

A macro to automate the P21E define excel specification for ADaMs

Daoqing Li, Dizal Pharma

ABSTRACT

The define.xml is a mandatory document for data submission to regulatory agencies. However, manual development of the define in P21E can be time-consuming and error-prone. This paper introduces a macro (M_ADAM_DEFINE) that automates the generation of P21E define excel specifications for ADaMs, which can be directly imported into P21E to create define.xml. The macro integrates various input sources, including individual ADaM specifications, ADaM xpts, ADaM-IG metadata, SDTM-IG metadata, company level Control Terminology and even corresponding SDTM P21E define excel specification. It then transforms them into a single P21E define excel specification. Consistency checks are conducted among these sources, and error messages will be generated if any issues are detected. The paper elucidates the parameters, constraints, and output files of the macro while providing illustrative examples of its usage and output. This macro can expedite the generation and validation of define.xml for ADaM datasets, thereby enhancing data submission efficiency and quality. Future research may explore the applicability of this macro to other dataset types such as SDTM, as well as its compatibility with alternative tools or platforms like SAS Clinical Standards Toolkit.

INTRODUCTION

Define.xml is a required piece for all electronic clinical trial data submissions that is designed to describe the contents and structures of SDTM, ADaM and even Analysis Results Metadata (ARM). It is a key instrument for the regulatory agency reviewers to understand how the data is organized and analyzed. CDISC published define-xml v2.0/v2.1 specifications to guide the development and provided a style sheet that can display define.xml as a formatted, browser-based and human-readable report. The excellent quality of the define.xml is a “business card” for the regulatory agencies.

So how to create a define.xml? It is based on the XML language, but it is not really easy to write it using an XML script. Currently, Pinnacle 21 Enterprise (P21E), a well-known and widely used platform to prepare clinical trial data submission packages among industries, offers multiple ways, such as scanning uploaded data (ADaM.xpt), copying from reference study, using an excel specification in P21E format, etc. However, most of the methods still require a lot of manual work in P21E interface, including but not limited to specifying origin of variables, adding the comments or the methods for variable derivation, cleaning the Control Terminology. Typically, there are dozens of datasets and dozens of variables for each dataset. It is time-consuming and error-prone to populate these variables one by one.

We noticed that there are many similarities between P21E define excel specification and ADaM specifications. Both contain the list of variables, label for each variable, and most importantly, the derivation of variables. Why not design ADaM specifications to be closer to P21E excel specification? Based on this idea, we came up with an ADaM specification format containing most of the information that define.xml needed, and the design does not increase much extra work in ADaM specifications development during CSR preparation compared to traditional ADaM specifications.

A SAS macro named M_ADAM_DEFINE is developed to integrate all the individual ADaM specification into a single P21E define excel specification, which can be imported into P21E to generate a completed define.xml. There are several additional sources, for example, ADaM xpts, ADaM-IG metadata, SDTM-IG metadata, company level Control Terminology, and even corresponding SDTM P21E define excel specification. ADaM specifications provide the primary information that is essential for define.xml. ADaM xpts are read to provide variable attributes, including label, length, data type, and more. Metadata contribute to improving the compliance to IG of define.xml. If a SDTM variable is the predecessor for a certain ADaM variable, SDTM P21E define excel specification will be used to carry information to ADaM define, such as control terminology. M_ADAM_DEFINE performs a lot of consistency checks between ADaM specifications, xpts, IG metadata, Control Terminology. These checks will highly improve the

quality of define.xml, compliance to define specification and even detect potential issues in ADaM datasets.

PINNACLE 21 ENTERPRISE DEFINE EXCEL SPECIFICATION

Before reading this paper, it is strongly recommended to read the guideline '[CDISC Define-XML Specification Version 2.1](#)'. This guideline will give you a comprehensive overview and help you understand this paper better.

Following illustrate the 11 tabs and their columns in P21E define excel specification, which are also called 11 metadata.

- Define
 - Attribute, Value
 - Include the details about study name, study description, protocol name.
- Datasets
 - Dataset, Label, Class, Structure, Key Variables, Repeating, Reference Data, Comment, Developer Notes
- Variables
 - Order, Dataset, Variable, Label, Key Variables, Data Type, Length, Significant Digits, Format, Mandatory, Assigned Value, Codelist, Core Variable, Origin, Pages, Method, Predecessor, Role, Comment, Developer Notes
- ValueLevel
 - Same as Variables but add one additional column Where Clause.
- Codelists
 - ID, Name, NCI Codelist Code, Data Type, Order, Term, NCI Term Code, Decoded Value
- Dictionaries
 - ID, Name, Data Type, Dictionary, Version
- Methods
 - ID, Name, Type, Description, Expression Context, Expression Code, Document, Pages
- Comments
 - ID, Description, Document, Pages
- Documents
 - ID, Title, Href
- Analysis Displays
 - ID, Title, Document, Pages
- Analysis Results
 - Display, ID, Description, Variables, Reason, Purpose, Selection Criteria, Join Comment, Documentation, Documentation Refs, Programming Context, Programming Code, Programming Document, Pages

