



PhUSE US Connect 2019

Paper CT09

Simplifying P21 Process for Define Creation

Ghanshyam Jangid, Merck & Co., Inc., North Wales, PA, USA

ABSTRACT

Define.xml files are currently required for FDA submissions. Pinnacle 21 (P21) Enterprise can be used to generate define.xml using an ADaM specification. The purpose of the define.xml is to inform the reviewer on what datasets, variables, controlled terms, and other specified metadata were used for building ADaM datasets. During the process of creating the Define.xml, many issues can be encountered. One such issue was the need to identify and update Significant Digits and Data Types for ADaM dataset variables for the P21 specification. P21-identifies an issue if the digits are missing and the numeric variable datatype is float. We need to find these issues and correct them manually for each variable after importing the specifications to P21. This process can be time consuming and requires a manual update in the specifications.

This paper will explain how we can programmatically update the P21 ADaM specification for Significant Digits and Data type using a SAS® macro. This macro will update Significant Digits and Data type for columns variables, value level and code lists using a data driven approach and provides an efficient solution towards producing a high quality Define.xml.

INTRODUCTION

Define.xml is an essential part of a New Drug Application (NDA) submission package submitted to regulatory agencies like the Food and Drug Administration (FDA). It is used to describe CDISC SDTM and ADaM datasets for submissions to the Regulatory Agencies. When included in an NDA submission package, this document increases the level of automation and improves the efficiency of the Regulatory Review process, making it highly desirable with every NDA submission. The Define.xml Generator is based on an input ADaM Specification file created using Excel, allowing a user to focus more on the metadata content of the study rather than a complex XML syntax. The Define.xml generator of the Pinnacle 21 community software performs two functions:

- Create a Pinnacle 21 format spec
- Generate a Define.xml file using a Pinnacle 21 format spec

A Define.xml contains the metadata information of the submission datasets and its variables. Different categories of the metadata information are organized as individual tabs in the Pinnacle 21 format specification. This information is categorized into the following ten categories

1. Study
2. Datasets
3. Variables
4. Value Level Metadata
5. Where Clauses for Parameter Value Level Metadata
6. Codelists
7. Dictionaries
8. Methods
9. Comments
10. Documents

DATA TYPE CONSIDERATIONS

CDISC SDTM, SEND and ADaM variable data types are specified as either character ("Char" and "Num", respectively) however, the define.xml based on the CDISC Operational Data Model (ODM) supports a richer set of data types. The following table illustrates the mappings between Define.xml and ADaM data types:

Define-XML Data Type	Submission Data Type	Length	Considerations
text	Char	Maximum Allowable length	SAS Version 5 transport files restrict variable lengths to 200 characters.



integer	Num	The largest allowable integer width	Use for numeric or equivalent variables that have discrete whole values (non-fractional). Can be positive, negative, or zero. ADaM date variables, are provided as integers.
float	Num	The largest allowable whole number width plus the maximum number of decimal digits.	Use for numeric variables that may contain a fractional component. It represents the set of all the decimal numbers with arbitrary lengths.
datetime	Char	N/A	Use if values for SDTM or Send variable represent Date Times (YYYYMM-DDTHH:MM:SS.SS)
date	Char	N/A	Use if values for SDTM or SEND variable represent complete (YYYYMM-DD) dates

Significant Digits are the number of digits following the decimal point in a floating-point number and required when Data type is “float”.

All Numeric Data are assigned as float by default when importing an ADaM specification into Pinnacle 21. P21 identifies an issue if the digits are missing and when the datatype is float. The Data type for numeric variables needs to be identified as integer or float and the Significant Digits for float Variables needs to be specified and updated as per ADAM datasets.

