SDTM-ETL 5.0: Summary of New Features

Author: Jozef Aerts, XML4Pharma

Last update: **2025-01-25**

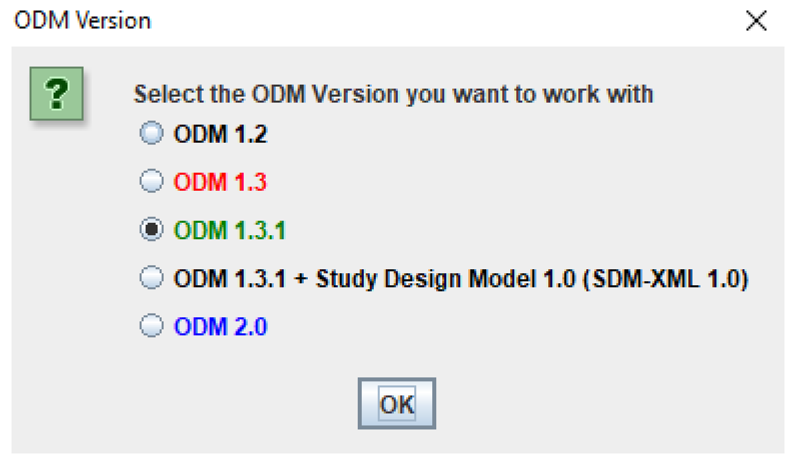
Summary

This document contains a summary of the most important new features of SDTM-ETL 5.0 and bug fixes.
There are many minor improvements and new features that are not described in this document, but that can be found in other manuals / tutorials of SDTM-ETL 5.0.

Table of Contents

Support for ODM-2.0

With some limitations, support has been added for the new CDISC ODM version 2.0 standard. When starting the software, the user is presented the following dialog, including the choice for ODM v.2.0:



