Vital Signs Results and Change from Baseline
- 4) Maximum On-Treatment QT/QTc Interval Results
- 5) Maximum On-Treatment Increase from Baseline in QT/QTc Interval

Input Datasets:

Here is the must-have ADaM variables needed for summarybyvisit_reporting macro call. These are the example of parallel design clinical trials:

- **ADSL** data set - USUBJID, TRT01A, TRT01AN
- **ADLB** data set - USUBJID, PARAMCD, PARAMN, AVAL, AVISIT, AVISITN

Table 3 t_eg_summary.doc: Summary Statistics for Standard Digital ECG by Treatment Group and Visit

```

/* Initialization */

%include "..\standard.sas";
%let tblname=%bquote(%scan(&PROGNAME, 1, .));

/* Get the data */

proc datasets library=work nolist nodetails;
  copy in=ad out=work;
  select adsl adeg;
quit;
run;

/* Process data */

data adeg;
  set adeg;
  where upcase(anl02fl)='Y' and aval ne . and avisitn ne .;
run;

/* Call macro to generate report */

%summarybyvisit_reporting( popds      = %str(adsl/SAFFL),
                          classds    = adeg,
                          rptname     = eg,
                          analvar    = aval chg);

```

%shift_reporting:

Macro shift_reporting performs statistical analyses for Laboratory and ECG baseline vs. post-baseline variables. The report created by this macro is aligned with the report layout in Biometrics Standard TL V1.0. Specifically, this macro produces the following reports in Standard TL V1.0:

- 1) ECG Shift-from-Baseline including and not including column for "Missing"
- 2) Chemistry Shift from Baseline including and not including column for "Missing"
- 3) Hematology Shift from Baseline including and not including column for "Missing"
- 4) Number and Percentage of Subjects with Hematology Toxicity Grade Shifts
- 5) Number and Percentage of Subjects with Chemistry Toxicity Grade Shifts

Input Datasets:

Here is the list of ADaM variables needed for shift_reporting macro call. These are the example of parallel design clinical trials:

- **ADSL** data set - USUBJID, TRT01A, TRT01AN
- **ADLB** data set – USUBJID, TRT01A, TRT01AN, AVISITN, PARAM, LBRPTLBL, ANRIND, BTOXGRN, BTOXGRHN

Table 4 t_lb_shift_grade.doc: Chemistry Toxicity Grade Shift-from-Baseline

```

/* Initialization */

%include "..\standard.sas";
%let tblname=%bquote(%scan(&PROGNAME, 1, .));

```

```

/* Get the data */

proc datasets library=work nolist nodetails;
  copy in=ad out=work;
  select adsl adlb;
  quit;
run;

/* Process data */

proc sort data=adlb out=adlb nodupkey;
  where abfl='Y' and parcat1="CHEMISTRY" and TABLESFL='Y';
  by usubjid lbrptl BTOXGRN MXGR_T;
run;

/* Call macro to generate report */

%shift_reporting(popds=adsl,classds=adlb,rptname=CHEMTOX,base=BTOXGRN,shift=MXGR_T,thr=hcv,debug=1)
;

```

%cm_reporting:

Macro `cm_reporting` performs statistical analyses for variable that can be classified into several levels, such as ATC1TERM, ATC2TERM, CMDECOD. All the parameters in this macro are keyword parameters. The report created by this macro is aligned with the report layout in Biometrics Standard TL V1.0. Specifically, this macro produces the following reports in Standard TL V1.0:

- 1) Number and percentage of Subjects with concomitant medications by Treatment Group, ATC1, ATC2, PT

Input Datasets:

Here is the list of must-have ADaM variables needed for `cm_reporting` macro call. These are the example of parallel design clinical trials:

- **ADSL** data set - USUBJID, TRT01A, TRT01AN
- **ADCM** data set – USUBJID, ATC1TERM, ATC2TERM, CMDECOD, CMPRIOR

Table 5 t_conmed.doc: Number and percentage of Subjects with concomitant medications by Treatment Group, ATC1, ATC2, PT

```

/* Initialization */

%include "..\standard.sas";
%let tblname=%bquote(%scan(&PROGNAME, 1, .));

/* Get the data */

proc datasets library=work nolist nodetails;
  copy in=ad out=work;
  select adsl adlb;
  quit;
run;

/* Call macro to generate report */

%cm_reporting(  trtvar = trt01an,
                popfl  = saffl,
                phase  = cmtvpbfl,

```

```
sortcols = 9 10,
options = atcpt);
```

Figures:

%Merrorbar (Error Bar Plot):

Macro %Merrorbar produces a Kaplan-Meier plot for the input analysis dataset.

