PharmaSUG China 2023 - Paper DS-126

Automation of SDTM Define-XML

Jundong Ma, Dizal Pharma, Shanghai, China

ABSTRACT

In this paper, I will introduce a SAS macro-based tool to extract metadata from Architect Loader Spreadsheet (ALS), annotated Case Report Form (aCRF), and SDTM datasets to generate the SDTM Define.xlsx, which can be directly transferred to SDTM Define.xml and validated by Pinnacle 21 (P21). The tool supports both Define-XML version 2.0 or 2.1 in English or Chinese. In addition, the Define.xlsx can also be used to update the SDTM datasets from SDTM-like datasets with efficiency and traceability.

Keywords: CDISC, SDTM, Define-XML, define.xml, metadata

INTRODUCTION

Define.xml is an integral component of CDISC (Clinical Data Interchange Standards Consortium) standards as it defines variables used in a specific dataset along with their characteristics such as labels, formats, lengths etc. It serves as a metadata repository that facilitates accurate interpretation and submission of clinical trial data to regulatory authorities.

Pinnacle 21 Community is a free and user-friendly open source toolkit by Pinnacle 21. It is commonly used by SPs (statistical programmers) to validate SDTM and ADaM datasets. In addition to CDISC data conformance validation, Pinnacle 21 Community also features other tools such as the Define.xml generator. The function is to generate a Define.xml file using a Pinnacle 21 format spec. Pinnacle 21 format spec can be transferred to Define.xml, in addition, it can also help SPs to review the SDTM metadata and even be used for metadata-driven automation.

Traditionally, creating a Pinnacle 21 format spec involved labor-intensive manual processes.

For example, as illustrated in Figure 1, to generate the Define.xml, a programmer starts with the manual generated SDTM Spec, and transforms it either manually or by macro to Pinnacle 21 compliant format of Define.xlsx, then use Pinnacle 21 to transform it to Define.xml.

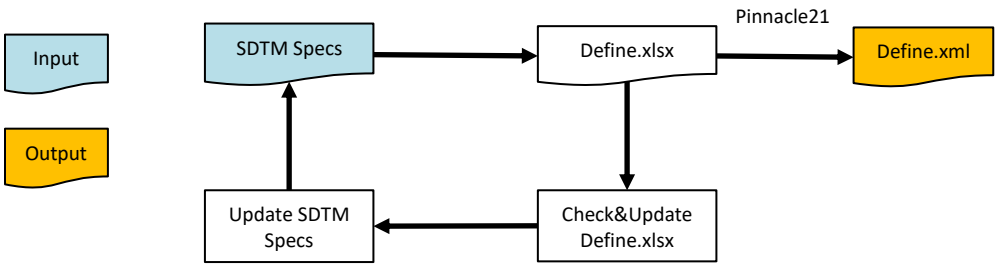


Figure 1 Traditional Process to Generate Define.xml

In this process, manually created SDTM Specs is required. Currently, there's no industry standard template available, each company have their own templates of the Spec. SPs need to manually check and update the SDTM Specs and the Define.xlsx if any mistakes.

The process has several drawbacks:

1. For outsourced or legacy studies, there’s no SDTM Specs available.
2. Additional efforts required to translate the SDTM Specs for submission to NMPA.

3. Additional efforts required to manually update methods and comments, since SDTM Specs are designed to guide statistical programmers instead for HA review.
4. Potential discrepancies between SDTM Specs and SDTM datasets, which causes discrepancies between Define.xml and SDTM datasets.
5. Additional efforts required for multiple submissions.

To address these challenges, automating the Define generation process has emerged as an attractive solution.

METADATA DRIVEN FLOW

As illustrated in Figure 2, this approach starts with using SDTM datasets or SDTM-like datasets as input to get SDTM metadata. Powered by Industry standards and internal standards, the SDTM metadata together with other study specific metadata drives the automation of Define.xlsx.