ODM Version

Select the ODM Version you want to work with

○ ODM 1.2

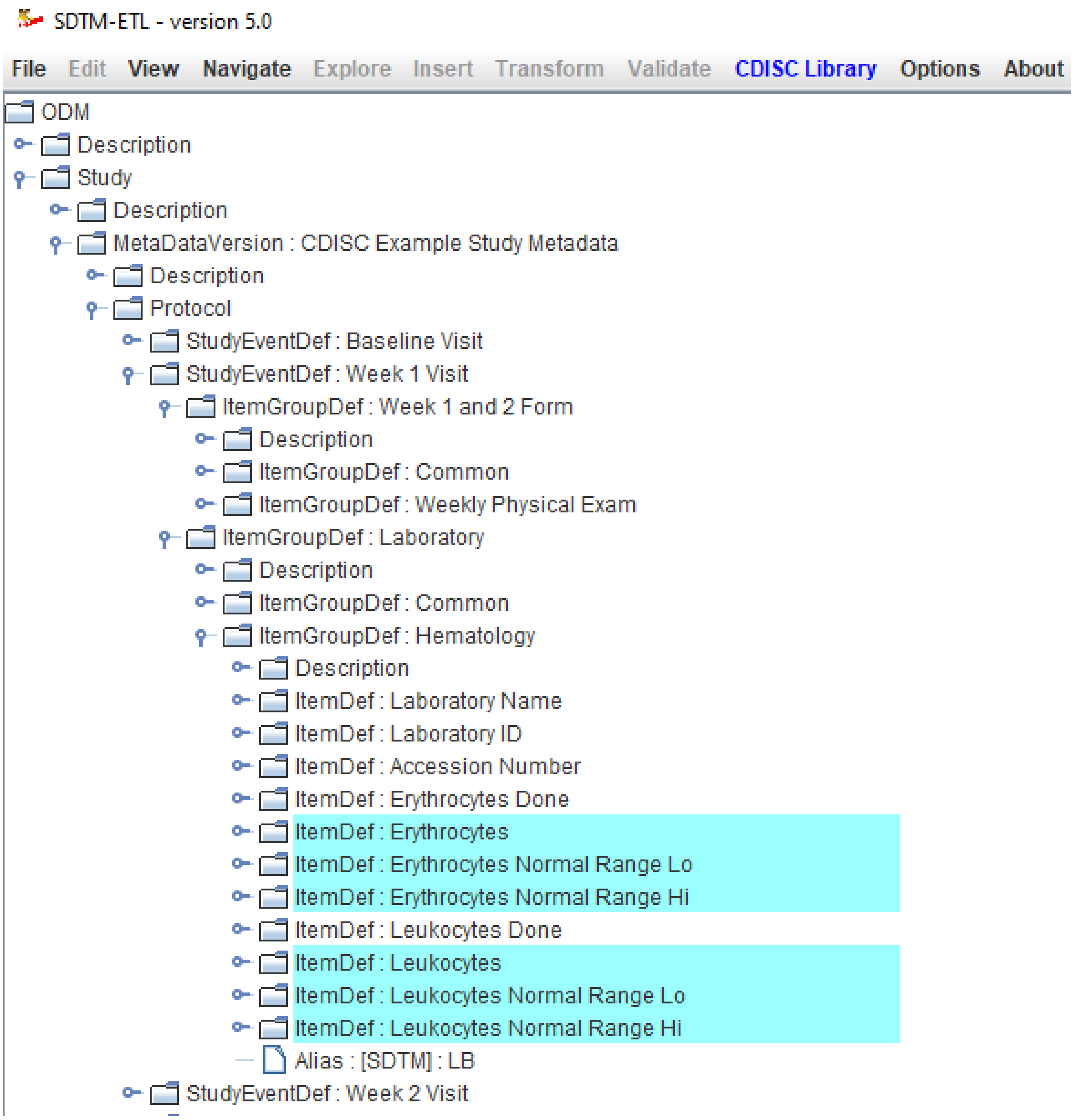
○ ODM 1.3

● ODM 1.3.1

○ ODM 1.3.1 + Study Design Model 1.0 (SDM-XML 1.0)

○ ODM 2.0

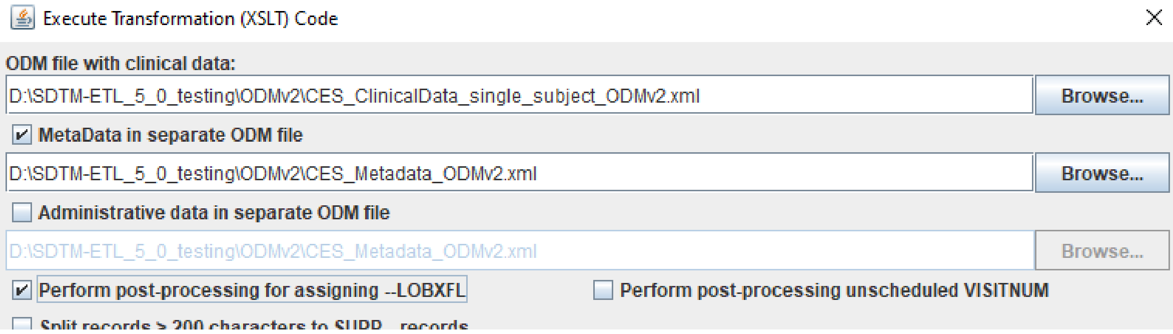
also allowing to select ODM version 2.0.
When selected, the user can then start loading an ODM v.2.0 file using the menu "File - Load ODM File", e.g. leading to:



SDTM-ETL - version 5.0

```
ODM
  Description
  Study
    Description
    MetaDataVersion : CDISC Example Study Metadata
      Description
      Protocol
        StudyEventDef : Baseline Visit
        StudyEventDef : Week 1 Visit
          ItemGroupDef : Week 1 and 2 Form
            Description
            ItemGroupDef : Common
            ItemGroupDef : Weekly Physical Exam
          ItemGroupDef : Laboratory
            Description
            ItemGroupDef : Common
            ItemGroupDef : Hematology
              Description
              ItemDef : Laboratory Name
              ItemDef : Laboratory ID
              ItemDef : Accession Number
              ItemDef : Erythrocytes Done
              ItemDef : Erythrocytes
              ItemDef : Erythrocytes Normal Range Lo
              ItemDef : Erythrocytes Normal Range Hi
              ItemDef : Leukocytes Done
              ItemDef : Leukocytes
              ItemDef : Leukocytes Normal Range Lo
              ItemDef : Leukocytes Normal Range Hi
              Alias : [SDTM] : LB
        StudyEventDef : Week 2 Visit
```

Also, when executing the developed mappings, an ODM v.2.0 file with clinical data should be used, e.g.


