Paper SM14

# ID-SPECTACKLE: Making ADaM specifications “Fun”

Sophie Dekeyser, IDDI, Louvain-la-Neuve, Belgium
Lionel Debecq, IDDI, Louvain-la-Neuve, Belgium

## ABSTRACT

ID-Spectackle is a user-friendly graphical interface to guide statistical programmers. This tool is meant to help and prepare the ADaM Specifications and create the Define-XML for submission. It helps to limit a lot of manual input by automating the metadata collection and provides guidance to fill in the specifications. As the basis of the specifications, a SAS® Macro collects metadata, with support of the ADaM IG and NCI Controlled Terminology.
This automated process presents multiple advantages:
- it avoids error-prone copy-pasting of redundant information among sources,
- it ensures that specifications and datasets are aligned,
- it provides consistency checks with completion rules,
- the evolution of standards can be implemented directly in the system,
- it ensures that the specifications format is compliant with the one from Pinnacle 21 (P21),
- it clarifies and deploys a standard process company wide.

## INTRODUCTION

Study Specifications are notorious to be the last thing a programmer wants to do.
While study specifications are, to a certain extent, an intricate part of any study submission under the form of Define-XML, it is an important document, but painful to write, no matter when they are created.
Another difficulty is specifications are not standardized amongst companies. Fortunately Pinnacle 21 sorts this out by providing a template that is easy enough to implement.

Unfortunately, a lot of manual entries are required when completing study specifications. This has many disadvantages, including the time spent completing the document with all required information, usually from scratch, and the risk of error with manual completion.

## ID-SPECTACKLE SMALL STORY

To summarize, the main objective of our job is the production of analysis reports. But creating a statistical report is a complex undertaking. This process does not merely entail the utilization of SDTM datasets and the execution of analysis program. The initial stage is the generation of "analysis datasets", which serve as the foundation for the subsequent programming of tables, figures, and listings. However, the production of analysis datasets is not as straightforward as one might assume. It is essential to consider the Statistical Analysis Plan (SAP) and study documents, such as the protocol and CRF. Furthermore, analysis datasets must adhere to CDISC ADaM standard and be validated with P21. A complete validation with P21 would include Define-XML, therefore the completion of study specifications. For all submissions, the specifications should be included, therefore the Define-XML, which allows reviewers to understand what has been done on the data. These specifications must be written in a clear and understandable manner.

However, the process of writing specifications is not straightforward. This is why we needed to automate the generation of specifications as often as possible. An idea was proposed that would reduce the time required to produce specifications, improve the quality, engage the programmer in a timely and accurate creation of documentation, (not only once at the end of programming, like it could be done in P21, but all along the creation of the ADaM datasets), and automate the updates performed on them. The idea was ID-Spectackle.

## GLOBAL IDEA

The global idea behind ID-Spectackle is a two-step process: first step is to gather all possible information automatically from the data themselves, CDISC Implementation Guides (IG), and Terminologies (National Cancer Institute (NCI) Controlled Terminologies (CT)); second step is to enter all derivations algorithms, comments, and specificities.

The first step is handled by a specific all-in-one SAS® Macro, in combination with CDISC SDTM/ADaM IGs and NCI CTs. The second step is handled by a user-friendly WebUI (**Web U**ser **I**nterface) built in R, with the R-Shiny package. The WebUI is conceived with the P21 template and Define-XML 2.x as standards in mind.

## %ADaM

There wouldn't be an ID-Spectackle without the ADaM macro. This is the angular stone of the specification automation. Its initial function was to bring all the metadata together from the data as completely as possible.

```
/* minit.sas */
%macro minit;
    %global adamig adamct sdtmig sdtmct;
    %let adamig=1.3;
    %let adamct=;
    %let sdtmig=3.4;
    %let sdtmct=2024-06-13;
%mend minit;

/*adsl.sas*/
%macro ADSL;
    %minit;

    /*
       Derivations
          ...
     */


    proc sort data=final00
              out =final_adsl(label="Subject-Level Analysis Dataset");
        by USUBJID;
    run;

    %ADAM(ds_     = final_adsl
         ,adam_   = derived.adsl
         ,libsdtm_= data
         ,class_  = ADSL
         ,popfl_  = ITTFL FASFL SAFFL
         ,newkey_ = USUBJID);
%mend adsl;
%adsl;
```

*Snippet 1 Example program*

While the main goal is to bring all the metadata together from the data, the different functionalities can be divided into three sub-categories: (1) Check for consistency with the data model; (2) Collect Metadata; (3) Finalize the dataset.

(1) We need to ensure the compliance with the CDISC ADaM IG (name, label, length) before finalizing the dataset on disk and creating the associated XPT. We could, of course, rely on P21 to validate everything on the variables according to the model, however everything that can be done upfront is preferable and saves time.

There are multiple checks in place: label length, missing labels for variables and dataset, mandatory linked variables (from data model), variable types (from data model), variables with name higher than eight characters, assigns (or replaces) labels for standard variables, validates the key by checking for duplicates, compares dataset data and attributes between previous and actual run to check for differences.

(2) The metadata are collected in the following metadata datasets: metadataset, metavariable, and metacodelist. ValueLevelMetadata (VLM) is not in the scope of the macro, as it is very specific to the project and derivations.

Variable name, label, format and order are initially taken from the SAS Dictionary dataset for both ADaM Derived dataset and the predecessors SDTM variables.

For SDTM, there is not much to do, we must collect the metadata for the variables kept in the analysis dataset and collect the codelists associated to the SDTM IG.

The origin for the SDTM Variables kept for the analysis, both as supporting variables or as traceability variables, is defaulted to "Predecessor". SUPPQUAL domain variables are set to "Derived" with a preset algorithm based on the variable name such as "Equals to SUPP--.QVAL where SUPP--.QNAM='…'". If any Controlled Terminology applies on the variable, it is checked against the applicable SDTM IG version data model and only the values present in the SDTM dataset are collected for codelist.