Input Dataset

Macro %Merrorbar requires that the input dataset be of the structure as below.

	AVISIT	TRT01A	AVISITN	N	_NAME_	YVALUE	TRT01AN
1	BL	Pop B	0	14	lo	6.3375857982	1
2	BL	Pop B	0	14	mean	6.6833152447	1
3	BL	Pop B	0	14	hi	7.0290446911	1
4	4	Pop B	4	14	lo	0.8175023054	1
5	4	Pop B	4	14	mean	1.1017145048	1
6	4	Pop B	4	14	hi	1.3859267042	1
7	12	Pop B	12	9	lo	0.5053253246	1
8	12	Pop B	12	9	mean	0.8472426434	1
9	12	Pop B	12	9	hi	1.1891599621	1
10	24	Pop B	24	1	lo	0.6989700043	1
11	24	Pop B	24	1	mean	0.6989700043	1
12	24	Pop B	24	1	hi	0.6989700043	1
13	FUW4	Pop B	52	1	lo	0.6989700043	1
14	FUW4	Pop B	52	1	mean	0.6989700043	1
15	FUW4	Pop B	52	1	hi	0.6989700043	1

For every visit/timepoint, three records exist; the midpoint of the error bar which can either be the mean, median etc i.e. _NAME_="mean", one for the lower bar (_NAME_="lo") and the other for the upper bar(_NAME_="hi"). Missing values are allowed. Both the numeric as well as character versions of the variable for the time point (X-axis). Also treatment identifier TRT01A and TRT01AN are required. "N" variable represents the number of subjects at each visit and is required only if the number of subjects at each visit need to be displayed below each time point. The character values of the grouping variable (TRT01A) and the time point variable (AVISIT) are used as labels for the legend and X-axis respectively.

Figure 1 f_errorbar1: Error Bar plot for efficacy measure

```
/* Initialization */

%include "..\standard.sas";
%let tblname=%bquote(%scan(&PROGNAME, 1, .));

/* Get the data */

proc datasets library=work nolist nodetails;
  copy in=ad out=work;
  select adsl adhc;
  quit;
run;

/* Process data */

data srcdata;
  set adhc;
  if avisitn=950 then avisitn=0;
  else if avisitn=1001.06 then avisitn=0.1;
```

```

-----;
else if avisitn=8888 then avisitn=52;
run;

proc means data=srcdata noprint;
  by trt01an trt01a avisitn avisit;
  var aval;
  output out=wtg4(drop=_type__freq_) n=n mean=mean stderr=stderr lclm=lo uclm=hi;
run;

proc transpose data=srcdata out=final prefix=bar;
  by trt01an trt01a avisitn avisit mean stderr n;
  var lo mean hi;
run;

%merrorbar( inds=final,
            err_mean=mean,
            yvalue=bar1,
            xlabel=%nrstr(Study Week),
            ylabel=%str(Mean and 95% CI of Log HCV RNA),
            leg_pos=(inside left top),
            ticks=C,
            countvar=y);

```

Listings:

For listings, we have developed template programs instead of macros. Most of the listings were done using CDISC SDTM data sets. The following two module macros used in the listing programs.

%checkvar.sas -- automatically detect if information for optional column exist, if yes, the column will be created.
%varlen.sas -- set variable length in report

c. User Acceptance Testing (UAT):

We first developed Test Specification and Test Script for each of the macro. The UAT of the macros were done while working on the real life project. We developed QC programs for standard TFLs without using any macros and then compare the production reports with that of QC of reports. With this principle, we have been able to test the functionality of the macros as well as have been able to deliver the project deliverable with quality, speed and values.

Example of Test Specification for ae_reporting macro:

Test Specifications:

Implementation of the tests described below is done using the document
 ...\\standard_table\reference_documents\qc\ae_reporting – TestScript001.doc.