Execute Transformation (XSLT) Code

ODM file with clinical data:
D:\SDTM-ETL_5_0_testing\ODMv2\CES_ClinicalData_single_subject_ODMv2.xml

☑ MetaData in separate ODM file
D:\SDTM-ETL_5_0_testing\ODMv2\CES_Metadata_ODMv2.xml

□ Administrative data in separate ODM file
D:\SDTM-ETL_5_0_testing\ODMv2\CES_Metadata_ODMv2.xml

☑ Perform post-processing for assigning --LOBXFL   □ Perform post-processing unscheduled VISITNUM

□ Split records > 200 characters to SUPP-- records

<u>Limitations</u>:
- currently, validation of ODM v.2.0 files (upon loading the ODM file with metadata) is limited to XML-Schema validation. The part that checks on "ref-def" relations still need to be updated
- Support for ODM v.2.0 is currently limited to studies for which the study design has the classic structure "Visit" (ODM: StudyEvent), "Form" (ODM: ItemGroupDef/@Type='Form'), "Subform" (ODM: ItemGroupDef/@Type='Section'), "Question/Item" (ODM: ItemDef).
No support is currently provided for "nested" subforms, i.e. "ItemGroup" within "ItemGroup" that both are of type "Section". Nested subforms will be implemented in SDTM-ETL version 5.1.

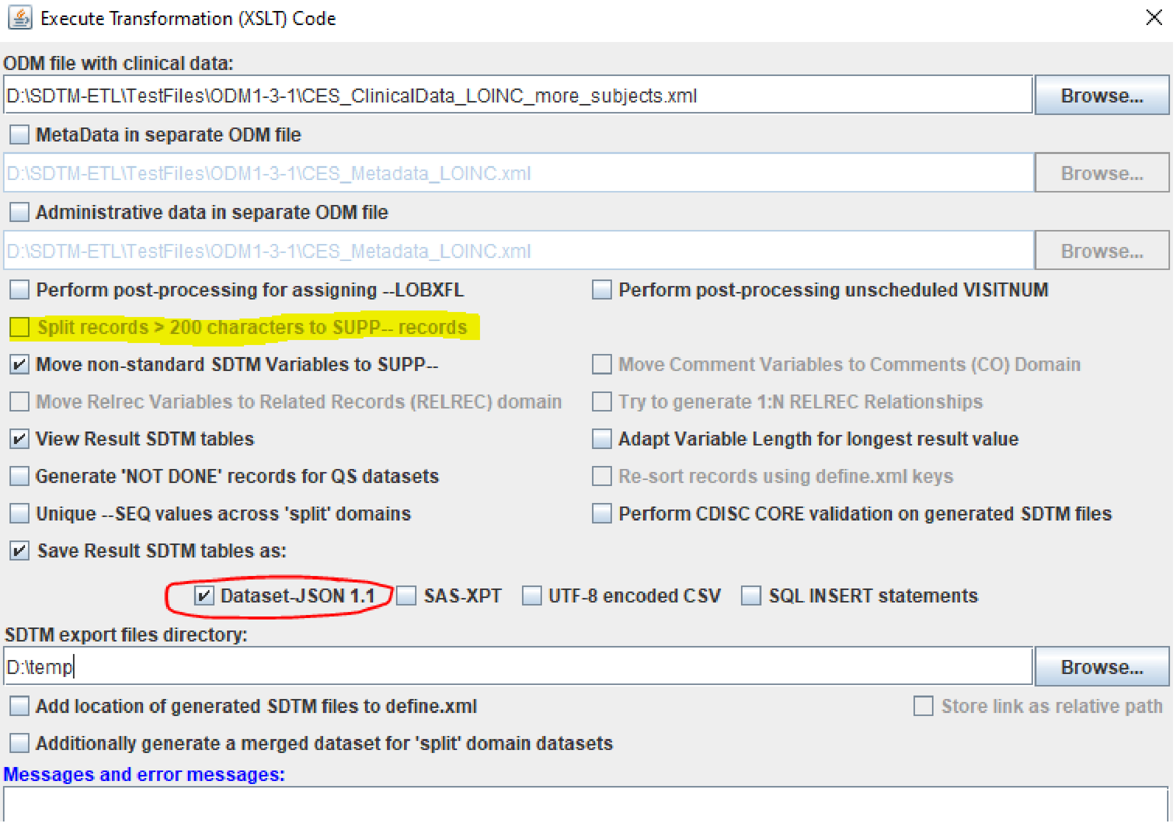
One single menu "Transform - Generate Transformation (XSLT) Code"

In previous version of SDTM-ETL, there were different menus for generation of the transformation code, one for SAS-XPT, one for Dataset-JSON, one for Dataset-XML and one for CSV (Comma-separated Values) format. This has now been replaced by a single menu "Transform - Generate Transformation (XSLT) Code", so that there is now only a single type of XSLT generated that can be used for different dataset formats. This has the additional advantage that when the XSLT is stored for "offline" processing (button "Save Transformation (XSLT) Code), it can be used to generate datasets in any of the supported formats.