For the ADaM data, most of the variables are set to "Derived" and a derivation algorithm will be required. Some algorithms are defined per variable in the "data model" file for the usual ones, e.g. ADY. However, all cannot be automatically prefilled and will require user input in the WebUI.

The custom variables are unknown to the model and neither origin nor any method can be prefilled for them, these will also require user input in the WebUI.

To make the link between the data and the metadata, it primarily relies on the CDISC SDTM IG, the CDISC ADaM IG, the NCI ADaM CT and the NCI SDTM CT.
The "data model" file for ADaM contains the latest IG version available, 1.3 at time of writing, while SDTM is based on the IG 3.2 to 3.4. The IG and CT parameters are set for the study as global macro parameters (Snippet 1).

The "data model" file for ADaM is provided in XLSX format and is customized in-house, as we have added extra information such as reference variable for numeric equivalent (e.g. PARAMCD/PARAMN), some standard derivation algorithms and some other information proper to our usage (flags and references).

| Variable_ Name | Variable_label | Variable _type | Core | Origin | Algorithm | OrdCat | OrdC atn | OrdS catn | Length_ max | Length_reco mmended | Codelist_ name | Codelist_ decode | Mandatory _variable | Dictionary |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TRTP | Planned Treatment | Char | Cond | Derived | | Record-Level | 2 | 1 | 200 | 200 | TRT | | | |
| TRTPN | Planned Treatment (N) | Num | Perm | Assigned | Numeric counterpart of TRTP | Record-Level | 2 | 2 | 8 | 8 | TRTN | TRT | TRTP | |
| TRTPG$$ | Planned Pooled Treatment | Char | Perm | Derived | | Record-Level | 2 | 5 | 200 | 200 | TRTP | | | |
| TRTPG$$N | Planned Pooled Treatment | Num | Perm | Assigned | Numeric counterpart of TRTPG$$ | Record-Level | 2 | 6 | 8 | 8 | TRTPN | TRTP | TRTPG$$ | |

Dataset | Key | ADSL | BDS | OCCDS | SDTM_order | Codelist

The SDTM "data model" contains only variables with Controlled Terms for the metadata collection.

| DATASET | VARNAME | CDISC_ID | CDISC_NAME | CDISC_CODE | Version |
|---|---|---|---|---|---|
| AE | AEACN | ACN | Action Taken with Study Treatment | C66767 | SDTM-IG 3.4 |
| AE | AEACNDEV | DEACNDEV | Device Events Action Taken with Device | C111110 | SDTM-IG 3.4 |
| AE | AECONTRT | NY | No Yes Response | C66742 | SDTM-IG 3.4 |
| AE | AEENRF | STENRF | Relation to Reference Period | C66728 | SDTM-IG 3.4 |
| AE | AEENRTPT | STENRF | Relation to Reference Period | C66728 | SDTM-IG 3.4 |
| AE | AELOC | LOC | Anatomical Location | C74456 | SDTM-IG 3.4 |
| AE | AEOUT | OUT | Outcome of Event | C66768 | SDTM-IG 3.4 |

SDTM-IG 3.2 | SDTM-IG 3.3 | SDTM-IG 3.4

The NCI CT files are downloaded directly from the NCI website with PROC HTTP (Snippet 2) and imported in SAS dataset. If the file exists on the server with the adequate date version, the file is imported. If it doesn't, the response is usually an error 404 and the process stops.

```
filename fout "&file";
filename hdrs temp;
proc http
    %if %length(&version)=0 %then %do;
    url="https://evs.nci.nih.gov/ftp1/CDISC/&CDISC./&CDISC.%20Terminology.xls"
    %end;
    %else %do;
    url="https://evs.nci.nih.gov/ftp1/CDISC/&CDISC./Archive/&CDISC.%20Terminology%20&version..xls"
    %end;
    method="get" out=fout headerout=hdrs; debug level=1;
run;

data _null_;
    infile hdrs scanover truncover;
    input @'HTTP/1.1' code 4. message $255.;
    call symputx('status_code',code,'g');
run;
```

*Snippet 2 PROC HTTP to download either the latest version of the CT or a specific one, providing it exists status_code, read in "headerout")*

If an upgrade of a CT is required (e.g. SDTM upgraded the CT Version), the *_forceSDTMCT* or *_forceADAMCT* parameter (the parameters are both set by default to N, while Y is for enabling the feature) would be used one time in ADSL to overwrite the downloaded file, then a rerun of every ADaM dataset with the parameter set to N is required to overwrite the Terminology version in the metadata and upgrade to the terms, code, decode, codelist code, etc.

Every variable not in IG might have a codelist, except if the number of values for the variable exceeds the threshold *(_nvalues_ct*, with a default value of 20 per variable). Variables with a codelist in SDTM or ADaM IG will automatically get an associated codelist, these variables don't depend on the threshold.

(3) The finalization process creates a SAS dataset (from *ds_* parameter and specified by *adam_* parameter) where all character variables are resized to their maximum length and writes it to disk. The macro also includes an option (*modlord_*, Y or N) to either keep the ordering of the variables as the same as programmed or set it to the order suggested by the model.

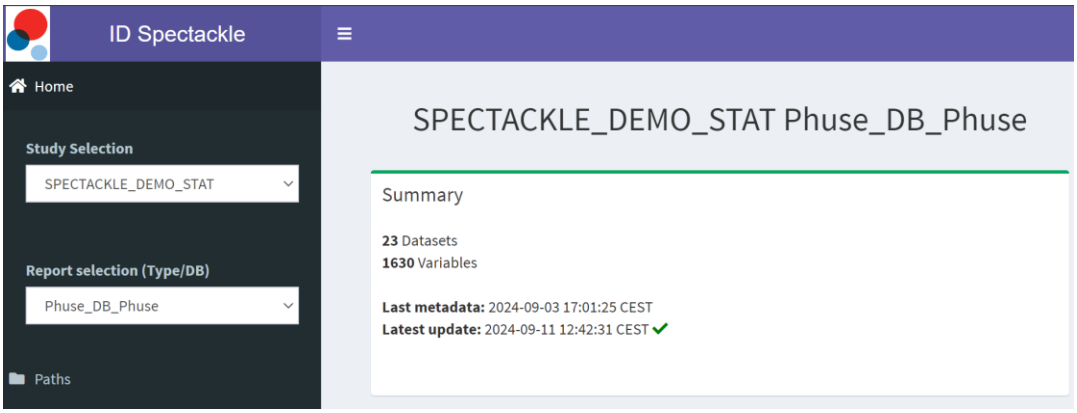
All the parameters available for the macro and their usage are listed in the table below:

| | |
|---|---|
| *ds_* | Temporary dataset name used as source to create final ADaM dataset. e.g. FINAL_ADSL |
| *adam_* | Library and dataset name for the final ADaM Dataset. e.g. DERIVED.ADSL |
| *libsdtm_* | SDTM library where study data are located |
| *class_* | Class of dataset. Only values accepted are ADSL, BDS, OCCDS, OTHER |
| *popfl_* | One or more population flags are required for ADSL. e.g. ITTFL FASFL SAFFL |
| *newkey_* | Unique Key. Serves to identify unique records within dataset. Should be same as sorting key. |
| *modlord_* | Ordering of the dataset's variables suggested by CDISC ADaM IG. Otherwise, keep the order defined during programming. N, or Y by default. |
| *forcexpt_* | In case of ERROR for some checks such as label length > 40 characters that would prevent the XPT being generated, force despite the error. Y, or N by Default. |
| *ADAMIG_* | Specify the ADaM IG version, for Define-XML e.g. 1.3 |
| *ADAMCT_* | NCI ADaM CT version, i.e. yyyy-mm-dd. Version should exist, leave empty for latest version. |
| *_forceADAMCT* | Force upgrade to specified version. Y, or N by Default. |
| *SDTMIG_* | Specify the SDTM IG version, for Define-XML e.g. 3.4 |
| *SDTMCT_* | NCI SDTM CT version, i.e. yyyy-mm-dd. Version should exist, leave empty for latest version. |
| *_forceSDTMCT* | Force upgrade to specified version. Y, or N by Default. |
| *_DTCPart* | Handles partial/incomplete date/time in SDTM. —DTC Variables, for Define-XML<br>By default, the partial/incomplete date/time are not taken into account and set to date or datetime. User has to set to Y to use the functionality.<br><br>It is a recent addition to the increase in the information provided in the Define-XML. It takes the value of a –DTC variable and checks for the missing component of the iso8601 date/time format applied to the variable. The different cases are reported in the table below:<br>All components of a date or date/time are present: date: *yyyy-mm-dd*; datetime: *yyyy-mm-ddThh:mm:ss*<br>Missing any right part of the date or date/time: partialDate: *yyyy, yyyy-mm*; partialTime: *hh:mm, hh*; partialDatetime: *yyyy-mm-ddThh:mm, yyyy-mm--Thh:mm:ss, yyyy-mm-ddThh, etc.*<br>Missing any component that are not partial (with any component missing not on the right): incompleteDatetime: *--mm-ddThh:mm, yyyy-mm-ddThh:-:ss, etc.* |
| *variables_codelist* | Specify a list of variables for codelist. |
| *_nvalues_ct* | Threshold for number of values to consider as codelist, if the number exceeds the threshold, the values are not treated as a codelist. Default is 20. |
| *debug_* | Debug option to keep interim datasets in WORK library. N, or Y by Default. |

## ID-SPECTACKLE

With a few images, this section presents ID-Spectackle, our user-friendly graphical interface which facilitates the creation of specifications and Define-XML and makes it (almost) fun.

### DEFINE STUDY AND DATABASE



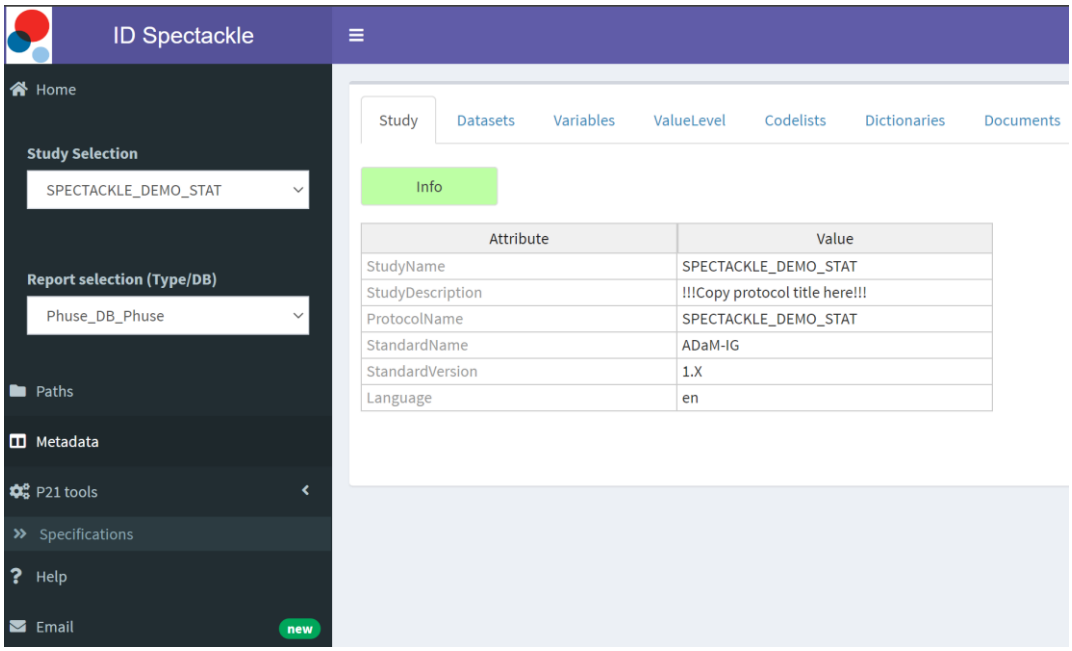

The first step is to select the Study and Report for which we are completing the specifications. Report refers to a combination of the type of analysis performed (IDMC, interim, final analysis etc) and database (DB) version.