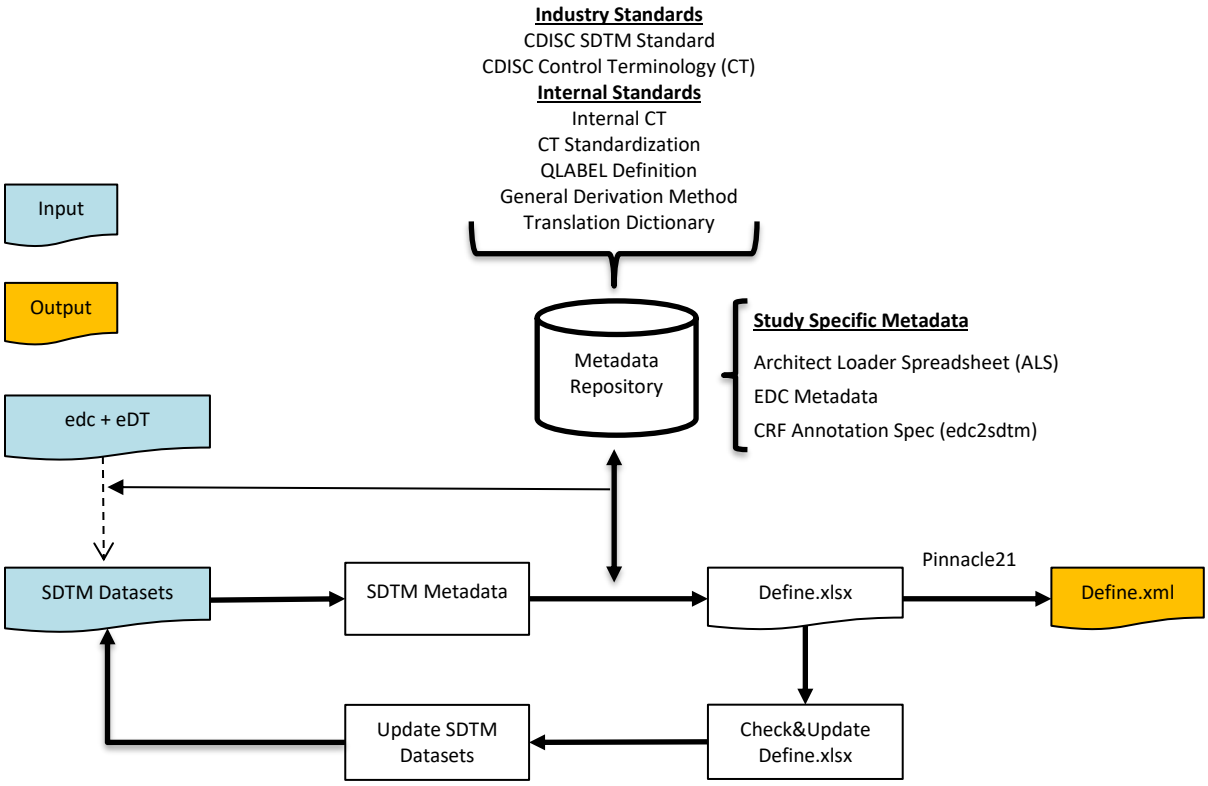


Figure 2 Metadata Driven Process to Generate Define.xml

Nowadays, SDTM mapping becomes more and more easy with the development of standards and metadata repository. Based on structural and transformational metadata managed in a repository, metadata-driven data flow can automatically generate SDTM-like datasets.

In Dizal, we have an automatic process for CRF annotation, which starts from ALS (from RAVE system), and a blank CRF, and a CRF Annotation Spec (edc2sdtm) template to generate a study level edc2sdtm.xlsx (to be updated manually for the study specific CRF pages), which contains all the information for CRF annotation. With the metadata from ALS, edc2sdtm.xlsx, edc data, eDT data and study protocol, the information of all the source data is available. Powered by industry standards and internal standards, we can extract metadata of source data, and generate the SDTM-like datasets, which will be used as a starting point of the SDTM define-xml process. First step of the process is to generate

the SDTM metadata based on SDTM datasets. Then the SDTM metadata are used to generate a draft Define.xlsx, which can be transformed to Define.xml and validated by Pinnacle 21. Besides, there're some build-in checks to highlight potential issues in the Define.xlsx. Then the Define.xlsx can be further used to modify the SDTM datasets in reverse. If any updates happened to the SDTM datasets, SPs need to regenerate the define.xlsx based on the updated SDTM datasets.

The process has several advantages compared with traditional process:

1. All steps are almost automated: generation, translation, up-versioning of SDTM CT, up-versioning of define.xml, build-in checks, update SDTM.
2. Always consistent between SDTM datasets and the SDTM-Define.
3. Always compliant to Pinnacle 21.
4. Applicable for outsourced or legacy studies.

GENERATE SDTM METADATA

A SAS macro was developed to get SDTM metadata. The macro use SDTM datasets as inputs and outputs metadata related to dataset structures, variable attributes, value level metadata, and all unique values.