Support for CDISC Dataset-JSON v.1.1

CDISC has developed an alternative for SAS Transport 5 ("XPT") format) as the standard for regulatory submission: Dataset-JSON 1.1, and successfully piloted it at the FDA. It is therefore expected that FDA, later followed by PMDA and NMPA, will include Dataset-JSON into the "standards catalog" in the course of 2025. The use of Dataset-JSON not only allows to overcome all the limitations of SAS Transport 5, the latter causing a lot of problems when generating SDTM, SEND and ADaM datasets, but will also enable to use APIs and RESTful web services (RWS) for exchange of datasets (or parts of them), essentially meaning that regulatory review can already be started when only part of the data is available, and can be finalized within short time after "last data point collected". See e.g. our article "The need for Speed" for further details and thoughts.

When executing the transformation of a set of developed mappings, the dialog will now show as:



Execute Transformation (XSLT) Code

ODM file with clinical data:

D:\SDTM-ETL\TestFiles\ODM1-3-1\CES_ClinicalData_LOINC_more_subjects.xml [Browse...]

□ MetaData in separate ODM file

D:\SDTM-ETL\TestFiles\ODM1-3-1\CES_Metadata_LOINC.xml [Browse...]

□ Administrative data in separate ODM file

D:\SDTM-ETL\TestFiles\ODM1-3-1\CES_Metadata_LOINC.xml [Browse...]

□ Perform post-processing for assigning --LOBXFL
□ Perform post-processing unscheduled VISITNUM
□ Split records > 200 characters to SUPP-- records
☑ Move non-standard SDTM Variables to SUPP--
□ Move Comment Variables to Comments (CO) Domain
□ Move Relrec Variables to Related Records (RELREC) domain
□ Try to generate 1:N RELREC Relationships
☑ View Result SDTM tables
□ Adapt Variable Length for longest result value
□ Generate 'NOT DONE' records for QS datasets
□ Re-sort records using define.xml keys
□ Unique --SEQ values across 'split' domains
□ Perform CDISC CORE validation on generated SDTM files
☑ Save Result SDTM tables as:

☑ Dataset-JSON 1.1 □ SAS-XPT □ UTF-8 encoded CSV □ SQL INSERT statements

SDTM export files directory:

D:\temp [Browse...]

□ Add location of generated SDTM files to define.xml
□ Store link as relative path
□ Additionally generate a merged dataset for 'split' domain datasets

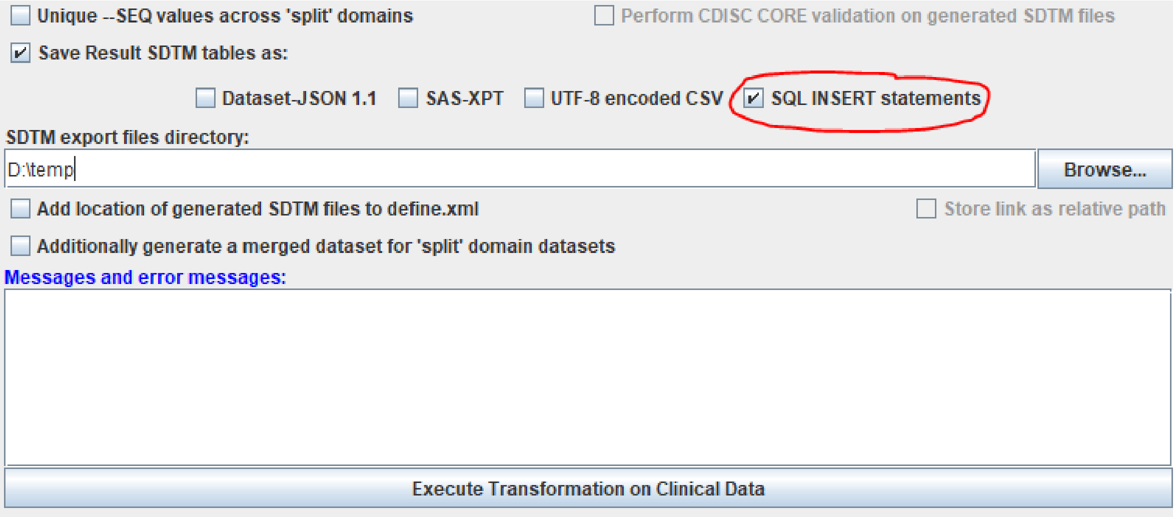
Messages and error messages:

Remark that when "Dataset-JSON 1.1" is selected as the format for the to be generated datasets, the checkbox "Split records > 200 characters to SUPP-- records" is automatically disabled, as unlike in SAS-XPT, Dataset-JSON does not have a limitation that values may not exceed 200 characters.