At the beginning of a project, new derived ADaM datasets and, of course, new metadata are created by the %ADaM Macro. At that stage, only metadata stored into metadata datasets are avalaible. Nothing has been added or modified in ID-Spectackle. Metadata are displayed in “Metadata boards”, or “Metaboards” for short.

During the course of a project, DB will change to newer extraction from Data Management. The specifications do not have to be done again at each rerun due to a new DB. The system can copy over the previously completed specifications. The system allows the recovery of the previous selected DB as base for the specifications and use the new metadata datasets as an additional source (e.g. extra variables, new codelists, etc). It is also possible to recover the specifications from another similar study and copy them for reuse.

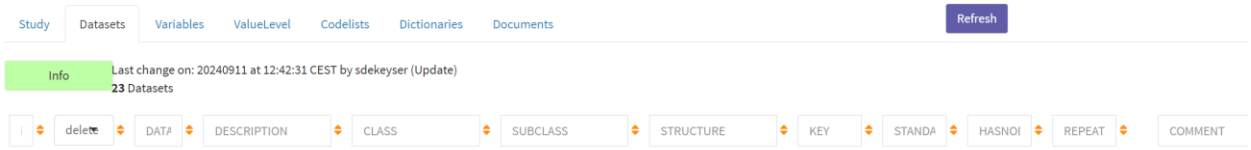
### METABOARD STUDY

This is the board with information about the study itself, such as Name, Description (from Protocol), IG version used.



**METABOARD DATASETS**

The datasets metaboard contains the metadata for derived ADaM datasets, such as Name, Description, Class, etc. Most of the board is prefilled by the %ADaM Macro.The dark grey cells can be updated; the others are fixed. Columns with a label containing “only XML2.1” are columns needed for Define-XML version 2.1 specifically (true for all metaboards).




| DATASET | DESCRIPTION | CLASS | SUBCLASS (only XML2.1) | STRUCTURE | KEY | STANDARD (only XML2.1) | HASNODATA (only XML2.1) | REPEATING | COMMENT |
|---|---|---|---|---|---|---|---|---|---|
| ADSL | Subject-Level Analysis Dataset | SUBJECT LEVEL ANALYSIS DATASET | | One record per subject | USUBJID | ADaM-IG 1.2 | no | no | |
| ADABO | ABO and antibody Analysis Dataset | ADAM OTHER | | One record per subject | USUBJID | ADaM-IG 1.2 | no | yes | |
| ADAE | Adverse Events Analysis Dataset | OCCURRENCE DATA STRUCTURE | ADVERSE EVENT | One record per subject per adverse event | USUBJID, AETERM, ASTDT, MXSEVGRN | ADaM-IG 1.2 | no | yes | |

**METABOARD VARIABLES**

The Variables metaboard contains metadatas for all variables of all datasets, such as label, type, potential codelist, origin etc. To further describe this board, we consider ADAE as an example and select all ADAE variables thanks to the filter above the board, among many other filters available.

The next screenshot presents the “Predecessor” variables from ADSL which are automatically completed with their associated codelists.

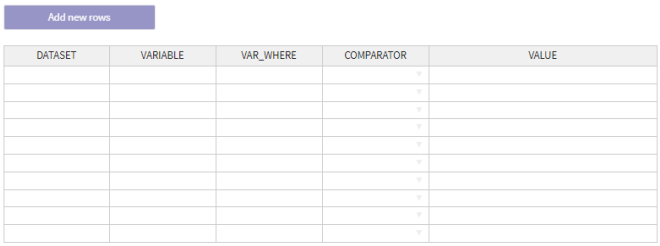
| DATASET | VARIABLE | LABEL | DATA_TYPE | LENGTH | SIGN_DIGIT | FORMAT | CORE | CODELIST | COMMON (only XML2.1) | ORIGIN | SOURCE (only XML2.1) | Predecessor/Method | HASNODATA (only XML2.1) | COMMENT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADAE | STUDYID | Study Identifier | text | 8 | | | Req | | | Predecessor | | ADSL.STUDYID | no | |
| ADAE | USUBJID | Unique Subject Identifier | text | 18 | | | Req | | | Predecessor | | ADSL.USUBJID | no | |
| ADAE | SUBJID | Subject Identifier for the Study | text | 9 | | | Perm | | | Predecessor | | ADSL.SUBJID | no | |
| ADAE | SITEID | Study Site Identifier | text | 4 | | | Perm | | | Predecessor | | ADSL.SITEID | no | |
| ADAE | AGE | Age | integer | 2 | | | Perm | | | Predecessor | | ADSL.AGE | no | |
| ADAE | SEX | Sex | text | 1 | | | Perm | SEX | | Predecessor | | ADSL.SEX | no | |
| ADAE | COUNTRY | Country | text | 3 | | | Perm | | | Predecessor | | ADSL.COUNTRY | no | |
| ADAE | SAFFL | Safety Population Flag | text | 1 | | | Perm | NY | | Predecessor | | ADSL.SAFFL | no | |

On this second screenshot, when information is missing and requires user input, the cell is highlighted in blue as a visual cue to help specifications completion. When ORIGIN is “Assigned”, information is expected in the COMMENT column, which is highlighted in blue too. When ORIGIN is “Derived”, and when no standard algorithm (method) exists, information is expected in the Predecessor/Method column, also in blue. When a variable is not in the ADAM IG then ORIGIN and SOURCE must be completed by the user.