Macro parameters of macro %m_sdtm_read_sdtm:

- **lib_sdtm** (Required):
  libname of SDTM datasets.
- **valuelevel_modify** (Not Required):
  Update categories for valuelevel. For example, if you want to remove LBORRESU and LBSTRESU from valuelevel and update categories for LBORRES and LBSTRESC. The parameter valuelevel_modify should be set as "LBORRESU: | LBSTRESU: | LBORRES: LBSPEC LBTESTCD | LBSTRESC: LBSPEC LBTESTCD".
  Default categories:
  - XXORRES/XXORRESU/XXSTRESC/XXSTRESU: XXNAM + XXTESTCD (except LB)
  - LBORRES/LBORRESU/LBSTRESC/LBSTRESU: LBNAM + LBCAT + LBTESTCD
  - TSVAL: TSPARMCD
  - SUPPXX.QVAL: QNAM + QORIG
- **force_run** (Not Required):
  In order to reduce running time, hashing_file function is used to detect any updates to the input files. No update, no rerun. In order to rerun when no updates to input files, macro parameter FORCE_RUN should be set as Y to force rerun. This is useful when the macro variable valuelevel_modify is updated but the SDTM datasets are the same.

The SDTM metadata are archived in the study metadata library as four SAS datasets. Here're the screenshots of these datasets.

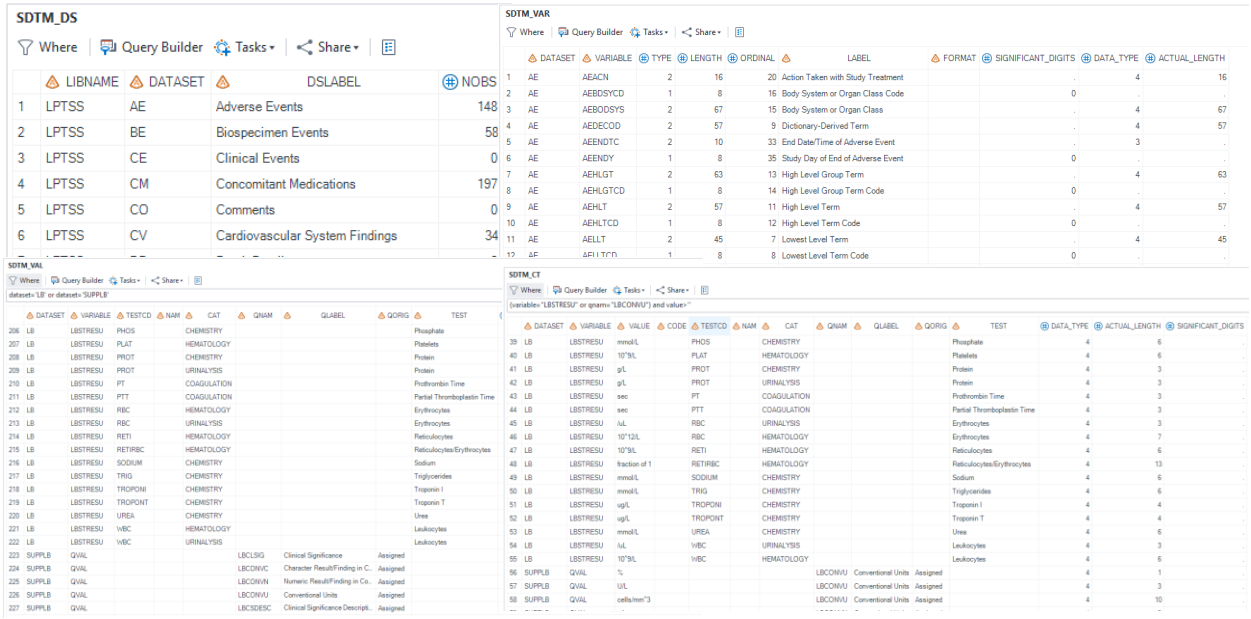


Figure 3 SDTM Metadata

GENERATE DEFINE.XLSX

Several SAS macros were developed to use SDTM metadata (required) and als (not required), edc2sdtm (not required) as inputs and output a Define,xlsx to be transferred to Define.xml by Pinnacle 21.

The tool supports both English and Chinese, based on macro parameter &language, which should be assigned in study level autoexec.sas. If not assigned, English is set as default.