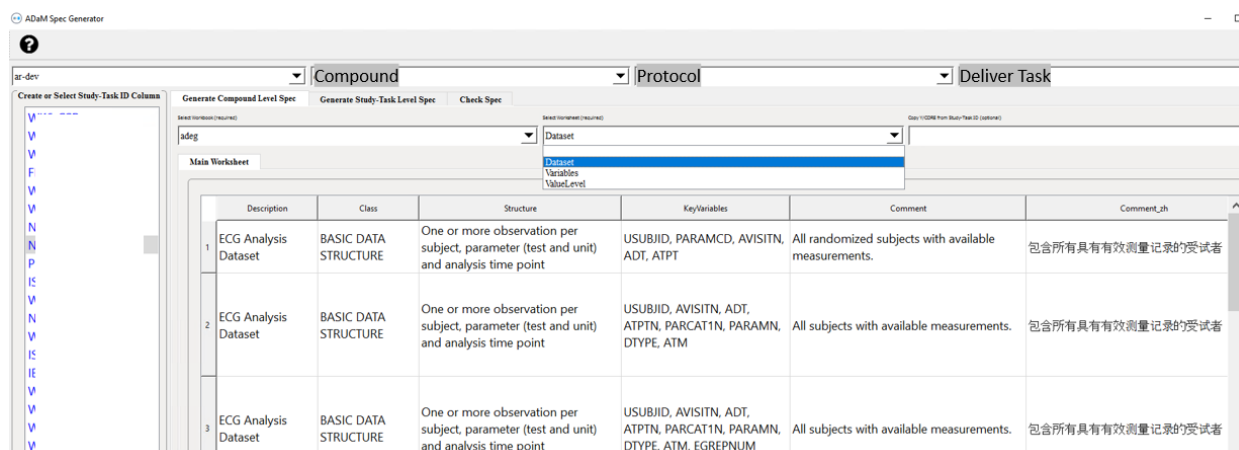
ADAM SPECIFICATION

At Dizal, the designed ADaM specification is a spreadsheet that can be easily manipulated and maintained by a user-friendly App called 'ADaM Spec Generator' created by Python. For further details about the App, please refer to the article [“ADaM Specifications Automation from Company Standards to Study”](#). The specification has three tabs (Dataset, Variables and ValueLevel), which have similar structures to P21E excel define specification. The following is an example of ADEG.

DATASET

The Dataset tab contains the following columns: Dataset, Description, Class, Structure, Purpose, KeyVariables, Comment, Document, Pages. For the Chinese specification, the Comment_zh column in the App is valid to generate Comment.

The Dataset tab accounts for creating the Datasets in the P21E define excel specification. Please note that the Class column should follow CDISC define-xml control terminology. Typically, there are four classes used for ADaM define, i.e., SUBJECT LEVEL ANALYSIS DATASET, OCCURRENCE DATA STRUCTURE, BASIC DATA STRUCTURE, ADAM OTHER.



	Description	Class	Structure	KeyVariables	Comment	Comment_zh
1	ECG Analysis Dataset	BASIC DATA STRUCTURE	One or more observation per subject, parameter (test and unit) and analysis time point	USUBJID, PARAMCD, AVISITN, ADT, ATPT	All randomized subjects with available measurements.	包含所有具有有效测量记录的受试者
2	ECG Analysis Dataset	BASIC DATA STRUCTURE	One or more observation per subject, parameter (test and unit) and analysis time point	USUBJID, AVISITN, ADT, ATPTN, PARCAT1N, PARAMN, DTYPE, ATM	All subjects with available measurements.	包含所有具有有效测量记录的受试者
3	ECG Analysis Dataset	BASIC DATA STRUCTURE	One or more observation per subject, parameter (test and unit) and analysis time point	USUBJID, AVISITN, ADT, ATPTN, PARCAT1N, PARAMN, DTYPE, ATM, EGREPNUM	All subjects with available measurements.	包含所有具有有效测量记录的受试者

Display 1. Dataset tab in ADaM Spec Generator

VARIABLES

The Variables tab contains the following columns: Dataset, Variable, Label, Data Type, Length, Format, Codelist, Origin, Comment, Document, Pages. The Comment column include the derivation rules for each variable. The Origin, Document and Pages columns are not relevant for the ADaM development but facilitate define generation.

ADaM Spec Generator

Compound Protocol Deliver Task

Create or Select Study-Task ID Column

Generate Compound Level Spec Generate Study-Task Level Spec Check Spec

Send Worksheet (required)

adeg Variables

Main Worksheet

Variable	Label	Data Type	Length	Format	Codelist	Origin	Comment	Comment_zh
43 PARAM	Parameter	text	200		ADEG_PARAM	Assigned	ADEG.PARAM=EG.EGTEST concatenated with EG.EGSTRESU for original records; ADEG.PARAM=EG.EGTEST concatenated with EG.EGSTRESU and 'Average Value' for averaged records.	基于ADEG.PARAMCD受控术语列表型赋值
44 PARAMCD	Parameter Code	text	8		ADEG_PARAMCD	Predecessor	EG.EGTESTCD	EG.EGTESTCD
45 PARAMCD	Parameter Code	text	8		ADEG_PARAMCD	Derived	EG.EGTESTCD; ADEG.PARAMCD=EG.EGTESTCD concatenated with 'AV' for averaged records.	EG.EGTESTCD
46 PARAMCD	Parameter Code	text	8		ADEG_PARAMCD	Predecessor	EG.EGTESTCD; ADEG.PARAMCD=EG.EGTESTCD concatenated with 'AV' for averaged records.	各研究源数据ADEG.PARAMCD
47 PARAMN	Parameter (N)	integer	8		ADEG_PARAMN	Assigned	The numeric value for ADEG.PARAMCD based on Codelist	基于ADEG.PARAM受控术语列表的赋值

Display 2. Variables tab in ADaM Spec Generator

The Codelist column in the ADaM specification is linked to company-level control terminology, which combines the CDISC control terminology and company data standards and is maintained regularly. Users can also customize the Codelist according to the study design by making minor modifications to the current company-level control terminology or adding a new codelist. The Codelist column is used to generate Codelists tab in the P21E define excel specification.