Properties

Datasets

Variables

Value Level

Codelists

Terms

Methods

Comments

Analysis Displays

Analysis Results

Issues 11

Q Search

Order ▾	Dataset ▾	Variable ▾	Label ▾	Type ▾	Length ▾	Digits ▾	Format ▾
38	ADAE ▾	ADURN	Analysis Duration (N)	float ▾	8		
84	ADAE ▾	AEDURDD	Duration of Adverse Event Diff of Dates	float ▾	8		
84	ADAECM ▾	PEQDCF	Prednisone Equivalent Dose Conv. Factor	float ▾	8		
85	ADAECM ▾	PEQD	Prednisone Equivalent Dose (mg/day)	float ▾	8		
48	ADCM ▾	CMDOSE	Dose per Administration	float ▾	8		

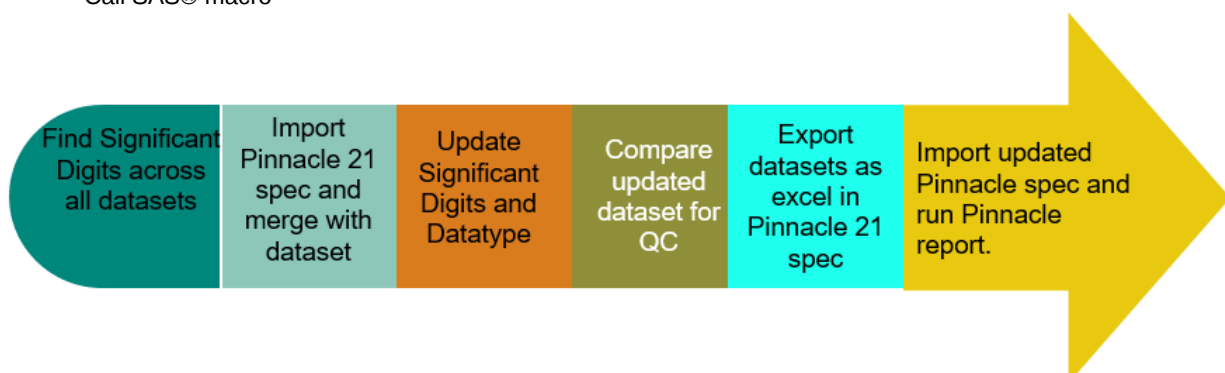
Display 1. After importing ADAM specification into P21

After importing the specifications to P21, if any issues are discovered they need to be corrected manually in P21 Specifications. Correcting these issues manually and addressing them in the specification would be very time consuming, tedious and attention-demanding.

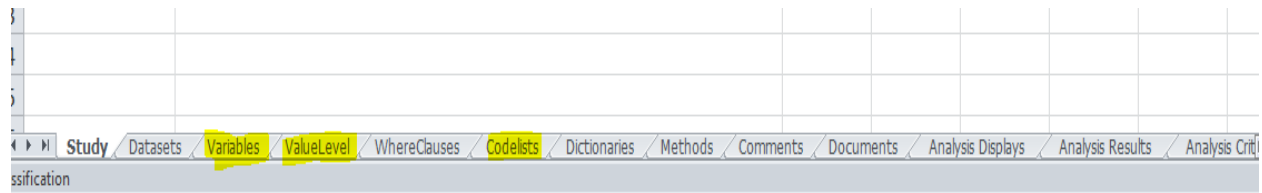
Creating a SAS® macro to dynamically update Significant Digits and Data type for columns variables, value level and code lists in P21 Specification provides an efficient solution towards producing a high quality Define.xml in an efficient manner.

PROCESS FLOW

- Import ADAM spec and run Pinnacle report.
- Export Pinnacle spec
- Call SAS® macro



Display 2. Process flow for SAS® macro



Display 3. Pinnacle 21 Specification

Highlighted tabs ('Variable' 'ValueLevel' 'Codelists') will be dynamically updated by the SAS® macro for Significant Digits and Data type in Pinnacle Specification.

E	F	G
Data Type	Length	Significant Digit
float	8	4
float	8	4
float	8	2
float	8	2

Display 4. Example of the updated Significant Digits for highlighted tab ('Variable' 'ValueLevel' 'Codelists').

