PharmaSUG 2021 - Paper DS-165

Validation of CDISC specifications using VALSPEC utility macro

Hrideep Antony, Syneos Health, Morrisville, North Carolina, USA

Aman Bahl, Syneos Health, Ontario, Canada

ABSTRACT

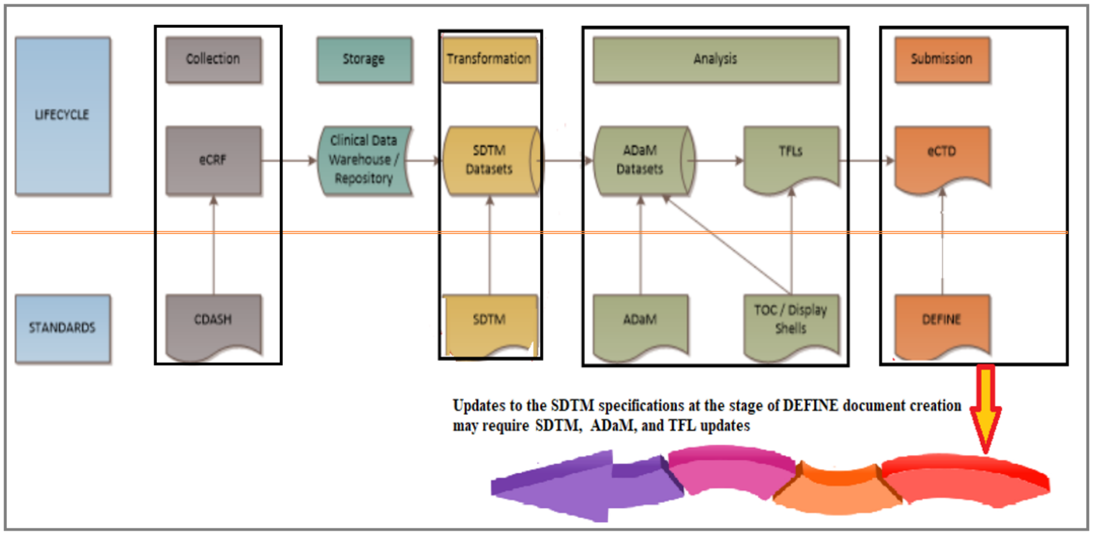
The define.xml is the cover section of the electronic common technical document (eCTD) submission and it provides a high-level summary of the metadata for all the data submitted for an NDA (new drug application). Define.xml will not only aid in the ease of regulatory review but also conveys a message on the overall quality of the submitted work to the reviewers and a good quality one can augment their trust in the results that are being submitted. Define.xml document quality is widely driven by the data and the specifications that are used to generate the file.

The accuracy of the SDTM/ADaM specifications is determined by parameters such as data compliance and the usage of controlled terminology. However, the majority of the discrepancies are only identified after the pinnacle validation. Specification updates based on the compliance findings while creating the final CRT package may result in post-lock updates to the SDTM/ADaM and TFLs. These post-lock updates can cause significant submission delays.

The process of identifying the issues in the specification document does not have to wait that long! This paper introduces an approach and a utility that will alert the author about the issues at an early stage of the creation of the data transformation process. Identifying the issues at an early stage will not only improve the quality of the overall submission but will also reduce the rework that is required and further ease the submission process.

INTRODUCTION

The lifecycle of the NDA process begins with the data collection stage to the submission stage as shown in figure 1 below. Most of these stages are well regulated by predetermined standards set by the regulatory agency to facilitate the streamlining of the approval process. Non-conformance and deviations to the set compliance standards at any stage will have a ripple effect on the overall compliance requirements, resulting in undue delays in the approval process. Such deviations from the standards if identified at an early stage will avoid any undue revisions and rework.



Updates to the SDTM specifications at the stage of DEFINE document creation may require SDTM, ADaM, and TFL updates

Figure 1. Drug submission process lifecycle

VALSPEC utility that is introduced in this paper is used to validate SDTM and ADaM specifications along with the data and streamline standard implementation at an early data transformation stage. The mission of this utility is to find potential compliance conformance issues at an earlier stage and thus avoiding any late-stage revisions.

VALSPEC UTILITY

This utility consists of three key components as shown in the flowchart below:

- Validation of specification metadata to check for compliance with CDISC standards.
- Validation of the data to check for compliance
- Categorization and generation of findings based on the severity of findings

The key functionality of this utility will be further discussed in this paper.