Generate Compound Level Spec Generate Study-Task Level Spec Check Spec

Send Worksheet (required)

adeg Variables

Main Worksheet Codelist

	ADaMName	Non_Extensible	ID	Name	Name_zh	NCI Codelist Code	Data Type	Term	Term_zh	NCI Term Code
1	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	EGVRMN	EGVRMN	ECG N
2	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	EGHRMN	EGHRMN	ECG N
3	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	PRAG	PRAG	PR Int
4	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	RRAG	RRAG	RR Int
5	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	QRSAG	QRSAG	QRS I
6	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	QTAG	QTAG	QT Int
7	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	QTCBAG	QTCBAG	QTCB
8	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	QTCFAG	QTCFAG	QTCF I
9	ADEG		ADEG_PARAMCD	Parameters CD of EG Tests	ECG参数简称		text	INTP	INTP	Interp

Display 3. Setting of codelists term in ADaM Spec Generator

Moreover, if Origin is 'Predecessor' and the predecessor variable comes from SDTM, the SDTM P21E define excel specification will be imported, the Codelist and terms for the corresponding variable will be transferred to the ADaM define.

VALUELEVEL

The ValueLevel tab has two extra columns compared to the Variables tab: 'Where Variable' and JoinID. This tab is designed based on the ValueLevel tab in the P21E define excel specification. User can add where clause variables and JoinID to set filtering of datasets. In the Display 4, the where clause PARAMCD EQ 'QTCFAG' is applied to get a subset of ADEG. Any variable in the corresponding ADaM, except the one in Variable column, can be used as a where clause variable. For JoinID, the following values are valid: EQ, NE, IN, NOTIN, LT, LE, GT, GE.

ADaM Spec Generator

ar-dev

Compound Protocol Delivery Task

Create or Select Study-Task ID Column

Generate Compound Level Spec Generate Study-Task Level Spec Check Spec

Select Workbook (required) adag ValueLevel

Select Worksheet (required) Copy Y (CORE from Study-Task ID (optional))

Main Worksheet

	Dataset	PARAMCD	Variable	Data Type	Length	Format	Codelist	Origin	Comment	Comment_zh	JoinID
2	ADEG	QTCFAG	AVALCAT1					Derived	if 450<ADEG.AVAL<=480 when ADEG.PARAMCD='QTCFAG' then ADEG.AVALCAT1='QTCF'>450 and <=480 msec', else if 480<ADEG.AVAL<=500 then ADEG.AVALCAT1='QTCF'>480 and <=500 msec', else if ADEG.AVAL>500 then ADEG.AVALCAT1='QTCF'>500 msec'.	如果450<ADEG.AVAL<=480则ADEG.AVALCAT1='QTCF'>450 and <=480 msec', 否则如果480<ADEG.AVAL<=500则 ADEG.AVALCAT1='QTCF'>480 and <=500 msec', 否则如果ADEG.AVAL>500则ADEG.AVALCAT1='QTCF'>500 msec'	EQ
3	ADEG	QTCFAG	AVALCAT1					Derived	if 450<ADEG.AVAL<=480 when ADEG.PARAMCD='QTCFAG' then ADEG.AVALCAT1='QTCF'>450 and <=480 ms', else if 480<ADEG.AVAL<=500 then ADEG.AVALCAT1='QTCF'>480 and <=500 ms', else if ADEG.AVAL>500 then ADEG.AVALCAT1='QTCF'>500 ms'.	如果450<ADEG.AVAL<=480则ADEG.AVALCAT1='QTCF'>450 and <=480 msec', 否则如果480<ADEG.AVAL<=500则 ADEG.AVALCAT1='QTCF'>480 and <=500 msec', 否则如果ADEG.AVAL>500则ADEG.AVALCAT1='QTCF'>500 msec'	EQ
5	ADEG	QTCFAG	AVALCAT1N					Assigned	The numeric value for ADEG.AVALCAT1 based on Codelist	基于ADEG.AVALCAT1变技术语列表的数值型赋值	EQ

Display 4. ValueLevel tab in ADaM Spec Generator

M_ADAM_DEINE

The table below shows the macro parameters and their usages.

Macro Parameter	Required?	Default	Usage
STUDYNAME	N		Sponsor internal name, can be same as Protocol Name.
PROTNAME	Y		Official study identifier.
PROTTITLE	Y		Official study title.
SDTMIGV	N	3.2	Corresponding SDTM IG version. IG metadata from CDISC library is used.
ADAMIGV	N	1.1	Corresponding ADaM IG version. IG metadata from CDISC library is used. SDTMIGV and ADAMIGV are only effective for define 2.0. For define 2.1, IG version should be set for each dataset in Datasets tab in the specification.
SDTMCTV	N	Latest Version	NCI SDTM Control Terminology applied. If missing, set to the latest version by default.
ADAMCTV	N		NCI ADaM Control Terminology applied. If missing, set to the latest version by default.
LANGUAGE	N	EN	Language used. Valid values are EN and ZH.
ADAM	N	ALL	ADaMs that included in define.xml, valid values are: ALL ADSL ADAE ADCM/etc. This parameter is case insensitive. ALL indicates that all the ADaMs with specifications provided will be included. Note that ADSL should always be included.
XPTLOC	N		The location of xpts that delivered to regulatory agencies. Set to the default folder in company directory structure if missing.
ADAMLOC	N		The location of ADaM datasets. If xpts are not provided, ADaM datasets will be read to provide relevant information. Set to the default folder in company directory structure if missing.
SPECLOC	N		The location of ADaM specifications. Set to the default folder in company directory structure if missing.