SAS MACRO FUNCTION

SAS script	Function
<pre> * list number of datasets involved; select distinct memname into :domains separated by '~' from dictionary.tables where libname = "ADAM" ; %let cnt = &sqllobs; *below step needs to be repeated for each numeric variable across all datasets; *list of all numeric variables across all datasets; proc contents data=ADAM.&single. out=&single._cont(where=(type=1)) noprint ; run; select distinct name into: num_vars separated by "#" from &single._cont; %let cnt_vars = &sqllobs; *find Significant Digit; data &domain._&domain_var; set &domain._&domain_var; var= strip(put(&domain_var, best.)); call scan (var,-1,p,length); d = findC(var,','T',-length(var)); if d then dec=length(var)-d; else dec=0; if dec>0; run; *select maximum Significant Digit; create table sig_dig_&domain._&domain_var as select distinct domain, variable, max(dec) as sig_digits from &domain._&domain_var; </pre>	Find out Significant Digits across all datasets and select maximum Digit for each variable. Create the final dataset which contains the maximum Significant Digits for each dataset based on each numeric variable.



<pre>Proc import datafile = "&path\&file_name.xlsx" out=spec_vars dbms= xlsx replace; sheet = '&sheet_name'; getnames=yes; run; *Create macro variable to maintain order in columns for P21 Specification during proc export procedure; proc contents data= spec_vars out=spec_var_temp ; run; proc sort data= spec_var_temp; by varnum; run; proc sql noprint; select name into:var separated by ' ' from spec_var_temp order by varnum; quit; %put &var.; proc sql; create table test as select a.* , b.domain, b.sig_digits from spec_vars as a left join __digit as b on a.dataset=b.domain and a.variable=b.variable; quit; *update ignificant digits for float variables and assigned Data type as integer for all remaining numeric variables; *retain same order of column as in spec; data test; retain &var.; set test; run;</pre>	Importing Pinnacle 21 Specification and merge with above dataset. Update the Significant Digits and Data type (integer VS float).
<pre>proc compare base=Pinnacle_spec_sheet compare=Updated_pinnacle_spec_sheet listall; run;</pre>	Review the PROC Compare report (P21 Specification sheet vs updated sheet) to make sure only the required changes have been done in the P21 Specification for Significant Digits and Data type.
<pre>proc export data=&dataset_name outfile="&path\&file_name.xlsx" label dbms=xlsx replace; sheet="&sheet_name"; run;</pre>	Export the SAS dataset as a Pinnacle 21 Specification in Excel format with same Sheet name.

Additional Steps for P21 Specification sheet(ValueLevel)

<pre>%do i = 1 %to &max.; proc sql noprint; create table __aval&i. as select strip(put(max(length(scan(strip(put(&&var&i,best.))- 1,'.'))),best.)) as aval, &i. as max from ADAM.&&dataset&i where paramcd in (&&paramcd&i) and index(strip(put(&&var&i,best.)),'.') > 0;</pre>	Find significant digits across domains for each variable based on paramcd mentioned in the method column of the ValueLevel sheet.
--	--



quit; %end;	
----------------	--

Note This process will be performed for three Pinnacle 21 Specification sheets('Variable' 'ValueLevel' 'Codelists').

CONCLUSION

This SAS® macro can dynamically update the Significant Digits and Data Type in a Pinnacle 21 Specification using a data drive approach and helps to avoid manual updates in the specification which saves significant time in creating a high quality Define.xml for Regulatory submission.

REFERENCES

1. CDISC Define-XML Team, "CDISC Define-XML Specification." CDISC, Version 2.0, 24 Apr. 2014, CDISC. Available at <http://www.cdisc.org/define-xml>
2. "Generating Define.xml from Pinnacle 21 Community" Available at <https://www.pharmasug.org/proceedings/2018/AD/PharmaSUG-2018-AD29.pdf>

ACKNOWLEDGEMENTS

Author would like to thank Sapan Patel for his valuable feedback and suggestions.

CONTACT INFORMATION

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

Ghanshyam Jangid
Merck & Co., Inc. USA
351 N Sumneytown Pike
North Wales, PA 19454
Work Phone: 267-305-2910
Email: Ghanshyam.jangid@merck.com