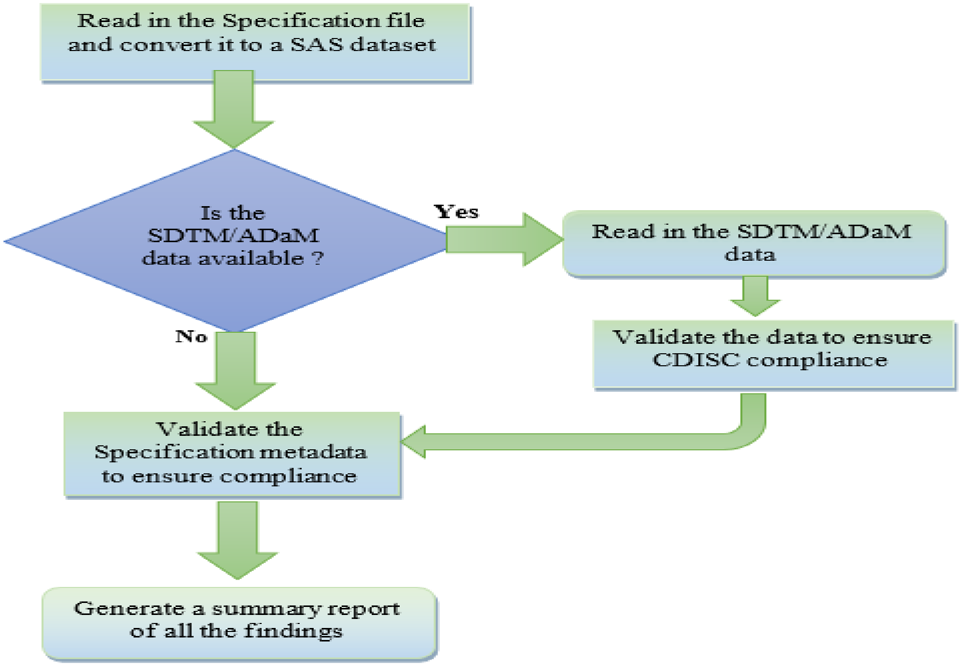

Read in the Specification file and convert it to a SAS dataset
Is the SDTM/ADaM data available ?
Yes
Read in the SDTM/ADaM data
Validate the data to ensure CDISC compliance
No
Validate the Specification metadata to ensure compliance
Generate a summary report of all the findings

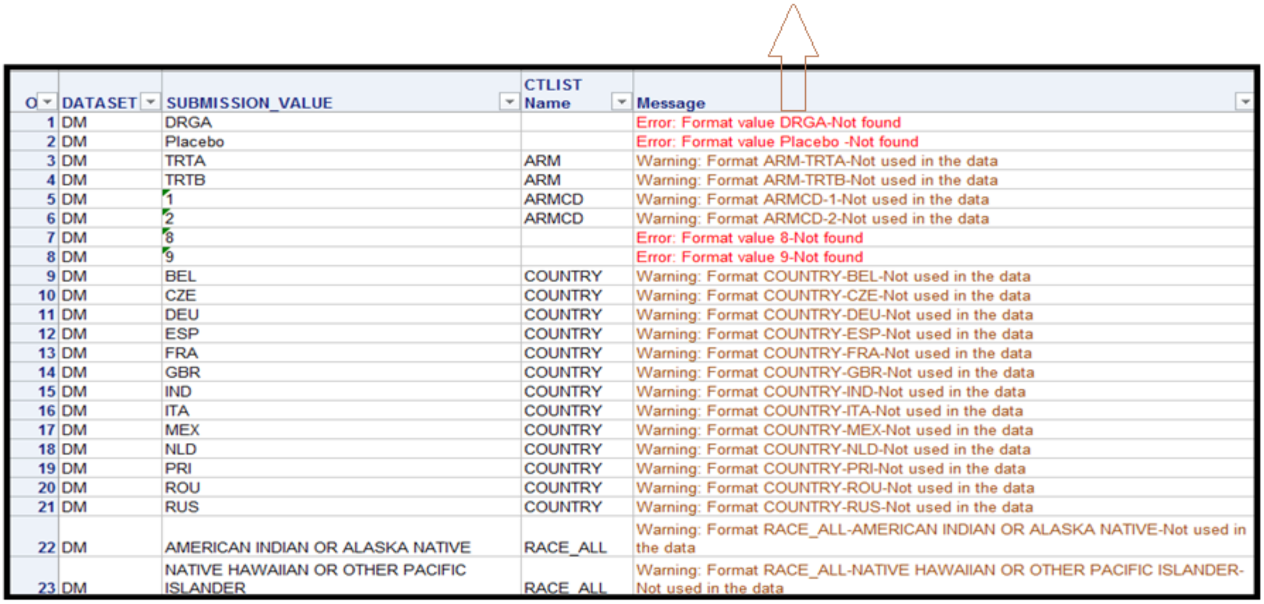
Figure 2. Process flowchart

VALSPEC SUMMARY OF FINDINGS OVERVIEW

One of the compliance checks that the utility performs is to check controlled terminology (CT) that is being referenced in the specification metadata and validate it against the standards.

Controlled terminology (CT) is the set of code lists and valid values used with data items within CDISC-defined datasets. Controlled terminology provides the values required for submission to FDA and PMDA in CDISC-compliant datasets. CDISC, in collaboration with the National Cancer Institute's Enterprise Vocabulary Services (EVS), supports the controlled terminology needs of CDISC foundational and therapeutic area standards. The references to the most current-controlled terminology can be referenced and downloaded at the CDISC website.

The summary of findings generated by the VALSPEC utility in this example of DM domain is classified as the error and warnings as shown in figure 3 below.



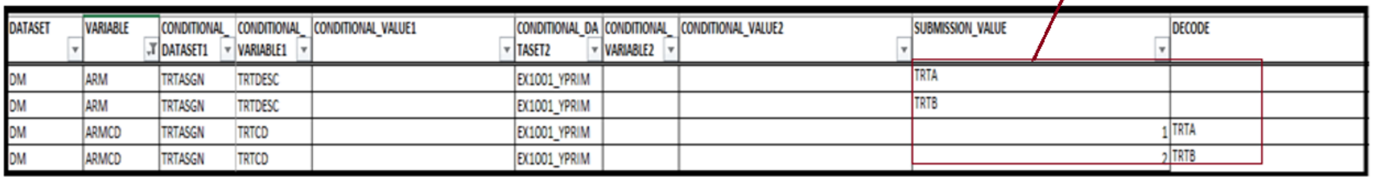
Error represents that there are data values that are not consistent with the values in the Specification