Macro Parameter	Required?	Default	Usage
SDTMDEFINE	N		Specify the path of SDTM define excel specification. If SDTM define excel specification is provided, the Codelist and CT terms for variables with ORIGIN set to Predecessor are derived from the SDTM define excel specification.
DEFINEV	N	2.0	Specify define version applied.
HELP	N		Put help information in log? Y/N

Table 1. The Parameters in M_ADAM_DEFINE

Next, a brief description of the steps used to generate the final P21E define excel specification is given.

1. Initialization. Perform the preliminary preparation, e.g., validating the macro parameters and setting default values, reading in NCI Control Terminology and IG metadata, preparing P21E define excel specification template, converting the ADaM xpts to work library, etc.
2. Import all the individual ADaM specifications and combine them together according to the tabs separately. Three datasets (DATASETS, VARIABLES, VALUELEVEL) are created, which contain all the ADaMs.

Derive the 'Where Clause' and filter the ADaMs as per VALUELEVEL, treat the subset as an independent dataset.

3. Generate the define metadata one by one, i.e., the tabs in the P21E define excel specification.
 - Define
Fill in values from macro parameters.

Attribute	Value
StudyName	XXXXXXXXXXXXXX
StudyDescription	XX
ProtocolName	XXXXXXXXXXXXXXXXXXXXXXXXXXXX
StandardName	ADaM-IG
StandardVersion	1.1
SDTMCTVersion	YYYY-MM-DD
ADaMCTVersion	YYYY-MM-DD
Language	zh

Display 5. Define tab in the P21E define excel specification

- Datasets

The Dataset tab in ADaM specification provides most of the required information. If the 'Comment' column in ADaM specification is not empty, then a comment ID will be assigned, such as 'ADEG_COM_'. Comment details will be populated in the Comment tab of P21E define excel specification. The columns 'Has No Data', 'Repeating' and 'Reference Data' will be derived according to actual ADaM data (ADaM.xpt).

Dataset	Label	Class	Structure	Key Variables	Repeating	Reference Data	Comment
ADEG	ECG分析数据集	基本数据结构	每个受试者、每个分析参数的每个分析时间点包含了一条或以上的记录	USUBID, AVISITN, ADT, ATPTN, PARCATIN, PARAMN, DTYPE, ATM, EGREPNUM	Yes	No	ADEG_COM_

Display 6. Datasets tab in the P21E define excel specification

ID	Description	Document	Pages
ADEG.COM	包含所有具有有效测量记录的受试者	ADRG	

Display 7. Comment tab in the P21E define excel specification

- Variables/ValueLevel

Dataset, Variable, Codelist and Origin in ADaM specifications will be carried over to P21E define excel specification. Label, Data Type, Length, Format are used for ADaM development and do not provide any effective information for define.xml.

Variables attributes in P21E define excel specification, i.e., Label, Data Type, Length, Significant Digits and Format, will be derived from actual ADaM datasets. In addition, Data Type, Length and Significant Digits should follow define.xml specification. All the values of a certain variable will be traversed to determine its attributes.

The combination of the columns Comment and Origin provides the most important information for define. The different values for Origin will trigger different operations on Comment. 'Predecessor', 'Derived', 'Assigned' are the most common values for Origin in ADaM define. In principle, Origin cannot be missing as per define.xml specification. The exception is for the variable also involved in ValueLevel, Origin can be left blank in Variable, since it has been specified in ValueLevel.

Origin	Comment
Predecessor	Enter Predecessor column in P21E define excel specification by the value of Comment
Assigned	Assign a comment ID in Variables/ValueLevel tab of P21E define excel specification, Comment details will be entered in the Comment tab of P21E define excel specification.
Derived	Assign a method ID in Variables/ValueLevel tab of P21E define excel specification, Comment details will be entered in the Method tab of P21E define excel specification.

Table 2. The rule of processing Comment to P21E define excel specification

- Codelists

The selected company level control terminologies are extraordinarily complete, including content and structure.

- Methods/Comments

Follow P21E define excel specification format to populate these two tabs, substantive information have been processed above.

ID	Name	Type	Description	expression
ADEG.ABLFL	衍生ADEG.ABLFL的算法	Computation	对于ADEG.DTYPE=AVERAGE的记录, 当每个受试者每个测试指标为首次用药前最近一次记录时标记ADEG.ABLFL=Y	
ADEG.ADT	衍生ADEG.ADT的算法	Computation	ADEG.ADT=EG EGDTC转换为数值型日期格式	
ADEG.ADY	衍生ADEG.ADY的算法	Computation	如果ADEG.ADT<ADSL.TRTSDT则ADEG.ADY=ADEG.ADT-ADSL.TRTSDT, 否则如果ADEG.ADT>=ADSL.TRTSDT则ADEG.ADY=ADEG.ADT-ADSL.TRTSDT+1	
ADEG.ANL01FL	衍生ADEG.ANL01FL的算法	Computation	当记录为ADEG.DTYPE=AVERAGE且ADEG.AVISIT1中不包含"计划外"且ADEG.AVISIT1="筛选期"时标记ADEG.ANL01FL=Y	
ADEG.ANL02FL	衍生ADEG.ANL02FL的算法	Computation	当ADEG.PARAMCD='INTP'时ADEG.AVALC为用药后最差等级时标记ADEG.ANL02FL=Y	

Display 8. Method tab in the P21E define excel specification

- Dictionaries/Document

Follow P21E define excel specification format to populate these two tabs. The Codelist set for coding variables in Variables/ValueLevel will be identified as a dictionary ID, and will be carried over to Dictionaries tab. Similarly, Document in Datasets/Variables/ValueLevel will be carried over to Document tab.