Direct Generation of "SQL Insert" statements and files

In previous versions of SDTM-ETL, "SQL Insert" statements could be generation indirectly by storing the datasets in the older "Dataset-XML" format, and then using the menu "Transform - Create 'SQL Insert' statements".

We have now replaced this by a direct transformation, by adding an additional option to the dialog in the last step of executing the transformation:



☐ Unique --SEQ values across 'split' domains

☐ Perform CDISC CORE validation on generated SDTM files

☑ Save Result SDTM tables as:

☐ Dataset-JSON 1.1 ☐ SAS-XPT ☐ UTF-8 encoded CSV ☑ SQL INSERT statements

SDTM export files directory: D:\temp

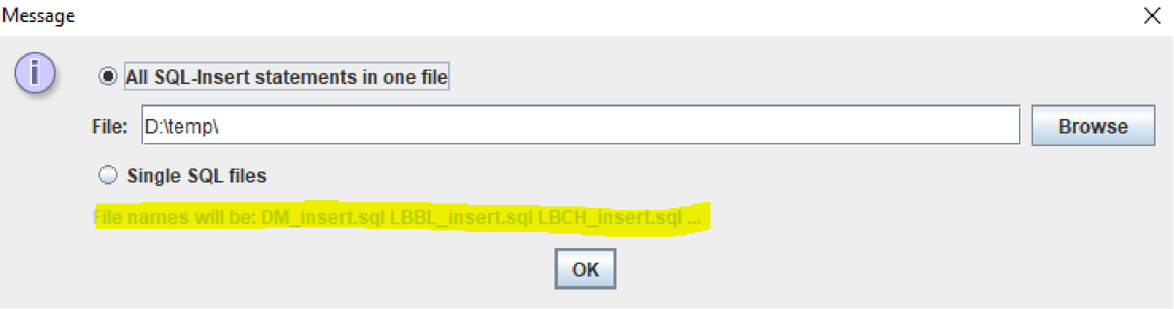
☐ Add location of generated SDTM files to define.xml

☐ Store link as relative path

☐ Additionally generate a merged dataset for 'split' domain datasets

Messages and error messages:

When then using the button "Execute Transformation …", near the end of the transformation, a dialog is being displayed:



Message

● All SQL-Insert statements in one file

File: D:\temp\

○ Single SQL files

File names will be: DM_insert.sql LBBL_insert.sql LBCH_insert.sql ...

allowing the user to choose between the creation of a single file with all SQL statements, or generating different files with SQL statements, one for each dataset involved.
In the first case, a full file name in the field "File:" needs to be provided. In the latter case, only a folder is expected and the separate files will be named according to the dataset name.

Discontinuation of support for Dataset-XML format

Now that it is clear that FDA will go for Dataset-JSON as an alternative for SAS-XPT, there is no choice anymore for Dataset-XML (a precursor of Dataset-JSON) as the output format. Users who however would still like to have the option for Dataset-XML output, can ask for a tailored version (no additional cost).

Dataset-JSON Viewer

In the past (before Dataset-JSON support was added), when users generated SDTM or SEND datasets in Dataset-XML format, there was the option to use the "Smart Submission Dataset Viewer" to inspect the generated Dataset-XML formatted datasets.
Now that Dataset-XML support has been replaced by Dataset-JSON 1.1 support, it is the question whether we still need or want to provide the "Smart Submission Dataset Viewer" together with the SDTM-ETL software, this although the newest version of the "Smart Submission Dataset Viewer" now fully supports Dataset-JSON 1.1.

For the moment, we decided not to do so, as, as part of the "Dataset-JSON Hackathon", a good number of "viewers" for Dataset-JSON has been made available by different authors, and we expect many more offerings (mostly "open source") to come.
As also a good number of articles and scripts are now available to load Dataset-JSON files in software like SAS and R, we have decided to encourage the users to make the choice which viewer to use themselves. This means that for the moment, we do not deliver the "Smart Submission Dataset Viewer" together with the software anymore.
Users can of course download the free and open-source "Smart Submission Dataset Viewer" from the SourceForge website, and use it to inspect the generated Dataset-JSON datasets.