O	DATASET	SUBMISSION_VALUE	CTLIST Name	Message
1	DM	DRGA		Error: Format value DRGA-Not found
2	DM	Placebo		Error: Format value Placebo -Not found
3	DM	TRTA	ARM	Warning: Format ARM-TRTA-Not used in the data
4	DM	TRTB	ARM	Warning: Format ARM-TRTB-Not used in the data
5	DM	1	ARMCD	Warning: Format ARMCD-1-Not used in the data
6	DM	2	ARMCD	Warning: Format ARMCD-2-Not used in the data
7	DM	8		Error: Format value 8-Not found
8	DM	9		Error: Format value 9-Not found
9	DM	BEL	COUNTRY	Warning: Format COUNTRY-BEL-Not used in the data
10	DM	CZE	COUNTRY	Warning: Format COUNTRY-CZE-Not used in the data
11	DM	DEU	COUNTRY	Warning: Format COUNTRY-DEU-Not used in the data
12	DM	ESP	COUNTRY	Warning: Format COUNTRY-ESP-Not used in the data
13	DM	FRA	COUNTRY	Warning: Format COUNTRY-FRA-Not used in the data
14	DM	GBR	COUNTRY	Warning: Format COUNTRY-GBR-Not used in the data
15	DM	IND	COUNTRY	Warning: Format COUNTRY-IND-Not used in the data
16	DM	ITA	COUNTRY	Warning: Format COUNTRY-ITA-Not used in the data
17	DM	MEX	COUNTRY	Warning: Format COUNTRY-MEX-Not used in the data
18	DM	NLD	COUNTRY	Warning: Format COUNTRY-NLD-Not used in the data
19	DM	PRI	COUNTRY	Warning: Format COUNTRY-PRI-Not used in the data
20	DM	ROU	COUNTRY	Warning: Format COUNTRY-ROU-Not used in the data
21	DM	RUS	COUNTRY	Warning: Format COUNTRY-RUS-Not used in the data
22	DM	AMERICAN INDIAN OR ALASKA NATIVE	RACE_ALL	Warning: Format RACE_ALL-AMERICAN INDIAN OR ALASKA NATIVE-Not used in the data
23	DM	NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER	RACE_ALL	Warning: Format RACE_ALL-NATIVE HAWAIIAN OR OTHER PACIFIC ISLANDER-Not used in the data

Warnings are issued for redundant information in the specification file that is not used in the data

Figure 3. Sample output from VALSPEC utility

In the example below, it can be noted that the terminology used in the DM domain is not consistent with data as shown in figure 4 below, causing an error to be displayed.

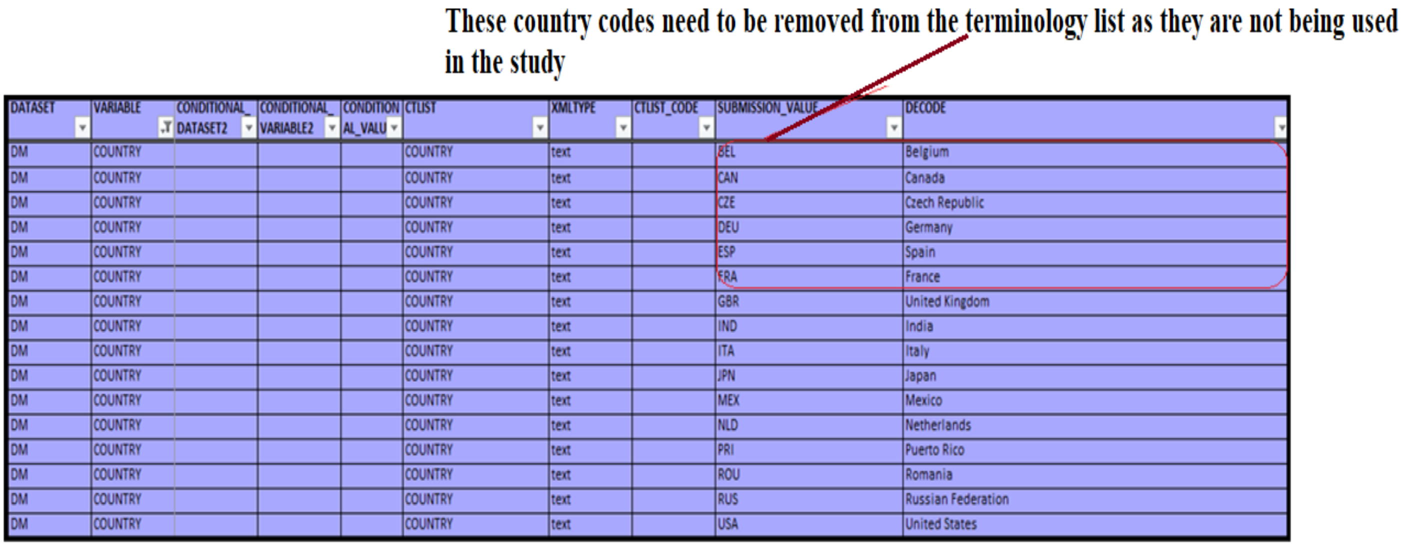


Inconsistent values in the submission value and SDTM data

DATASET	VARIABLE	CONDITIONAL_DATASET1	CONDITIONAL_VARIABLE1	CONDITIONAL_VALUE1	CONDITIONAL_DATASET2	CONDITIONAL_VARIABLE2	CONDITIONAL_VALUE2	SUBMISSION_VALUE	DECODE
DM	ARM	TRTASGN	TRTDESC		EX1001_YPRIM			TRTA	
DM	ARM	TRTASGN	TRTDESC		EX1001_YPRIM			TRTB	
DM	ARMCD	TRTASGN	TRTCD		EX1001_YPRIM			1	TRTA
DM	ARMCD	TRTASGN	TRTCD		EX1001_YPRIM			2	TRTB

Figure 4. The terminology used for ARM and ARMCD variables in DM

The warnings issued by the VALSPEC are due to the excess country codes that are not being removed from the DM domain but not used in the study as shown in figure 5 below.