ID	Step	User Requirement Reference	Test Description	Expected Results
TST-1	1	N/A	Capture the current date and time when execution of the test script starts and report it in the Actual Results section of the Test Script document.	Date captured. Date and time will be referred to, in following steps, as the Reference date.
TST-2	1	N/A	Check if required datasets listed in the Input Datasets section of the document ...\\standard_table\reference_docum	Datasets found and consistent with document.

			ents \ User Guide for ae_reporting.doc exists: ...\data\analysis\adsl.sas7bdat ...\data\analysis\adae.sas7bdat	
TST-3	1	USR-1	<p>Generate report 't-ae-summary.doc' using the macro from ...\standard_table\macrolib \ae_reporting.sas</p> <p>Use parameter RPTNAME = AE in the ae_reporting macro calling to generate t-ae-summary.doc.</p> <p>Compare the overall layout against the 't_ae_summary.doc' shell found in ...\standard_table\reference_documents\ DRAFT standard TL V9.doc</p> <p>Take an overall visual look at the results layout and values for any issues.</p>	<p>No specific issues detected.</p> <p>Header/Titles/Footnotes are consistent with Shell.</p>
TST-4	1	USR-1	<p>Generate the report 't-ae-summary_compare.doc' using the compare macro call %qcCompareDocumentBody(arg=,prefix=)by comparing the report generated without the ae_reporting macro call with the one generated with ae_reporting macro call.</p> <p>Review 't-ae-summary_compare.doc' for any issues related to the table core results (statistics).</p>	<p>No differences associated to the table results (statistics).</p>
-----	-----	-----	-----	-----
TST-n	-----	-----	-----	-----

e. Release of beta version to be used across the therapeutics:

After completing the UAT of each of the macro working on projects from different therapeutics, we developed the plan for roll of these macros within Biometrics Department. The plan consists of following:

- i. Developed User Guide for each of the macro containing example of macro calls.
- ii. Developed template program for macro calls for each of the standard TFL.
- iii. Developed '**Standard TFL User Manual**' stipulating functionality of all the macros along with example macro calls in different TFLs.
- iv. Developed training slides and provide live demonstration of macro calls and functionality within Biometrics Department in a macro training session.
- v. Identified resources to address questions or concerns, if any brought by the users of the macros.

CONCLUSION:

In summary, the SAS macros and template programs we developed can be used to automate the standard TFLs generation aligned with standard TFL shells document. The macros have been developed considering ADaM data sets which are based on CDISC and Vertex standards and conventions which helped us to restrict the number of macro parameters at a minimum to enable the user to use the macro efficiently without know the greater details of the

macros. Due to the modular approach of the macros, the managing of the macros become more robust as for any updates we need to make, we can update the associated module instead of updating the whole reporting macro. These macros can be used for any number of treatment columns and the width of treatment column can be handled using macro parameter. For any updates of a table shell, the module macro can be re-run and automatically update headfoot.sas which contains title and footnote information. Once the table shell document is setup, without the presence of special characters and formats, this process requires no human intervention, hence facilitates a seamless approach. The use of these macro result in significant reduction of programming work load and error-prone manual processing to define the treatment columns header and to obtain the titles and footnotes, especially following the modifications to table shells. It also helps to limit the need to manually enter information for project management activities. Another notable benefit comes from its built-in capability of producing output reporting data set to help the QC programmer to QC the report in the event of large reporting data set like coming out of LAB or Vital Signs related ADaM data set, therefore ensure complete implementation of the table shell requirements. The macros are easy to use, and improves both work efficiency and quality.

ACKNOWLEDGEMENTS:

Appreciation goes to Qunming Dong, PhD and Leif Bengtsson for their review and comments.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are registered trademarks or trademarks of their respective companies.

CONTACT INFORMATION:

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

Xinagchen (Bob) Cui, Ph.D.

Vertex Pharmaceuticals Incorporated
130 Waverly Street
Cambridge, MA 02139 - 4942
617-341-6069

Mominul Islam

Vertex Pharmaceuticals Incorporated
130 Waverly Street
Cambridge, MA 02139 - 4942
617-961-0913

Jiannan Hu

Vertex Pharmaceuticals Incorporated
130 Waverly Street
Cambridge, MA 02139 - 4942
617-961-7524

Sanjiv Ramalingam

Vertex Pharmaceuticals Incorporated
130 Waverly Street
Cambridge, MA 02139 - 4942
617-961-0831

Yanwei Han

Vertex Pharmaceuticals Incorporated
130 Waverly Street
Cambridge, MA 02139 - 4942
617-341-6736