| DATASET | VARIABLE | LABEL | DATA_TYPE | LENGTH | SIGN_DIGIT | FORMAT | CORE | CODELIST | COMMON (only XML2.1) | ORIGIN | SOURCE (only XML2.1) | Predecessor/Method | HASNODATA (only XML2.1) | COMMENT |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADAE | ADECOD | Analysis Dictionary-Derived Term | text | 43 | | | Perm | | | Derived | Sponsor | | no | |
| ADAE | ADURU | Analysis Duration Units | text | 4 | | | Cond | UNIT | | Assigned | Sponsor | | no | |
| ADAE | PREFL | Pre-treatment Flag | text | 1 | | | Cond | Y_BLANK | | Derived | Sponsor | | no | |
| ADAE | TRTEMFL | Treatment Emergent Analysis Flag | text | 1 | | | Cond | Y_BLANK | | Derived | Sponsor | | no | |
| ADAE | AREL | Analysis Causality | text | 11 | | | Perm | AREL | | Derived | Sponsor | | no | |
| ADAE | ARELN | Analysis Causality (N) | integer | 1 | | | Perm | ARELN | | Assigned | Sponsor | | no | Numeric counterpart of AREL |

**METABOARD VALUELEVEL**

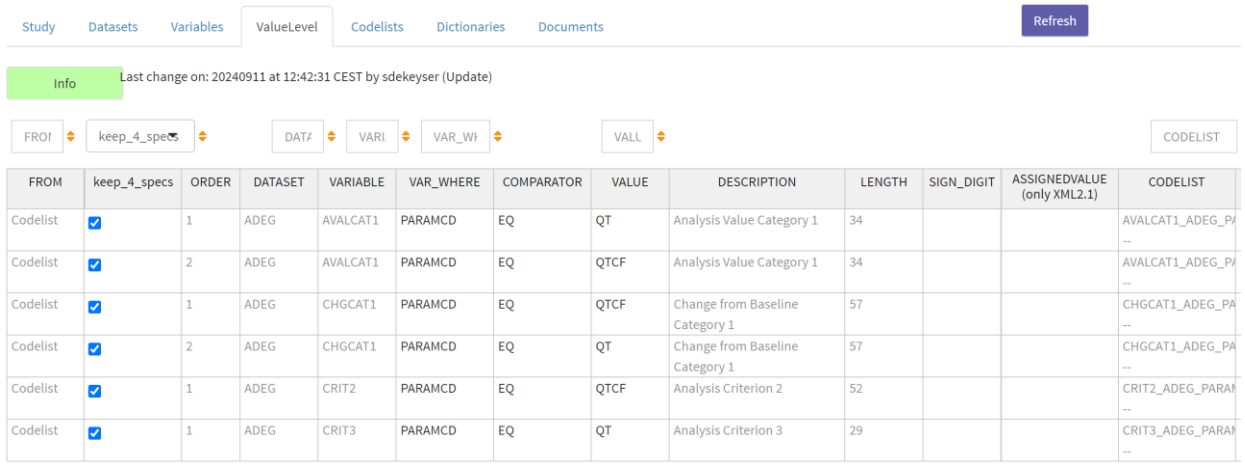
This metaboard is always empty at the beginning. It contains VLM information but is not prefilled by the %ADaM Macro as this is very specific to the study specifications implementation. All required information will be added manually as shown below:




Example

| | DATASET | VARIABLE | VAR_WHERE | COMPARATOR | VALUE |
|---|---|---|---|---|---|
| 1 | ADVS | AVAL | PARAMCD | IN | HEIGHT WEIGHT |
| 2 | ADEG | AVALCAT1 | PARAMCD | EQ | HR |

Then the ValueLevel data will be populated automatically in the metaboard.

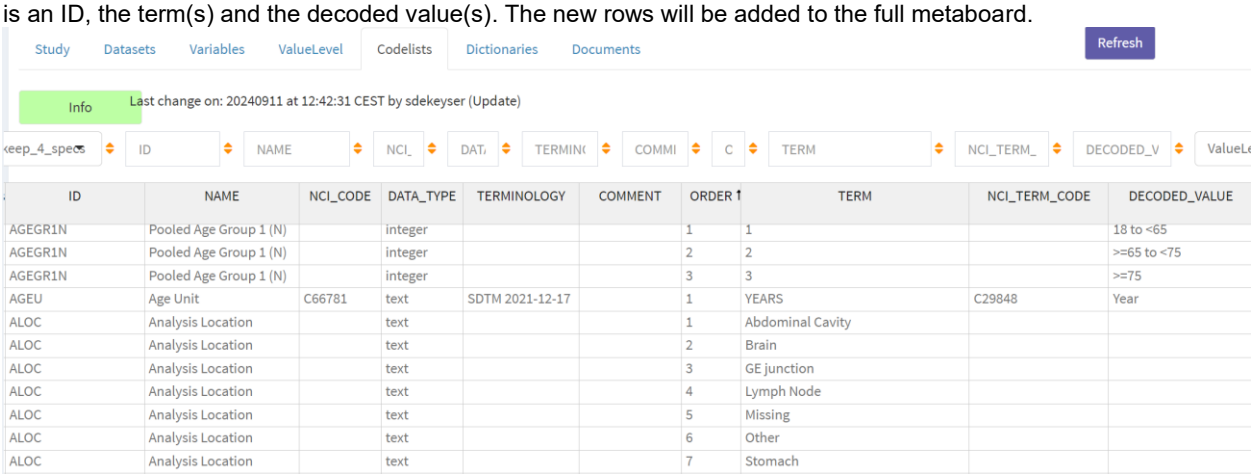



| FROM | keep_4_specs | ORDER | DATASET | VARIABLE | VAR_WHERE | COMPARATOR | VALUE | DESCRIPTION | LENGTH | SIGN_DIGIT | ASSIGNEDVALUE (only XML2.1) | CODELIST |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Codelist | ☑ | 1 | ADEG | AVALCAT1 | PARAMCD | EQ | QT | Analysis Value Category 1 | 34 | | | AVALCAT1_ADEG_PA -- |
| Codelist | ☑ | 2 | ADEG | AVALCAT1 | PARAMCD | EQ | QTCF | Analysis Value Category 1 | 34 | | | AVALCAT1_ADEG_PA -- |
| Codelist | ☑ | 1 | ADEG | CHGCAT1 | PARAMCD | EQ | QTCF | Change from Baseline Category 1 | 57 | | | CHGCAT1_ADEG_PA -- |
| Codelist | ☑ | 2 | ADEG | CHGCAT1 | PARAMCD | EQ | QT | Change from Baseline Category 1 | 57 | | | CHGCAT1_ADEG_PA -- |
| Codelist | ☑ | 1 | ADEG | CRIT2 | PARAMCD | EQ | QTCF | Analysis Criterion 2 | 52 | | | CRIT2_ADEG_PARA -- |
| Codelist | ☑ | 1 | ADEG | CRIT3 | PARAMCD | EQ | QT | Analysis Criterion 3 | 29 | | | CRIT3_ADEG_PARA -- |