These country codes need to be removed from the terminology list as they are not being used in the study

DATASET	VARIABLE	CONDITIONAL_DATASET2	CONDITIONAL_VARIABLE2	CONDITIONAL_VALU	CTLIST	XMLTYPE	CTLIST_CODE	SUBMISSION_VALUE	DECODE
DM	COUNTRY				COUNTRY	text		BEL	Belgium
DM	COUNTRY				COUNTRY	text		CAN	Canada
DM	COUNTRY				COUNTRY	text		CZE	Czech Republic
DM	COUNTRY				COUNTRY	text		DEU	Germany
DM	COUNTRY				COUNTRY	text		ESP	Spain
DM	COUNTRY				COUNTRY	text		FRA	France
DM	COUNTRY				COUNTRY	text		GBR	United Kingdom
DM	COUNTRY				COUNTRY	text		IND	India
DM	COUNTRY				COUNTRY	text		ITA	Italy
DM	COUNTRY				COUNTRY	text		JPN	Japan
DM	COUNTRY				COUNTRY	text		MEX	Mexico
DM	COUNTRY				COUNTRY	text		NLD	Netherlands
DM	COUNTRY				COUNTRY	text		PRI	Puerto Rico
DM	COUNTRY				COUNTRY	text		ROU	Romania
DM	COUNTRY				COUNTRY	text		RUS	Russian Federation
DM	COUNTRY				COUNTRY	text		USA	United States

Figure 5. Terminology list used in the DM domain having excess country codes

These excess country codes, if not removed will cause define.xml to populate unnecessary country codes that are not relevant to this study.