ID	Name	Data Type	Dictionary	Version
CTCAE	Common Terminology Criteria for Adverse Events	text	CTCAE	5
MEDDRA	Medical Dictionary for Regulatory Activities	text	MEDDRA	25.1
WHODD	Medications Dictionary	text	WHODD	GLOBALB3Mar22

Display 9. Dictionaries tab in the P21E define excel specification

ID	Title	Href
ADRG	分析数据审阅说明	adrg.pdf

Display 10. Documents tab in the P21E define excel specification

- Analysis Displays/Analysis Results

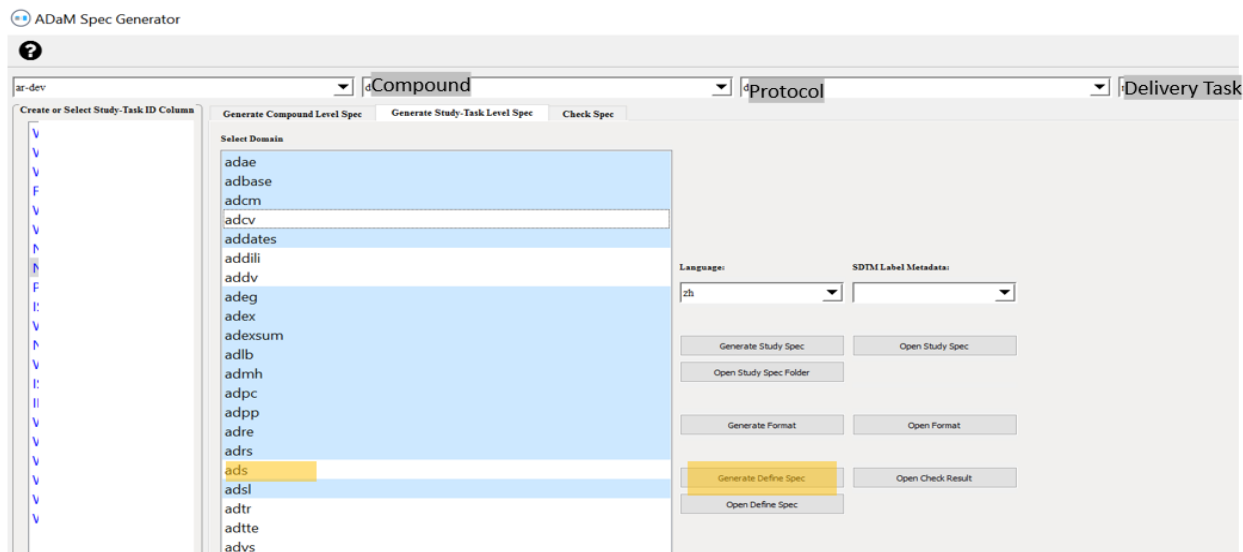
The empty tabs will be generated, and the details are required to be filled in manually. It will not cost much time since there are few analysis reports included generally.

Finally, the macro will produce the P21E define excel specification that can be import to P21E to get the nearly finished define.xml.

M_ADAM_DEFINE is integrated into ADaM Spec Generator and a bottom Generate Define Spec is added. Some required macro parameters, e.g., PROTNAME, PROTTITLE, are stored in a file called ADS. Even the company level control terminology, dictionaries and documents involved are also included in the ADS file. As shown in Display 12, user can select desired ADaMs from the left ADaM list and click the bottom Generate Define Spec to create P21E define excel specification.

Generate Compound Level Spec Generate Study-Task Level Spec Check Spec									
Select Your Macro (REQUIRED)			Select Your Names (REQUIRED)			Only 1 CODE from Study/Task ID (optional)			
ads			Codelists						
Main Worksheet			Study						
			Codelists						
			Documents						
			Dictionaries						
	ADaMName	Non_Extensible	ID	Name	NCI Codelist Code	Data Type	Term	NCI Term Code	Decoded Value
1	*ALL*	Y	NY	NY	C66742	text	N	C49487	No
2	*ALL*	Y	NY	NY	C66742	text		C48660	Not Applicable
3	*ALL*	Y	NY	NY	C66742	text	U	C17998	Unknown
4	*ALL*	Y	NY	NY	C66742	text	Y	C49488	Yes
5	*ALL*	Y	Y_NULL	Y_NULL	C66742	text	Y	C49488	Yes
6	*ALL*		TRT	Treatment		text	50 mg		
7	*ALL*		TRT	Treatment		text	100 mg		

Display 11. ADS setting interface in ADaM Spec Generator



Display 12. Generate Define Spec

COMPLIANCE ISSUE

In addition to the P21E define excel specification, M_ADAM_DEFINE also outputs the check results. There are many checks performed by the macro. Up to 28 check reports will be potentially outputted if any corresponding issue is detected. These checks focus on two aspects. One is the inconsistency among the ADaM specifications, ADaM xpts, IG metadata, etc. For example, the variable Label in the ADaM specification is different from the ADaM xpt. Although the label from actual dataset will be used in define directly, the issue will still be reported. Another check is to make sure that the final define excel specification conforms to define.xml specification. These checks will improve quality and enhance compliance with define.xml. Here are some examples of issue reported.