Macro Parameter	Required?	Valid Entry	Default	Usage
SDTMIG_VERSION	Y	3.2 | 3.3	3.2	Specify SDTM IG Version.
SDTMCT_VERSION	Y	20230331		Specify SDTM CT Version. Default is the latest version.
DEFINE_VERSION	Y	2.0 | 2.1	2.0	Specify Define Version.
SDTM_DEFINE	Y		&protpath/docs/sdtm/define.xlsx	Specify path of define.xlsx as both input and output.
SDTM_SPEC	N		&protpath/docs/sdtm/sdtm-spec.xlsx	Specify path of legacy sdtm-spec.xlsx if available. Use info from sdtm-spec to updata define.xlsx
READ_ORIGINAL_DEFINE	Y	Y | N	Y	Specify whether to read orginal define.xlsx as input. Y to read, N to neglect.
HELP	N	Y | N	N	Put help information in log?

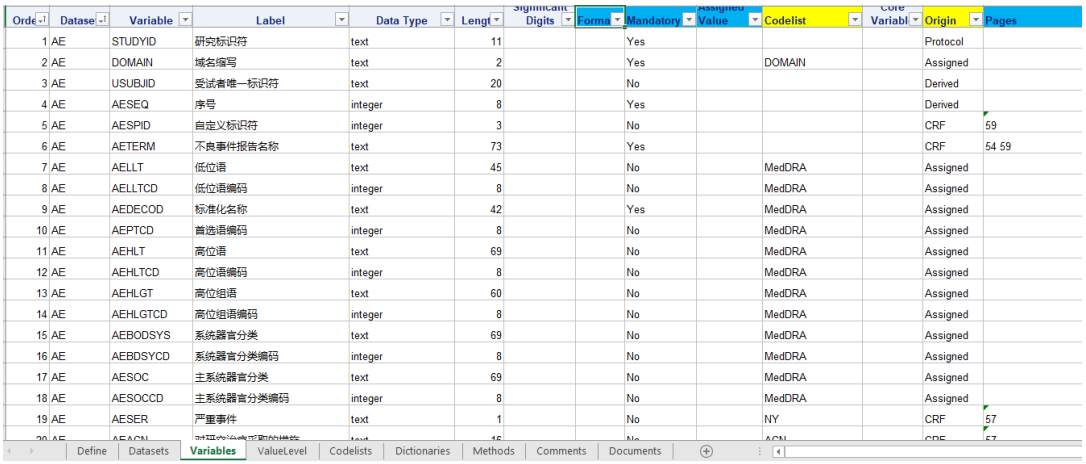
Table 1. Macro parameters of macro m_sdtm_define

Constraints:

- SDTM datasets or SDTM-like datasets must be ready before using the tool.
- Rerun when SDTM datasets are updated.
- edc2sdtm.xlsx and sdtm-spec.xlsx must be conform to Dizal standards if provided.

An example of the macro calls is as below.

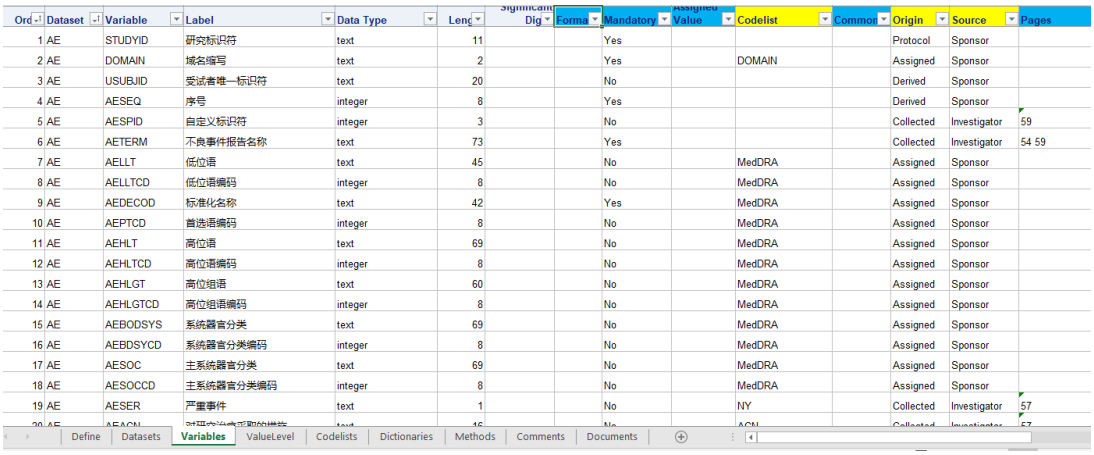
```
%*Generate/Update define.xlsx;
%m_sdtm_define(
    sdtmig_version=3.2,
    define_version=2.0,
    sdtmct_version=20230331,
    sdtm_define=&protpath/docs/sdtm/define.xlsx,
    read_original_define=Y);
```