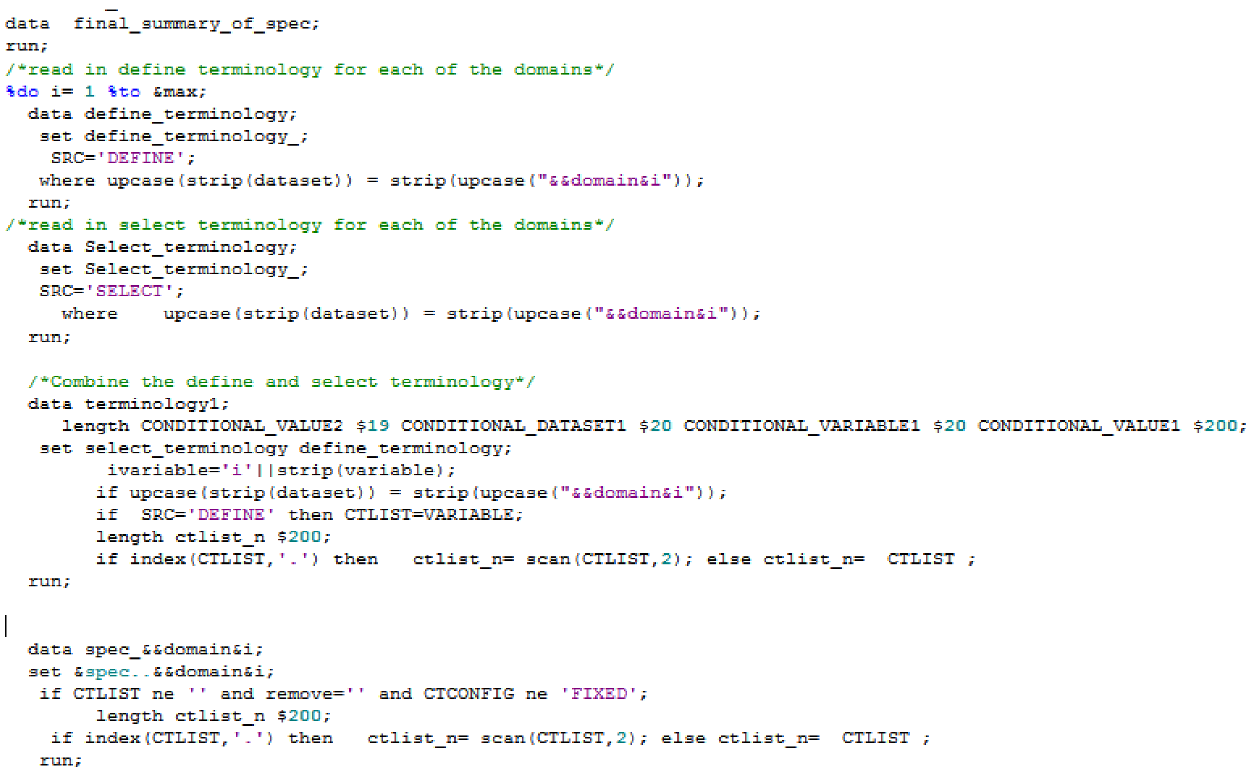
VALSPEC FUNCTIONALITY

The SAS code below is used in the VALSPEC utility to read in the terminology data from the DEFINE and SELECT terminology metadata.

```
/*Using the Tables tab information to select the domains that are needed to be valiadted*/
data Alldata;
set sdtm1.tables end=end;
   count+1;
 /*If dataname macro variable has ALL then validate all the domains in that study*/
  /* else validate only the spesific domain*/
   %if &dataname =ALL %then %do;
        where REMOVE=''    ;
   %end;
   %else %do;
        where REMOVE=''   and DATASET in(&dataname) ;
   %end;

       filenm_base=compress(DATASET,'~$');
       call symputx('domain'||put(count,4.-l),scan(filenm_base,1,'.'));
       /*Max macro variable will have the count of domains that need to be checked*/
           if end then call symputx('max',count);
run;
/*read in the Define terminology*/
%put &domain1;
  data define_terminology_;
  set &spec..define_terminology;
    SRC='DEFINE';
  run;

  /*read in the Select terminology*/
  data Select_terminology_;
   set &spec..select_terminology;
    SRC='SELECT';
    where REMOVE='';
  run;
```

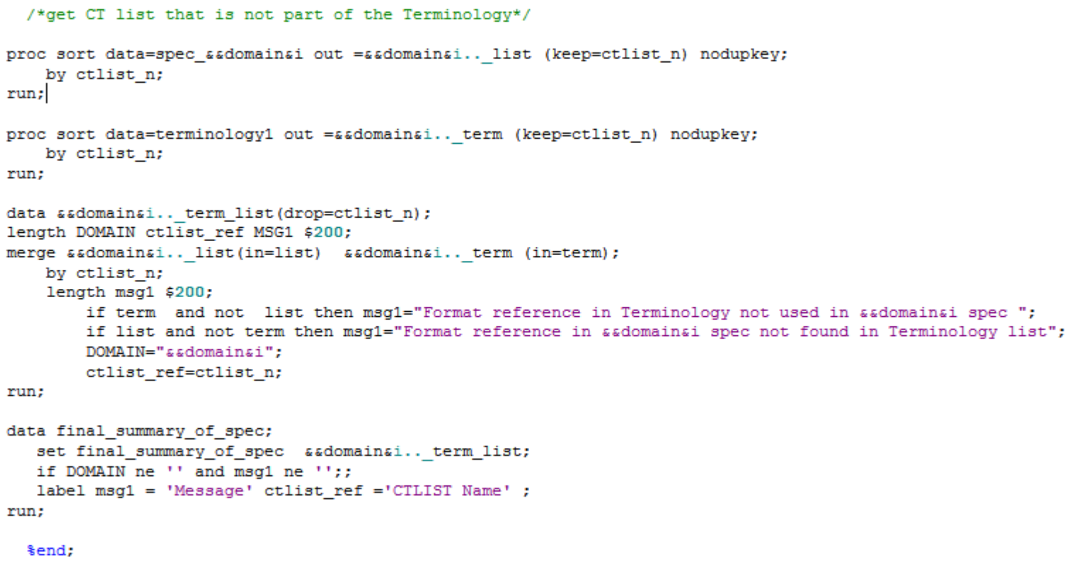
The macro variable “max” keeps a counter of all the domains that need to be checked using the utility. The parameter ‘dataname’ has a default value of “ALL” which indicates that the utility needs to check all the domains used in that study. If the VALSPEC only has to check one specific domain, then the corresponding domain name needs to be assigned to the “dataname” parameter.

Once the select and define terminology is being read, they are combined to create a repository of controlled terminology that is used in the study for each of the domains as shown in the code below. Note that the macro variable “i” will be used to distinguish the corresponding domains against the controlled terminology.



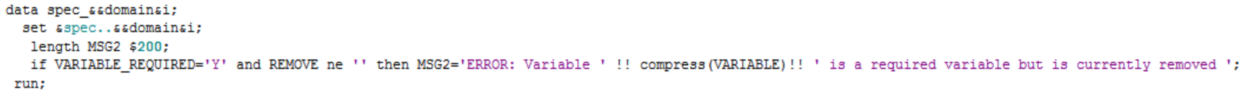
```
data  final_summary_of_spec;
run;
/*read in define terminology for each of the domains*/
%do i= 1 %to &max;
  data define_terminology;
   set define_terminology_;
    SRC='DEFINE';
   where upcase(strip(dataset)) = strip(upcase("&&domain&i"));
  run;
/*read in select terminology for each of the domains*/
  data Select_terminology;
   set Select_terminology_;
   SRC='SELECT';
     where    upcase(strip(dataset)) = strip(upcase("&&domain&i"));
  run;

  /*Combine the define and select terminology*/
  data terminology1;
     length CONDITIONAL_VALUE2 $19 CONDITIONAL_DATASET1 $20 CONDITIONAL_VARIABLE1 $20 CONDITIONAL_VALUE1 $200;
   set select_terminology define_terminology;
         ivariable='i'||strip(variable);
        if upcase(strip(dataset)) = strip(upcase("&&domain&i"));
        if  SRC='DEFINE' then CTLIST=VARIABLE;
        length ctlist_n $200;
        if index(CTLIST,'.') then   ctlist_n= scan(CTLIST,2); else ctlist_n=  CTLIST ;
  run;

  data spec_&&domain&i;
  set &spec..&&domain&i;
   if CTLIST ne '' and remove='' and CTCONFIG ne 'FIXED';
        length ctlist_n $200;
    if index(CTLIST,'.') then   ctlist_n= scan(CTLIST,2); else ctlist_n=  CTLIST ;
   run;
```

The controlled terminology references that are used in the specification document are not checked against the CT and corresponding messages are issued using the code below.