EXAMPLES

All the check reports are outputted into a file named define_checking_results.xml which can be displayed as a stylesheet. Each report occupies one tab.

- If there are duplicate terms for a certain Codelist ID, the issue will be reported since it is forbidden for define.xml.

DEFINE ERROR!									
Please check below repeat codelist terms in ADS file. Duplicates are not allowed in define.									
O	ADAMNAME	NON_EXTENSIBLE	ID	NAME	NCL_CODELIST_CODE	DATA_TYPE	ORDER	TERM	NCL_TERM_CODE
1	ADSL		RACE	种族	C74457	text	268	亚裔	C41260
2	ADSL		RACE	种族	C74457	text	267	亚裔	C41259

Output 1. Check Result - repeat_codelist_term_in_ads

- It is not reasonable to set different Origin and Comment and Method for one variable under the same filter condition.

DEFINE ERROR!							
Please check below where clauses, multiple definitions are not allowed for unique where clause.							
DATASET	VARIABLE	JOINID	ADTDESCD	ORIGIN	COMMENT	NUM	
ADDDATES	ADTDESC	EQ	LTADTIRC	Assigned	赋值方式为IRC最后一次有效肿瘤评估日期	2	
ADDDATES	ADTDESC	EQ	LTADTIRC	Derived	赋值方式为IRC最后一次有效肿瘤评估日期	2	

Output 2. Check Result - duplicates_for_where_clause

- If there is zero observation selected after applying the filter, such row in ValueLevel should be deleted. User can make decision to delete/modify the filtering in specification or in final P21E define excel specification, to avoid modifying specification in case the filtering cases do not happen when study is ongoing.

DEFINE ERROR!

Please check below where clause in valuelevel tab. No data selected when applying the where clause.

This issue will cause failure to import define.xlsx into P21E.

User could make decision to delete/modify the where clause in specification or in outputted define.xlsx (to avoid modifying specification in case the where clause cases are not happened at present during study ongoing)

DATASET	VARIABLE	JOINID	ADTDESCD
ADATES	ADT	EQ	LB2DTINV
ADATES	ADTDESC	EQ	LB2DTINV
ADATES	SRCDOM	EQ	LB2DTINV
ADATES	SRCSEQ	EQ	LB2DTINV
ADATES	SRCVAR	EQ	LB2DTINV

Output 3. Check Result - where_clause_issue

- It is required for the user to check whether the variable defined in the specification is forgotten to produce.

DEFINE WARNING

The following variables do not appear both in specification (variables and valuelevel tab) and dataset, please check specification and dataset.

Note the variable in dataset only will be ignored and not processed in define, the variable in specification only will be processed in define.

DATASET	VARIABLE	Variable Source
ADAE	AOCCSFL	Variable in Spec not in Dataset
ADAE	AOCCSIFL	Variable in Dataset not in Spec

Output 4. Check Result - var_existence_between_spec_ds

- In the first row of this example, besides the subject variable ADDILI.AVAL, there are two extra variables (ADDILI.PARAMCD, ADLB.AVAL) appearing in COMMENT, making it difficult to identify which one is the predecessor variable. Generally, the predecessor variable should be a single variable. When there is more than one variable in COMMENT, please consider modifying ORIGIN or setting a more appropriate COMMENT.

DEFINE WARNING

Please check below Predecessor value for origin in VARIABLES/VALUELEVEL tab. Predecessor may not be very applicable for these variables.

Either there are multiple predecessor datasets/variables set in Comment, or non-predecessor dataset/variable were set.

Please consider modify the Predecessor to other origin values or set Predecessor in VALUELEVEL if applicable.

Note the extracting of codelists from SDTM define spec for these Predecessor variables will not be performed even if the SDTM define spec exist.

The issued Predecessor in VARIABLES or VALUELEVEL	DATASET	VARIABLE	JOINID	PARAMCD	CODELIST	ORIGIN	COMMENT	Derived Predecessor Dataset from Comment	Derived Predecessor Variable from Comment
VARIABLES	ADDILI	AVAL				Predecessor	If ADDILI.PARAMCD is in (ALT, 'AST', 'ALP', 'BLU') then ADDILI.AVAL-ADLB.AVAL ADDILI.ADT-ADLB.ADT when ADLB.PARAMCD includes (ALT, 'AST') measurements are exist for every scheduled or unscheduled visit	ADLB	PARAMCD AVAL
VALUELEVEL	ADDILI	ADT	EQ	AT		Predecessor		ADLB	ADT PARAMCD

Output 5. Check Result - predecessor_issue

- For CV.CVORRESU/ CV.CVSTRESU/ EG.EGORRESU, different codelists are set in both Variables and ValueLevel. It is unnecessary. Keeping the codelist in either Variable or ValueLevel tab is recommended.

DEFINE WARNING

There are different CT set for the following variables in VARIABLE and VALUELEVEL tab in SDTM define.

Please note the CT set in VARIABLE will be used for predecessor variable CT derivation in ADaM define generation.