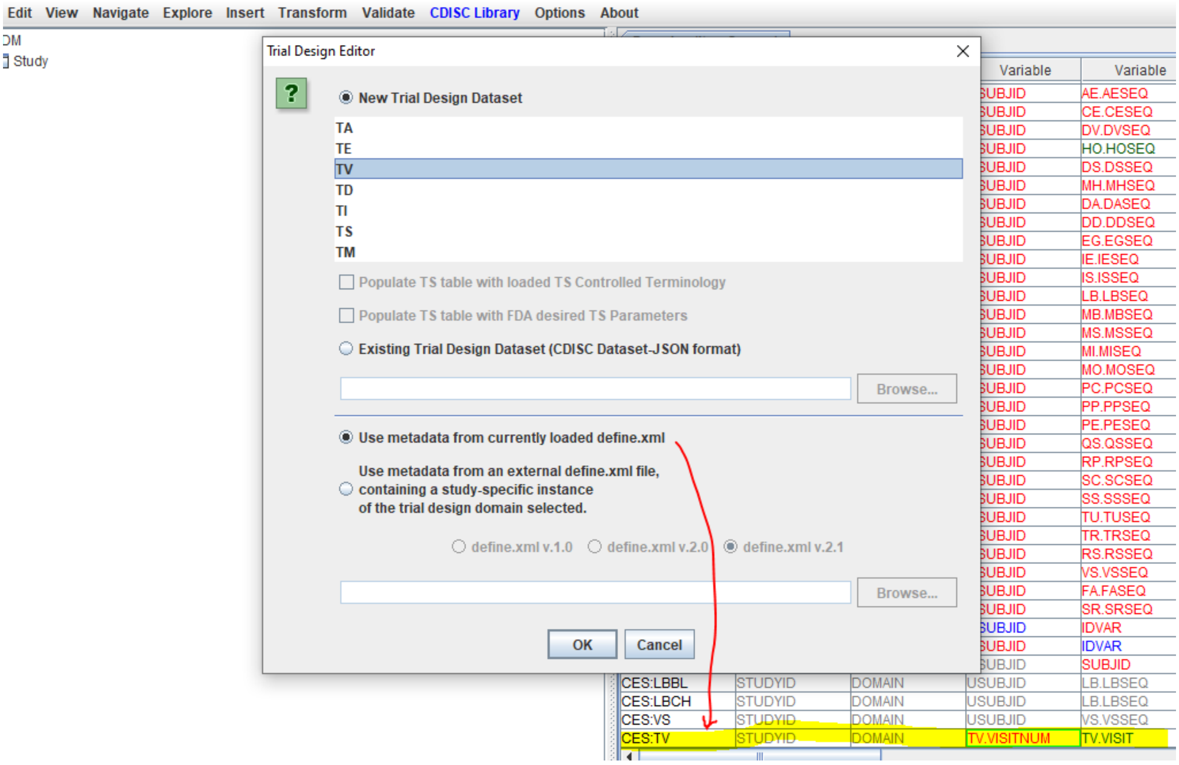
For open-source software supporting the Dataset-JSON format, see the CDISC Open Source Alliance (COSA) website.

Update of the Trial Design Editor

When using the "Trial Design Editor", which can also be started separately by a double-click "Start_TrialDesign_Editor.bat", one can store the generated trial design dataset either as SAS-XPT or as Dataset-JSON. New is that version 1.1 of Dataset-JSON is used instead of v.1.0.

This Dataset-JSON should be used when one expects that the dataset is not complete, and will later need to be reloaded for further editing / adaption / extension.

For example, when one wants to start developing the TV (Trial Visits) datasets, one will first create a "study-specific instance" of TV in SDTM-ETL, then use the menu "Edit - Trial Design Dataset" and select "Use metadata from currently loaded define.xml":



Trial Design Editor

● New Trial Design Dataset

TA
TE
TV
TD
TI
TS
TM

□ Populate TS table with loaded TS Controlled Terminology

□ Populate TS table with FDA desired TS Parameters

○ Existing Trial Design Dataset (CDISC Dataset-JSON format)