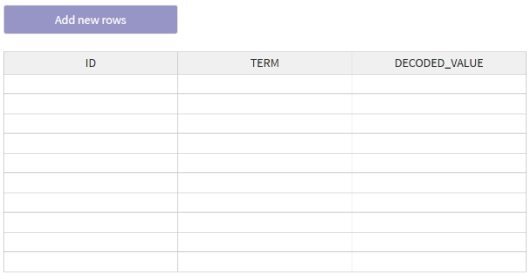
**METABOARD CODELISTS**

Regarding the codelists, a lot of them are prefilled by the %ADaM Macro.
However, some of them might not be needed (e.g. not really codelist). The column “delete” helps to clear these codelists. Because the metaboards are linked to one another, the codelist deleted will also be deleted in the “Variable” metaboard.

Codelists can also be manually added using the dedicated area at the bottom of the window. The required information is an ID, the term(s) and the decoded value(s). The new rows will be added to the full metaboard.

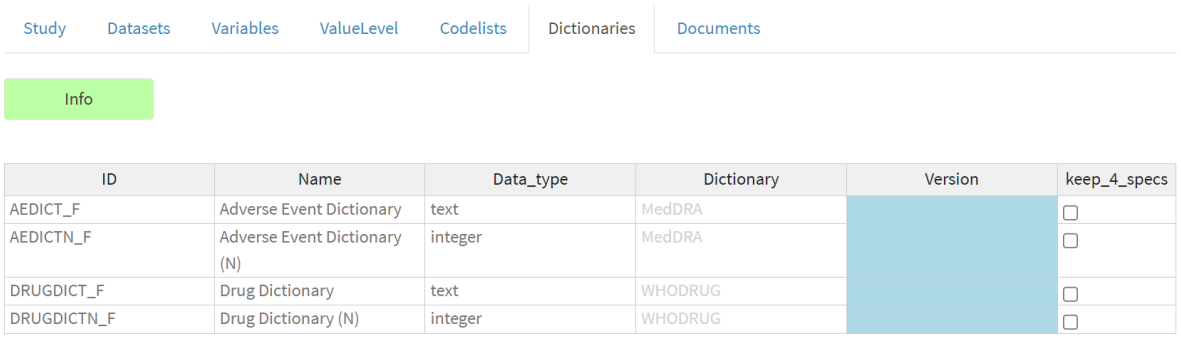



| ID | NAME | NCI_CODE | DATA_TYPE | TERMINOLOGY | COMMENT | ORDER | TERM | NCI_TERM_CODE | DECODED_VALUE |
|---|---|---|---|---|---|---|---|---|---|
| AGEGR1N | Pooled Age Group 1 (N) | | integer | | | 1 | 1 | | 18 to <65 |
| AGEGR1N | Pooled Age Group 1 (N) | | integer | | | 2 | 2 | | >=65 to <75 |
| AGEGR1N | Pooled Age Group 1 (N) | | integer | | | 3 | 3 | | >=75 |
| AGEU | Age Unit | C66781 | text | SDTM 2021-12-17 | | 1 | YEARS | C29848 | Year |
| ALOC | Analysis Location | | text | | | 1 | Abdominal Cavity | | |
| ALOC | Analysis Location | | text | | | 2 | Brain | | |
| ALOC | Analysis Location | | text | | | 3 | GE junction | | |
| ALOC | Analysis Location | | text | | | 4 | Lymph Node | | |
| ALOC | Analysis Location | | text | | | 5 | Missing | | |
| ALOC | Analysis Location | | text | | | 6 | Other | | |
| ALOC | Analysis Location | | text | | | 7 | Stomach | | |




**METABOARD DICTIONARIES**

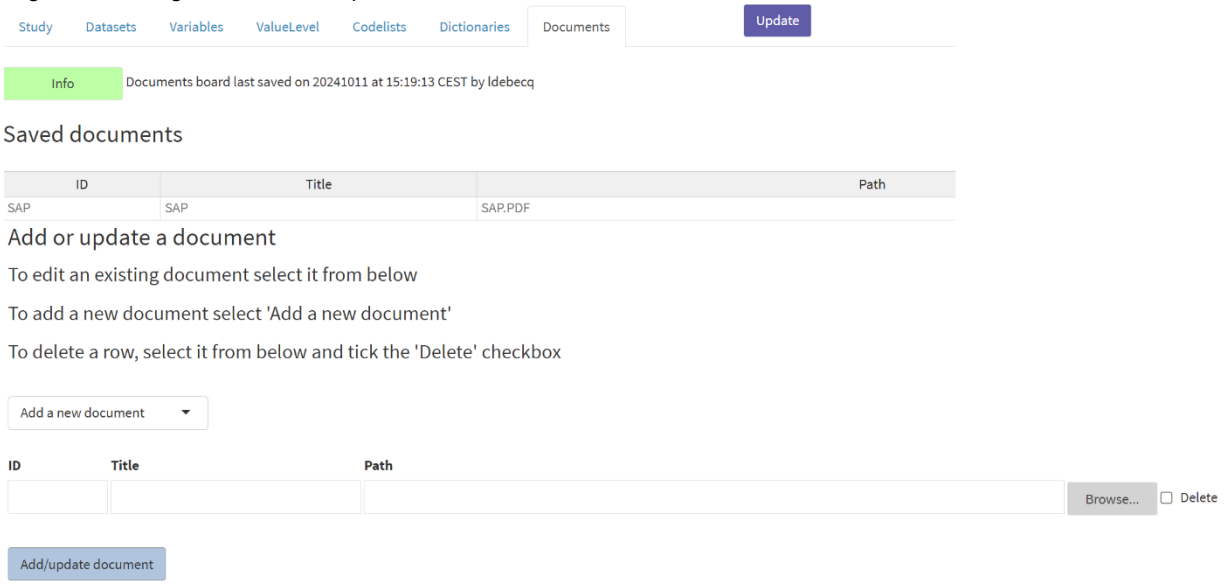
Depending on the Study, Dictionary for coding of medications, adverse events or procedures might have been used. That information is presented in the Dictionaries metaboard. The IDs are prefilled and contain data for AE/Procedures and Medications. Version is missing and needs to be added. As it requires user input, the cells are blue. "*keep_4_specs*" tick box (on the right) needs to be ticked if a dictionary should be in the specifications. By default, this is not included.