DATASET	VARIABLE	CODELIST	In VALUELEVEL Tab?
CV	CVORRESU	CVORRESU	N
CV	CVORRESU	CVORRESU.CV.CVTESTCD.EQ.LVEF	Y
CV	CVSTRESU	CVSTRESU	N
CV	CVSTRESU	CVSTRESU.CV.CVTESTCD.EQ.LVEF	Y
EG	EGORRESU	EGORRESU.EG.EGTESTCD.EQ.RRAG	Y
EG	EGORRESU	EGORRESU.EG.EGTESTCD.EQ.EGHRMN	Y
EG	EGORRESU	EGORRESU.EG.EGTESTCD.EQ.PRAG	Y
EG	EGORRESU	EGORRESU.EG.EGTESTCD.EQ.QTCFAG	Y
EG	EGORRESU	EGORRESU.EG.EGTESTCD.EQ.QRSAG	Y
EG	EGORRESU	EGORRESU.EG.EGTESTCD.EQ.QTAG	Y
EG	EGORRESU	EGORRESU	N

Output 6. Check Result - sdtm_ct_issue

- In ADAE specification, AE.AEBDSYCD/AE.AEBODSYS are specified as the predecessor variable for ADAE.AEBDSYCD/ADAE.AEBODSYS, however, AE.AEBDSYCD/AE.AEBODSYS do not appear in SDTM define excel specification.

DEFINE WARNING
Please check below Comment for Predecessor variable in VARIABLES/VALUELEVEL tab.
The predecessor SDTM variables in ADaM spec not found in SDTM define spec.

The issued Predecessor in VARIABLES or VALUELEVEL	DATASET	VARIABLE	ORIGIN	COMMENT
VARIABLES	ADAE	AEBDSYCD	Predecessor	AE.AEBDSYCD
VARIABLES	ADAE	AEBODSYS	Predecessor	AE.AEBODSYS

Output 7. Check Result - SDTM_var_issue_in_predecessor

- In this example, Origin for CRIT1 was set as 'Derived' in Variable tab while 'Assigned' was set in ValueLevel. It is a logic contradiction.

DEFINE ERROR!
Please check below origin values in variable and valuelevel tab which are not same.
Note all the ORIGIN will be set missing for valuelevel variables in Variables metadata of define despite what ORIGIN value set in ADaM define.

DATASET	VARIABLE	JOINID	PARAMCD	Origin in Variable Tab	Origin in ValueLevel tab
ADDILI	CRIT1	EQ	BILI	Assigned	Derived
ADDILI	CRIT1	EQ	ATBIL	Assigned	Derived
ADDILI	CRIT1	IN	ALT, AST, AT	Assigned	Derived
ADDILI	CRIT1	EQ	ALP	Assigned	Derived
ADDILI	CRIT1	EQ	ATBILALP	Assigned	Derived

Output 8. Check Result - xcheck_origin_var_valuelevel

- The red highlighted issue must be corrected. The Codelist ID CMBOR was specified for ADCM.CMBOR, but CMBOR was not in ADS file.

DEFINE WARNING/ERROR
Please check below inconsistency of codelist ID between specification and ads.
If codelist only in ads, please check and confirm whether to delete it from ads or add in specification. These codelists will still appear in define.
ERROR! If codelist only in specification, the issue will cause importing failure into P21E. Exception for AVISIT/AVISITN which CT term will be derived from ADaM datasets.

CODELIST	Codelists or Dictionary	Codelist in ADaM Spec or ADS SDTM Define Spec	ADS or SDTM Define Spec	DATASET	VARIABLE
AREL	Codelists	Only in ADS/SDTM Define Spec	ADS		
CMBOR		Only in ADaM Spec		ADCM	CMBOR
CTCAE	Dictionary	Only in ADS/SDTM Define Spec	ADS		
DOSCAT1	Codelists	Only in ADS/SDTM Define Spec	ADS		
DOSCAT1N	Codelists	Only in ADS/SDTM Define Spec	ADS		
DTYPE	Codelists	Only in ADS/SDTM Define Spec	ADS		
EFF_PARQUAL	Codelists	Only in ADS/SDTM Define Spec	ADS		
MEDDRA	Dictionary	Only in ADS/SDTM Define Spec	SDTM Define Spec		

Output 9. Check Result - diff_codelist_ID_spec_ads

- BMIBLGR1 is detected as the paired decode variable for BMIBLG1N. There is a Codelist BMIBLG1N set for variable BMIBLG1N. However, there is no Codelist set for BMIBLGR1. It is better to assign a Codelist for BMIBLGR1.

Define Warning
There is no CT ID specified for the following derived decode variable. Generally, it is better to assign CT for these decode variables.
For instance, APHASEN has a CT apshse, while APHASE has no CT.

DATASET	VARIABLE	CODELIST	Derived Potential Decoded Variable	Codelist for Derived Potential Decoded Variable
ADSL	BMIBLG1N	BMIBLG1N	BMIBLGR1	

Output 10. Check Result - decode_var_ct

- The red highlighted issue, where term values are in ADaM data but not in the ADS codelists tab, must be fixed.

DEFINE WARNING
Please check below inconsistency of codelist term between ADaM data and ads. These terms are not in both ADaM data and ads
Red highlighted is for term only in ADaM data, it must be fixed.