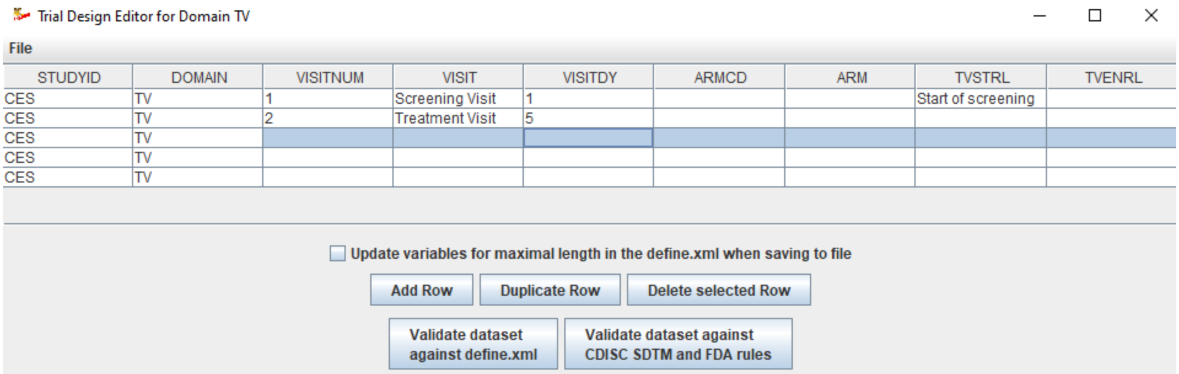
● Use metadata from currently loaded define.xml

○ Use metadata from an external define.xml file, containing a study-specific instance of the trial design domain selected.

○ define.xml v.1.0 ○ define.xml v.2.0 ● define.xml v.2.1

			Variable	Variable
			SUBJID	AE.AESEQ
			SUBJID	CE.CESEQ
			SUBJID	DV.DVSEQ
			SUBJID	HO.HOSEQ
			SUBJID	DS.DSSEQ
			SUBJID	MH.MHSEQ
			SUBJID	DA.DASEQ
			SUBJID	DD.DDSEQ
			SUBJID	EG.EGSEQ
			SUBJID	IE.IESEQ
			SUBJID	IS.ISSEQ
			SUBJID	LB.LBSEQ
			SUBJID	MB.MBSEQ
			SUBJID	MS.MSSEQ
			SUBJID	MI.MISEQ
			SUBJID	MO.MOSEQ
			SUBJID	PC.PCSEQ
			SUBJID	PP.PPSEQ
			SUBJID	PE.PESEQ
			SUBJID	QS.QSSEQ
			SUBJID	RP.RPSEQ
			SUBJID	SC.SCSEQ
			SUBJID	SS.SSSEQ
			SUBJID	TU.TUSEQ
			SUBJID	TR.TRSEQ
			SUBJID	RS.RSSEQ
			SUBJID	VS.VSSEQ
			SUBJID	FA.FASEQ
			SUBJID	SR.SRSEQ
			SUBJID	IDVAR
			SUBJID	IDVAR
			SUBJID	SUBJID
CES:LBBL	STUDYID	DOMAIN	USUBJID	LB.LBSEQ
CES:LBCH	STUDYID	DOMAIN	USUBJID	LB.LBSEQ
CES:VS	STUDYID	DOMAIN	USUBJID	VS.VSSEQ
CES:TV	STUDYID	DOMAIN	TV.VISITNUM	TV.VISIT

and then click "OK", leading to an empty table, which one can then starts filling, e.g.:

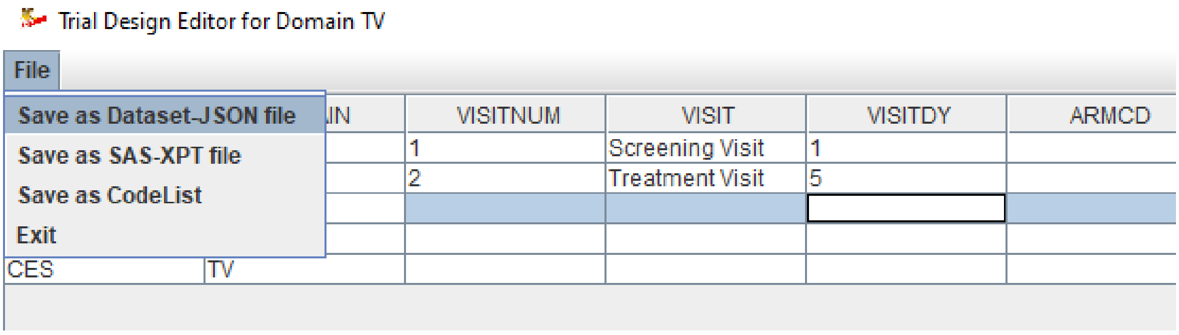

Trial Design Editor for Domain TV
File

STUDYID	DOMAIN	VISITNUM	VISIT	VISITDY	ARMCD	ARM	TVSTRL	TVENRL
CES	TV	1	Screening Visit	1			Start of screening	
CES	TV	2	Treatment Visit	5				
CES	TV							
CES	TV							
CES	TV							