**METABOARD DOCUMENTS**

The last metaboard is for "Documents". All necessary information needs to be entered. Entering a document would create a reference "DOC.ID". That reference will be available in other metaboards for document referencing.

E.g. Referencing the SAP for a specific method.



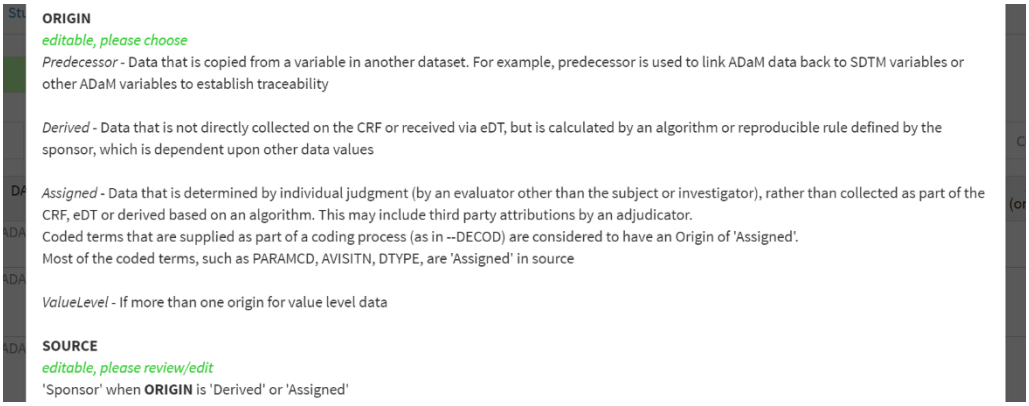
**MULTI-USER INTERFACE**


The system prevents modification by other team members while editing the specifications. There are two different states: “Read-only” and “Edition”.

The “Edition” mode will prevent any modification by anyone except the user who enabled the “Edition” mode, while the “Read-only” mode would allow any other team member to view and “Refresh” the data on screen, for each metaboard. ID-Spectackle will be locked for editing until the user who was in edition mode releases the system by switching to Read-only, which will also save any modification done

**USER-FRIENDLY INTERFACE**

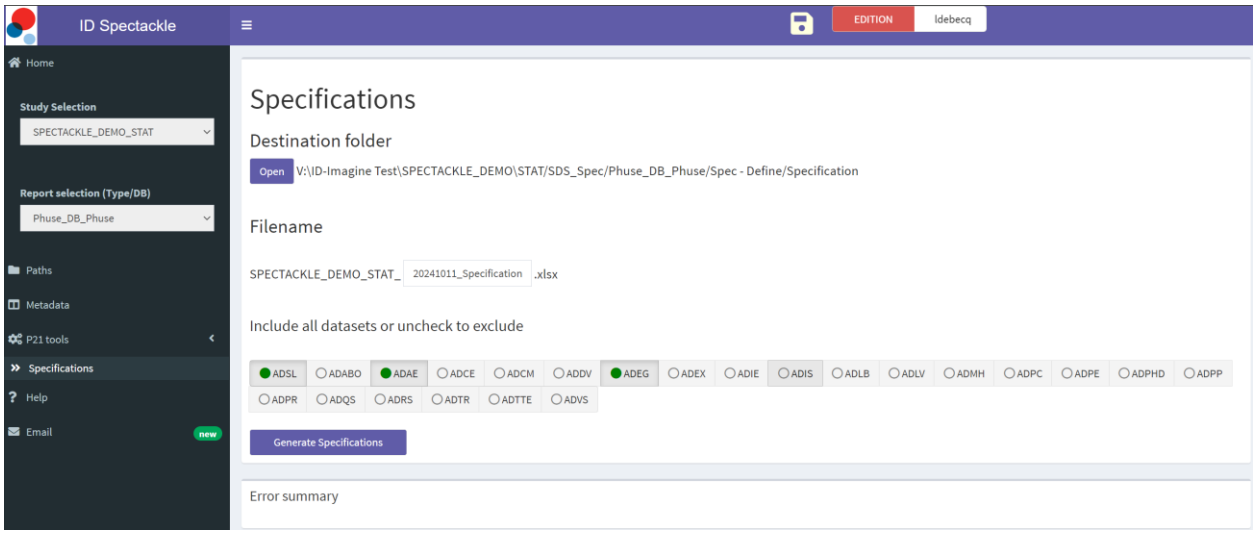
A nice feature of ID-Spectackle is that there is an info button on each metaboard. Clicking on this button, some guidance to complete the specifications will be accessible. Those recommendations ensure compliance with CDISC and P21. For example, for “Variables” metaboard, the info button will provide you with information about the “ORIGIN” options you can choose from: Predecessor, Derived, Assigned, and ValueLevel.