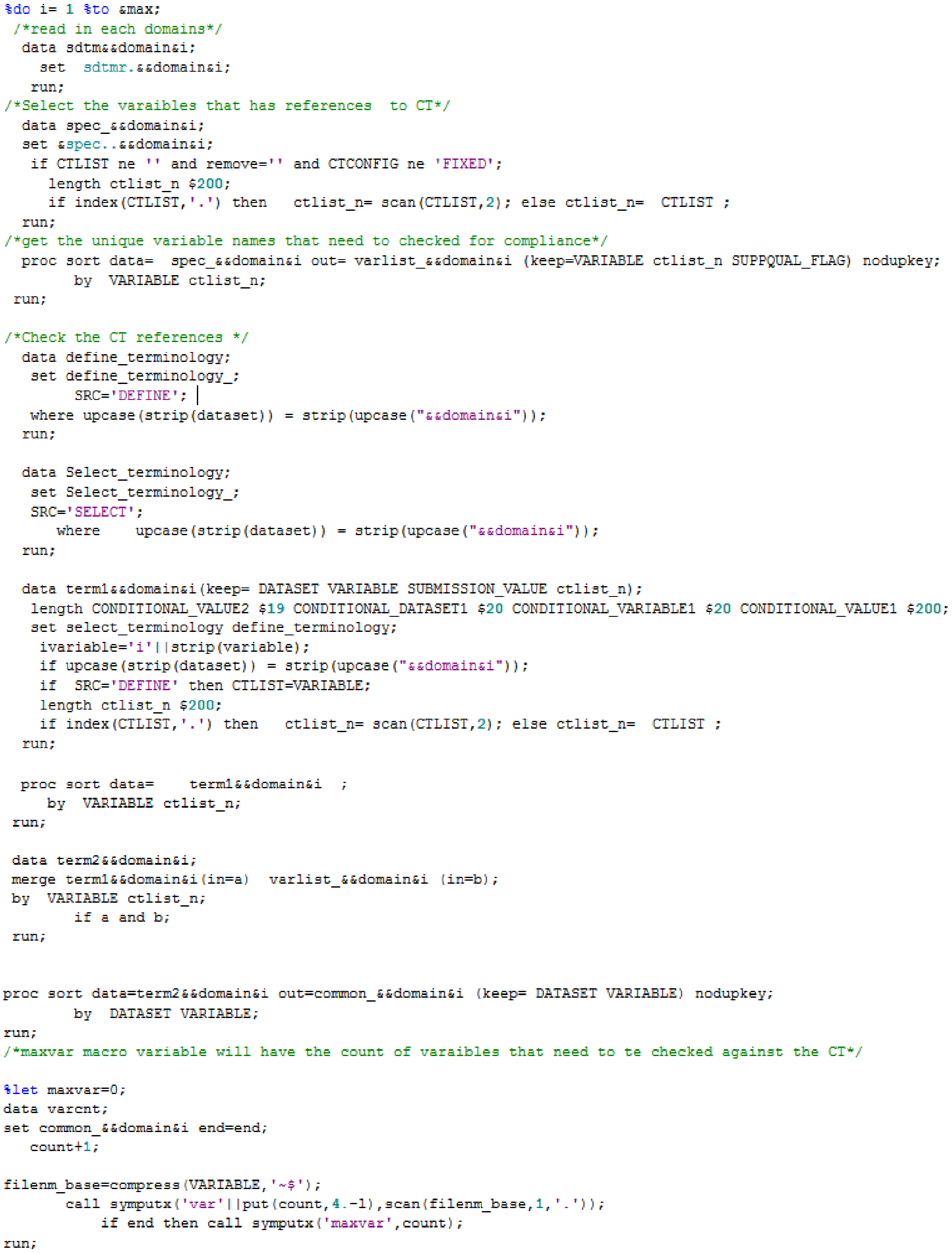


```
  /*get CT list that is not part of the Terminology*/

proc sort data=spec_&&domain&i out =&&domain&i.._list (keep=ctlist_n) nodupkey;
    by ctlist_n;
run;

proc sort data=terminology1 out =&&domain&i.._term (keep=ctlist_n) nodupkey;
    by ctlist_n;
run;

data &&domain&i.._term_list(drop=ctlist_n);
length DOMAIN ctlist_ref MSG1 $200;
merge &&domain&i.._list(in=list)  &&domain&i.._term (in=term);
    by ctlist_n;
    length msg1 $200;
        if term  and not  list then msg1="Format reference in Terminology not used in &&domain&i spec ";
        if list and not term then msg1="Format reference in &&domain&i spec not found in Terminology list";
        DOMAIN="&&domain&i";
        ctlist_ref=ctlist_n;
run;

data final_summary_of_spec;
   set final_summary_of_spec  &&domain&i.._term_list;
   if DOMAIN ne '' and msg1 ne '';;
   label msg1 = 'Message' ctlist_ref ='CTLIST Name' ;
run;

  %end;
```