Orde	Dataset	Variable	Label	Data Type	Length	Significant Digits	Format	Mandatory	Assigned Value	Codelist	Core Variable	Origin	Pages
1	AE	STUDYID	研究标识符	text	11			Yes				Protocol	
2	AE	DOMAIN	域名缩写	text	2			Yes		DOMAIN		Assigned	
3	AE	USUBJID	受试者唯一标识符	text	20			No				Derived	
4	AE	AESEQ	序号	integer	8			Yes				Derived	
5	AE	AESPID	自定义标识符	integer	3			No				CRF	59
6	AE	AETERM	不良事件报告名称	text	73			Yes				CRF	54 59
7	AE	AELLT	低位语	text	45			No		MedDRA		Assigned	
8	AE	AELLTCD	低位语编码	integer	8			No		MedDRA		Assigned	
9	AE	AEDECOD	标准化名称	text	42			Yes		MedDRA		Assigned	
10	AE	AEPTCD	首选语编码	integer	8			No		MedDRA		Assigned	
11	AE	AEHLT	高位语	text	69			No		MedDRA		Assigned	
12	AE	AEHLTCD	高位语编码	integer	8			No		MedDRA		Assigned	
13	AE	AEHLGT	高位组语	text	60			No		MedDRA		Assigned	
14	AE	AEHLGTCD	高位组语编码	integer	8			No		MedDRA		Assigned	
15	AE	AEBODSYS	系统器官分类	text	69			No		MedDRA		Assigned	
16	AE	AEBDSYCD	系统器官分类编码	integer	8			No		MedDRA		Assigned	
17	AE	AESOC	主系统器官分类	text	69			No		MedDRA		Assigned	
18	AE	AESOCCD	主系统器官分类编码	integer	8			No		MedDRA		Assigned	
19	AE	AESER	严重事件	text	1			No		NY		CRF	57

Define | Datasets | Variables | ValueLevel | Codelists | Dictionaries | Methods | Comments | Documents

Figure 4 Define.xlsx (define2.0)



Orde	Dataset	Variable	Label	Data Type	Length	Significant Digits	Format	Mandatory	Assigned Value	Codelist	Common	Origin	Source	Pages
1	AE	STUDYID	研究标识符	text	11			Yes				Protocol	Sponsor	
2	AE	DOMAIN	域名缩写	text	2			Yes		DOMAIN		Assigned	Sponsor	
3	AE	USUBJID	受试者唯一标识符	text	20			No				Derived	Sponsor	
4	AE	AESEQ	序号	integer	8			Yes				Derived	Sponsor	
5	AE	AESPID	自定义标识符	integer	3			No				Collected	Investigator	59
6	AE	AETERM	不良事件报告名称	text	73			Yes				Collected	Investigator	54 59
7	AE	AELLT	低位语	text	45			No		MedDRA		Assigned	Sponsor	
8	AE	AELLTCD	低位语编码	integer	8			No		MedDRA		Assigned	Sponsor	
9	AE	AEDECOD	标准化名称	text	42			Yes		MedDRA		Assigned	Sponsor	
10	AE	AEPTCD	首选语编码	integer	8			No		MedDRA		Assigned	Sponsor	
11	AE	AEHLT	高位语	text	69			No		MedDRA		Assigned	Sponsor	
12	AE	AEHLTCD	高位语编码	integer	8			No		MedDRA		Assigned	Sponsor	
13	AE	AEHLGT	高位组语	text	60			No		MedDRA		Assigned	Sponsor	
14	AE	AEHLGTCD	高位组语编码	integer	8			No		MedDRA		Assigned	Sponsor	
15	AE	AEBODSYS	系统器官分类	text	69			No		MedDRA		Assigned	Sponsor	
16	AE	AEBDSYCD	系统器官分类编码	integer	8			No		MedDRA		Assigned	Sponsor	
17	AE	AESOC	主系统器官分类	text	69			No		MedDRA		Assigned	Sponsor	
18	AE	AESOCCD	主系统器官分类编码	integer	8			No		MedDRA		Assigned	Sponsor	
19	AE	AESER	严重事件	text	1			No		NY		Collected	Investigator	57

Define | Datasets | Variables | ValueLevel | Codelists | Dictionaries | Methods | Comments | Documents

Figure 5 Define.xlsx (define2.1)

Both define-xml versions 2.0 and 2.1 are supported since both versions are used in Dizal. Upgrading of the define-xml version can be done automatically by assigning the macro parameter define_version = 2.1.