**CREATION OF SPECIFICATIONS IN EXCEL**

Once all metaboards are filled, then you can create the excel specifications using the button “Specifications” on the left side. It is possible to choose to create specifications in Excel for all analysis datasets or only for a part of them. This could be useful if you need, for example, to give specifications only for analysis datasets used for a key analysis report. In the example below, specifications will be created in Excel only for ADSL, ADAE and ADEG.

Such selection would also help during the validation process by enabling a short Define-XML, therefore a P21 validation on the selected datasets.



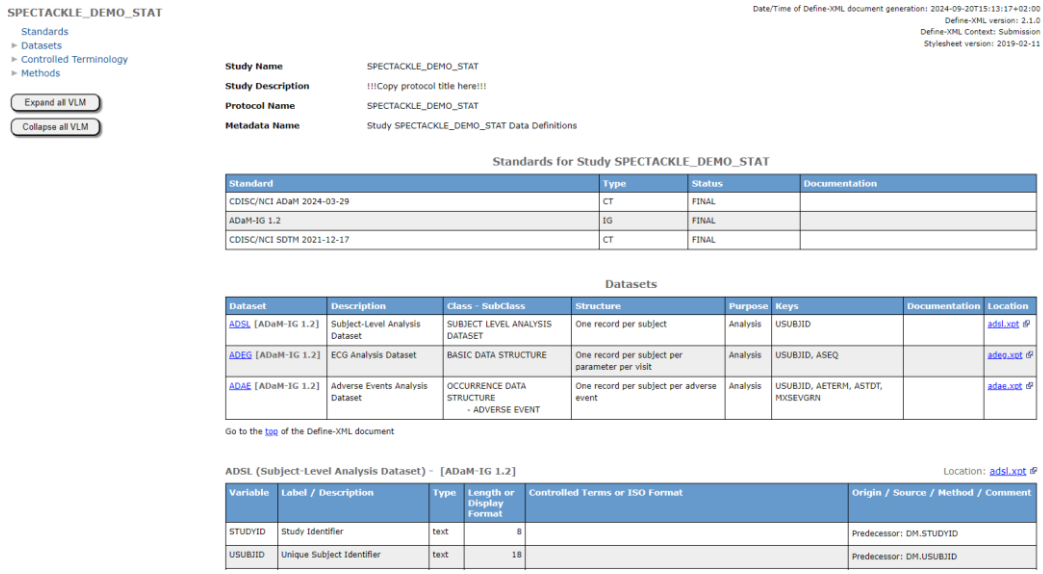

A summary of inconsistencies and errors identified in the specifications, if any, is displayed in the interface after the excel file generation.
In the excel specifications, there will be the expected sheets from the P21 template: Define, Datasets, Variables, ValueLevel, Codelists, Dictionaries, Methods, Comments and Documents.

| Dataset | Label | Class | SubClass | Structure |
|---|---|---|---|---|
| ADSL | Subject-Level Analysis Dataset | SUBJECT LEVEL ANALYSIS DATASET | | One record per subject |
| ADAE | Adverse Events Analysis Dataset | OCCURRENCE DATA STRUCTURE | ADVERSE EVENT | One record per subject |
| ADEG | ECG Analysis Dataset | BASIC DATA STRUCTURE | | One record per subject |

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

**DEFINE-XML USING P21**

Once excel specifications are created using ID-Spectackle, then Define-XML can be created using P21.




**SHORT DEMONSTRATION (QR CODE)**

Scanning the following QR Code, a short demonstration will be available on YouTube.

## CONCLUSION

ID-Spectackle is the combination of a macro and a web-interface. This combined approach for the specifications offers many advantages over the traditional approach.

The numerous benefits of our combined approach allow for a leaner, faster, easier and more standardized way of creating specifications and Define-XML.

It is less error-prone to user entry. Indeed, as an automated approach, all that is programmed is also in the metadata and therefore in the study specifications. The Macro ensures that all data/metadata are aligned with each other, while the user interface (1) allows for visual cues for items requiring a user input ; (2) provides inter-connection between metadatas i.e. Metadata are all connected to one another and any update on one is reflected on another; (3) specifications for metadata (variables, labels, length, etc.) contains exactly the derived dataset's metadata since analysis dataset and study metadata are created at the same time ; (4) helps to save time when updating specifications from newer data ; (5) ensures CDISC Compliance: Control Terminologies and ADaM IG and allows for upgrade of version ; (6) since the specifications produced by ID-Spectackle follow the standard template from P21, it provides us internal standardization and produces an understandable document that can be given to our sponsors, which improves the review process ; (7) improves the validation of datasets by allowing the extraction of partial metadata ; (8) speeds up production of necessary material for validation (i.e. Define-XML and XPT) ; (9) allows the reuse of specifications for similar studies with guarantee that it will contains adequate metadata ; (10) is user-friendly and requires very little training time ; (11) is FUN.

## REFERENCES

### Glossary

| | |
|---|---|
| IG | Implementation Guide |
| NCI | National Cancer Institute |
| WebUI | Web User Interface |
| CT | Controlled Terminology |
| CRF | Case Report Form |
| ADaM | Analysis Data Model |
| SDTM | Standard Data Tabulation Model |
| QC | Quality Control |
| SAP | Statistical Analysis Plan |
| P21 | Pinnacle 21 |

CDISC Define-XML 2.1 Specifications: https://www.cdisc.org/standards/data-exchange/define-xml/define-xml-v2-1
CDISC ADaM IG 1.3: https://www.cdisc.org/standards/foundational/adam/adamig-v1-3
CDISC SDTM IG: https://www.cdisc.org/standards/foundational/sdtmig
Pinnacle 21: https://www.pinnacle21.com
NCI FTP for CDISC Controlled Terminologies: https://evs.nci.nih.gov/ftp1/CDISC

## ACKNOWLEDGMENTS

We would like to thank IDDI and more specifically our colleagues of the Department of Biostatistics, our colleague Stefania Giammanco for her knowledge and experience in R for the maintenance of the interface, and our estimate colleagues' programmers who took the time to read and provide feedback on this paper.

## CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:
Lionel.debecq@iddi.com
Sophie.dekeyser@iddi.com