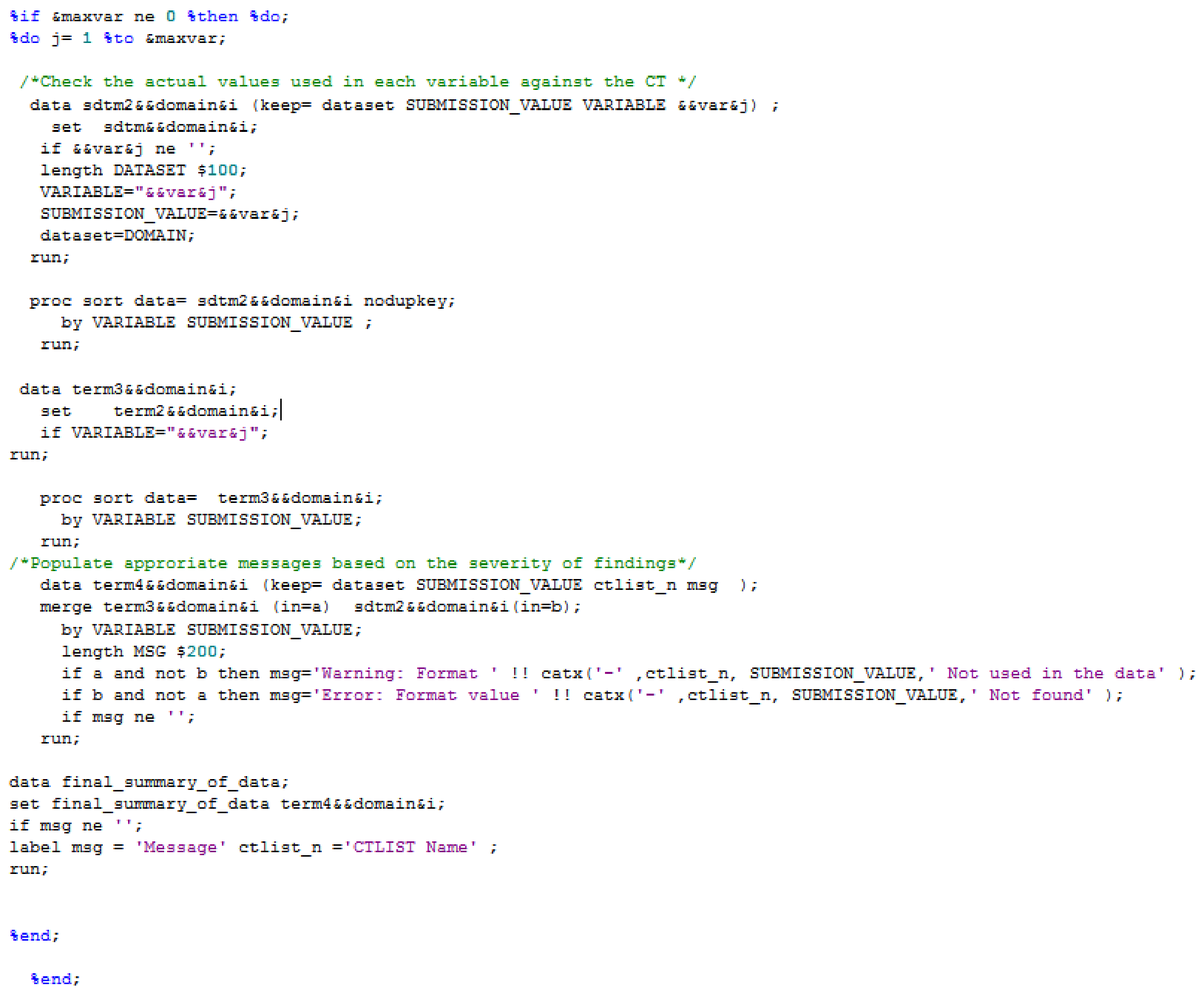
If a required variable is being removed in the specification an error message is created using the code below.