CODELIST	Term	Term Exist in ADaM data or ADS	SDTM Define Spec	ADS or SDTM Define Spec	DATASET	VARIABLE	JOINID	PARAMCD
ADEXSUM_PARAM	Any Dose Reductions	Only in ADS/SDTM Define Spec	ADS		ADEXSUM	PARAM		
ADEXSUM_PARAM	Number of Dose Reductions	Only in ADS/SDTM Define Spec	ADS		ADEXSUM	PARAM		
ADEXSUM_PARAMCD	DOSRED	Only in ADS/SDTM Define Spec	ADS		ADEXSUM	PARAMCD		
ADEXSUM_PARAMCD	NDOSRED	Only in ADS/SDTM Define Spec	ADS		ADEXSUM	PARAMCD		
ADEXSUM_PARAMIN	21	Only in ADS/SDTM Define Spec	ADS		ADEXSUM	PARAMIN		
ADEXSUM_PARAMIN	22	Only in ADS/SDTM Define Spec	ADS		ADEXSUM	PARAMIN		
ADPC_ATPT	Pre-dose	Only in ADaM data	ADS		ADPC	ATPT		

Output 11. Check Result - diff_codelist_term_spec_ads

- SDTM/ADaM IG specifies the control terminology for these variables, however, there are no Codelist set in specification. It is recommended for users to evaluate whether to add Codelist for the variables.

DEFINE WARNING
Please check below records, which might have an issue on CT applied.
For a variable with CT used in IG, either the codelist is missing or the nci codelist code is missing.

DATASET	VARIABLE	CODELIST	NCI codelist code applied in ADS	NCI codelist code applied in IG
ADAE	AEBDSYCD			MedDRA
ADDV	EPOCH			C99079
ADEG	EGSTRESC			C71150
ADEX	EXDOSFRM			C66726
ADEX	EXDOSFRQ			C71113
ADEX	EXDOSU			C71620
ADEX	EXROUTE			C66729
ADLB	LBNRIND			C78736
ADLB	LBSTRESC			C102580
ADPC	PCSPEC			C78734

Output 12. Check Result - CT_inconsistence_with_IG

EXPLORATION

P21E plays the role of transforming excel into XML files and can verify define.xml. It is highly likely that alternative solutions will be found. In fact, the above checks have some overlap with the validation of the P21E definition, e.g., checking whether the Class or the Origin are compliant with the NCI definition terms. P21E define validation can be entirely replaced when the checks become much more comprehensive. Regarding the transition from excel to xml, there are two alternatives as follows.

SAS CLINICAL STANDARDS TOOLKIT

The SAS Clinical Standards Toolkit is an open-source toolkit provided by SAS inc., which focuses on the standards defined by the CDISC. The latest version 1.7 has supported CDISC define-xml 2.0.

Like P21E, there are 7 metadata should be populated and the toolkit can transform the metadata to define.xml directly. The 7 metadata are source_study, source_tables, source_columns, source_values, source_codelists, source_documents and source_analysisresults. All the inputted sources that M_ADAM_DEFINE required can be utilized to generate the metadata except source_analysisresults. That means there is the possibility to produce xml through the toolkit instead of P21E.

The following displays an example that a macro is invoked to produce the xml.

```
* Read the source metadata;
%define_sourcetodefine(
  _cstOutLib=srcdata,
  _cstSourceStudy=sampdata.source_study,
  _cstSourceTables=sampdata.source_tables,
  _cstSourceColumns=sampdata.source_columns,
  _cstSourceCodeLists=sampdata.source_codelists,
  _cstSourceValues=sampdata.source_values,
  _cstSourceDocuments=sampdata.source_documents,
/* Uncomment the following when including Analysis Results Metadata for ADaM */
/* _cstSourceAnalysisResults=sampdata.source_analysisresults, */
  _cstFullModel=N,
  _cstCheckLengths=Y,
  _cstLang=en
);

* Create the Define-XML file;
%define_write(_cstCreateDisplayStyle=1);
```

PINNACLE 21 COMMUNITY

Pinnacle 21 Community (P21C) is an open-source App that provided by Pinnacle inc., which can also read in P21E excel define specification and transform it to xml.

CONCLUSION

Developing the ADaM specification through a designed format is required. It will take slightly more time during the specification preparation, e.g., adding Origin, specifying Codelist, and splitting the dataset into ValueLevel at the filtering level. However, the ADaM Spec Generator smooths the ADaM specification production and redeems the extra time spent. Through M_ADAM_DEFINE, it really speeds up the generation of define.xml extremely and can improve the quality.

The five advantages of this approach to generating define.xml are summarized below.

- Expand the functions of ADaM specifications, which are not only developed for ADaM datasets production, but also support for define generation. The ADaM specifications will no longer become useless files after the ADaM is finalized.
- Further improve the quality of ADaM by performing a lot of checks including detecting potential issues of ADaM, for instance, variable values have typos and cause the inconsistency with corresponding Codelist term.
- Enhance the efficiency to generate define.xml. Leveraging the P21E's ability to transform excel into XML, while avoiding a lot of manual work.
- Coordinated with the define validation metadata in P21E. Highly improved quality of define.xml and compliance with define.xml specifications.
- A promising approach that can reduce dependence on P21E in future, by utilizing the open-source toolkit to perform xml transformation, e.g., P21 Community, SAS Clinical Standards Toolkit.

REFERENCES

Zhi Xu etc.. 2022. "ADaM Specifications Automation from Company Standards to Study"
<https://www.lexjansen.com/pharmasug-cn/2022/AD/Pharmasug-China-2022-AD141.pdf>

CDISC Define-XML Team. "CDISC Define-XML Specification Version 2.1 (Final)". July 6, 2021.
<https://www.cdisc.org/standards/data-exchange/define-xml/define-xml-v2-1>

ACKNOWLEDGMENTS

Thanks to all statistical programming team members at Dizal for their help to facilitate the success.

RECOMMENDED READING

- CDISC Define-XML Specification Version 2.1

CONTACT INFORMATION

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

Name: Daoqing Li
Enterprise: Dizal
E-mail: Daoqing.Li@dizalpharma.com