There're several difference between the two versions [1]:

- Multiple standards and controlled terminologies can be associated
- Origin enhancements
- Subclass and HasNoData flag are added

Different color of column header is to guide SPs about whether the columns can be updated manually or not.

- Grey: Based on data or study files. Cannot be updated manually.
- Blue: Normally has a source but can be updated manually.
- Yellow: Current rule is not stable. Need to be checked and updated manually.

BUILD-IN CHECKS AND USE DEFINE.XLSX TO UPDATE SDTM

There're several build-in checks to find issues in the SDTM datasets and highlighted in the Define.xlsx:

1. Dataset label, variable label, and QLABEL in SUPPQUAL are too long or not consistent with CDISC or internal standard.
2. Duplicate records group by the Key Variables specified in Define.xlsx.
3. Non-standard Term in Codelists sheet.
4. Duplicate --STRESU group by --TESTCD.
5. Multiple code-lists with the same name.
6. Variables in the "Where Clause" of "ValueLevel" sheet not found in the "Variables" sheet.
7. Filter non-standard CT.

Most of these issues cannot be checked by current Pinnacle 21. SPs can easily identify these issues by reviewing the Define.xlsx. Further, some issues can be updated in the Define.xlsx, which will be used to update the SDTM datasets:

1. Update labels of datasets, variables, or QLABEL in SUPPQUAL.
2. Use Key Variables to derive –-SEQ.
3. Add or delete variables.
4. Executable SAS codes in the "Developer Notes" with running sequence and comments.

Once the SDTM datasets are final, the Define.xlsx should be re-generated based on the final SDTM datasets.

STEP 4: GENERATE DEFINE.XML

Once the Define.xlsx is ready, it's very easy to transform it to Define.xml by Pinnacle 21 Community.

To generate the Define.xml:

1. Visit https://www.pinnacle21.com/downloads, download the latest release of Pinnacle 21® Community and install it.
2. Select Define.xml
3. Generate Define from navigation menu
4. Choose the define standard (2.0 or 2.1)
5. Browse to your Define.xlsx
6. Click Generate to start the generation process.
7. Once complete, click Open Define.xml button to open and view the completed Define.xml in a web browser

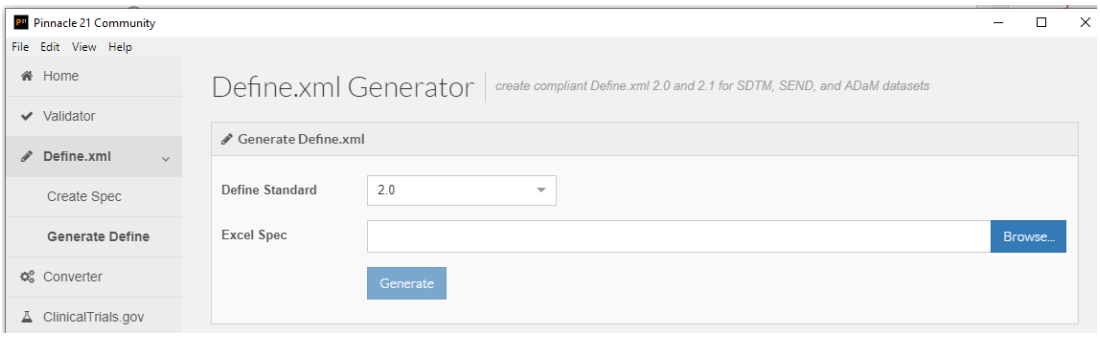


FIGURE 2 GENERATE DEFINE.XML SCREEN

CONCLUSION

By using an automation tool to generate the Define.xlsx, now the SDTM Define.xml is not only used to facilitate review by regulatory authorities, but also used by statistical programmers to monitor the SDTM metadata and improve the quality and efficiency of SDTM datasets.
It's a further step toward SDTM automation.

REFERENCES

[1] Trevor Mankus (2022). "Upgrading from Define-XML 2.0 to 2.1". Available at https://www.lexjansen.com/pharmasug/2022/DS/PharmaSUG-2022-DS-064.pdf

ACKNOWLEDGMENTS

The authors would like to thank the statistical programmer team from Dizal Pharma, for their great support and valuable inputs into this paper.

CONTACT INFORMATION

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

Jundong Ma

Dizal Pharma

E-mail: jundong.ma@dizalpharma.com