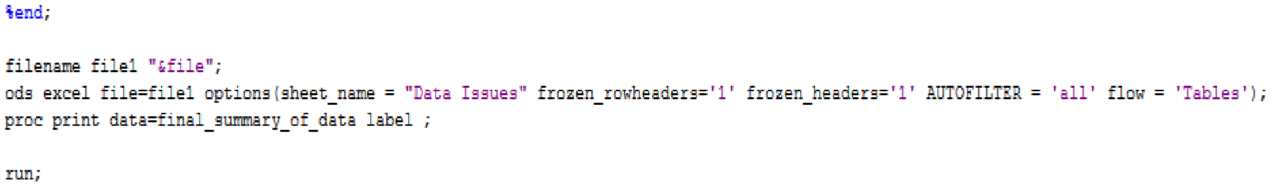
```
data spec_&&domain&i;
  set &spec..&&domain&i;
   length MSG2 $200;
   if VARIABLE_REQUIRED='Y' and REMOVE ne '' then MSG2='ERROR: Variable ' !! compress(VARIABLE)!! ' is a required variable but is currently removed ';
 run;
```



```
%do i= 1 %to &max;
 /*read in each domains*/
  data sdtm&&domain&i;
    set  sdtmr.&&domain&i;
   run;
/*Select the varaibles that has references  to CT*/
  data spec_&&domain&i;
  set &spec..&&domain&i;
   if CTLIST ne '' and remove='' and CTCONFIG ne 'FIXED';
     length ctlist_n $200;
     if index(CTLIST,'.') then   ctlist_n= scan(CTLIST,2); else ctlist_n=  CTLIST ;
  run;
/*get the unique variable names that need to checked for compliance*/
  proc sort data=  spec_&&domain&i out= varlist_&&domain&i (keep=VARIABLE ctlist_n SUPPQUAL_FLAG) nodupkey;
        by  VARIABLE ctlist_n;
 run;

/*Check the CT references */
  data define_terminology;
   set define_terminology_;
        SRC='DEFINE';
   where upcase(strip(dataset)) = strip(upcase("&&domain&i"));
  run;

  data Select_terminology;
   set Select_terminology_;
   SRC='SELECT';
      where    upcase(strip(dataset)) = strip(upcase("&&domain&i"));
  run;

  data term1&&domain&i(keep= DATASET VARIABLE SUBMISSION_VALUE ctlist_n);
   length CONDITIONAL_VALUE2 $19 CONDITIONAL_DATASET1 $20 CONDITIONAL_VARIABLE1 $20 CONDITIONAL_VALUE1 $200;
   set select_terminology define_terminology;
    ivariable='i'||strip(variable);
    if upcase(strip(dataset)) = strip(upcase("&&domain&i"));
    if  SRC='DEFINE' then CTLIST=VARIABLE;
    length ctlist_n $200;
    if index(CTLIST,'.') then   ctlist_n= scan(CTLIST,2); else ctlist_n=  CTLIST ;
  run;

  proc sort data=    term1&&domain&i   ;
     by  VARIABLE ctlist_n;
 run;

 data term2&&domain&i;
 merge term1&&domain&i(in=a)  varlist_&&domain&i (in=b);
 by  VARIABLE ctlist_n;
        if a and b;
 run;


proc sort data=term2&&domain&i out=common_&&domain&i (keep= DATASET VARIABLE) nodupkey;
        by  DATASET VARIABLE;
run;
/*maxvar macro variable will have the count of varaibles that need to te checked against the CT*/

%let maxvar=0;
data varcnt;
set common_&&domain&i end=end;
   count+1;

filenm_base=compress(VARIABLE,'~$');
       call symputx('var'||put(count,4.-l),scan(filenm_base,1,'.'));
           if end then call symputx('maxvar',count);
run;
```



```
%if &maxvar ne 0 %then %do;
%do j= 1 %to &maxvar;

 /*Check the actual values used in each variable against the CT */
  data sdtm2&&domain&i (keep= dataset SUBMISSION_VALUE VARIABLE &&var&j) ;
    set  sdtm&&domain&i;
   if &&var&j ne '';
   length DATASET $100;
   VARIABLE="&&var&j";
   SUBMISSION_VALUE=&&var&j;
   dataset=DOMAIN;
  run;

  proc sort data= sdtm2&&domain&i nodupkey;
     by VARIABLE SUBMISSION_VALUE ;
   run;

 data term3&&domain&i;
   set    term2&&domain&i;
   if VARIABLE="&&var&j";
run;

   proc sort data=  term3&&domain&i;
     by VARIABLE SUBMISSION_VALUE;
   run;
/*Populate approriate messages based on the severity of findings*/
   data term4&&domain&i (keep= dataset SUBMISSION_VALUE ctlist_n msg  );
   merge term3&&domain&i (in=a)  sdtm2&&domain&i(in=b);
     by VARIABLE SUBMISSION_VALUE;
     length MSG $200;
     if a and not b then msg='Warning: Format ' !! catx('-' ,ctlist_n, SUBMISSION_VALUE,' Not used in the data' );
     if b and not a then msg='Error: Format value ' !! catx('-' ,ctlist_n, SUBMISSION_VALUE,' Not found' );
     if msg ne '';
   run;

data final_summary_of_data;
set final_summary_of_data term4&&domain&i;
if msg ne '';
label msg = 'Message' ctlist_n ='CTLIST Name' ;
run;


%end;

  %end;
```



```
%end;

filename file1 "&file";
ods excel file=file1 options(sheet_name = "Data Issues" frozen_rowheaders='1' frozen_headers='1' AUTOFILTER = 'all' flow = 'Tables');
proc print data=final_summary_of_data label ;

run;
```

The VALSPEC utility macro is invoked using the macro call %valspec with the parameters as shown below. Dataloc and specloc parameters are used to locate the data and specification location.

```
%valspec(dataloc=,specloc=, ckdata=,dataname=);
```

This utility can be expanded based on the requirement of individual organizational requirements and evolving compliance requirements.

CONCLUSION

The quality of the SDTM/ADaM specifications has a key role in determining the quality of the define.xml document, which is a key part of the CRT package submitted to the regulatory agencies for approval.

Specification updates based on the compliance findings while creating the final CRT package may result in post-lock updates to the SDTM/ADaM and TFLs. These post-lock updates can cause significant submission delays.

VALSPEC utility is designed to alert such non-compliance aspects at an earlier stage thereby streamlining the submission process and minimizing any rework risks.

REFERENCES

How can we ensure our study data is FAIR (Findable, Accessible, Interoperable, and Reusable)? https://www.lexjansen.com/phuse/2019/ds/DS01.pdf

Automate Process to Ensure Compliance with FDA Business Rules in SDTM Programming for FDA Submission:https://www.sas.com/content/dam/SAS/support/en/sas-global-forum-proceedings/2019/3790-2019.pdf

Introduction to the CDISC Standards: https://www.lexjansen.com/pharmasug/2013/IB/PharmaSUG-2013-IB06.pdf

ACKNOWLEDGMENTS

Sincere thanks to Steve Benjamin, Director Statistical Programming, Biostatistics and Global Contracts and Aman Bahl, Associate Director, Statistical Programming, Clinical Division for their vision, great leadership, persistent support, and encouragement throughout and for their valuable assistance in reviewing this paper.

CONTACT INFORMATION

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

Hrideep Antony
Principal Statistical Programmer, Clinical Division, Syneos Health
Work Phone: +1- 984 459 4785
Email: hrideep.antony@syneoshealth.com
Web: http:/www.syneoshealth.com

Aman Bahl
Associate Director, Statistical Programming, Clinical Division, Syneos Health
Phone: +1-905 399 6715
E-mail: Aman.Bahl@syneoshealth.com
Web: http://www.syneoshealth.com