☐ Update variables for maximal length in the define.xml when saving to file

Add Row
Duplicate Row
Delete selected Row
Validate dataset against define.xml
Validate dataset against CDISC SDTM and FDA rules

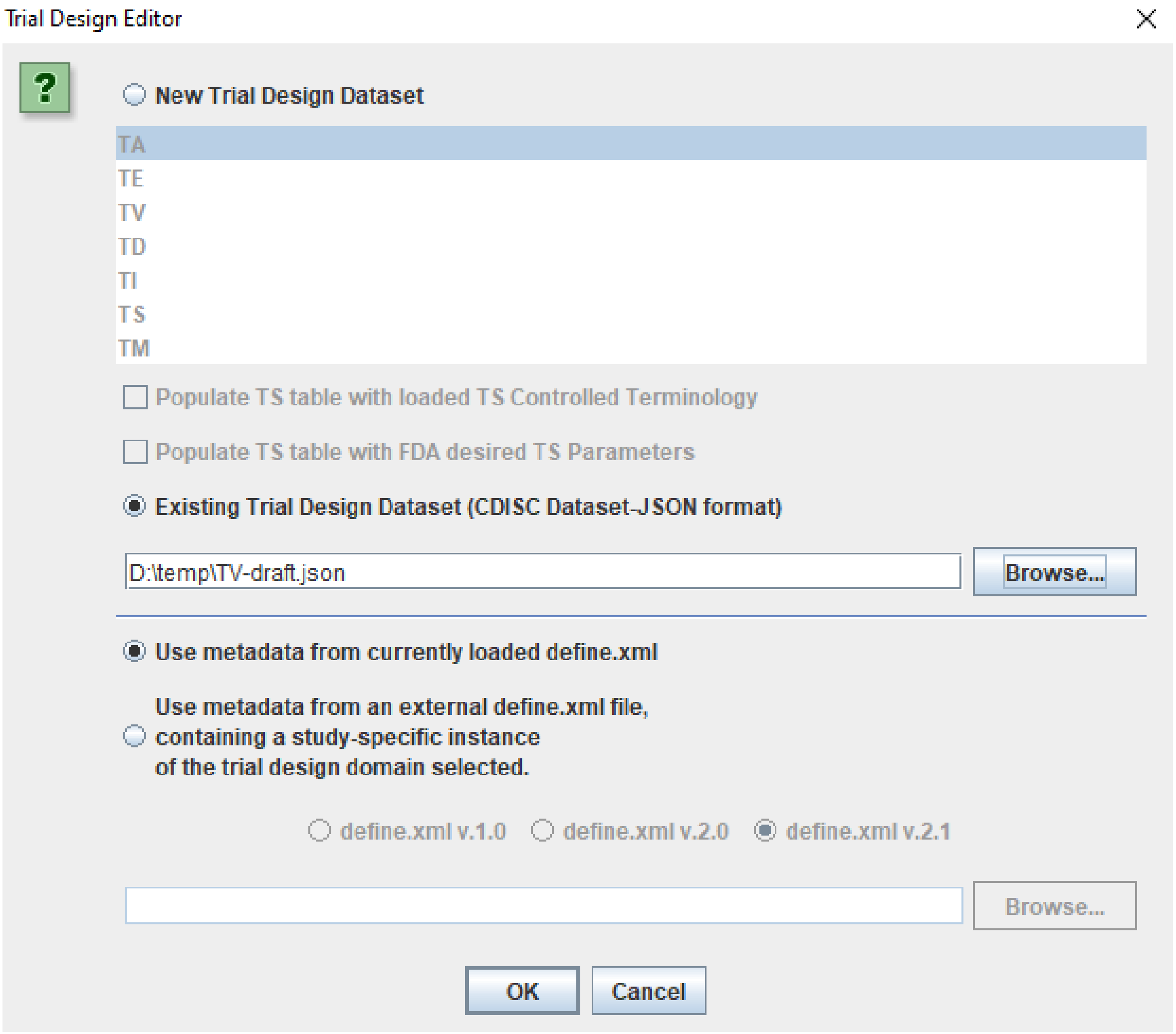
If one then wants to complete the table later, one should use the menu "File - Save as Dataset-JSON File":



	IN	VISITNUM	VISIT	VISITDY	ARMCD
		1	Screening Visit	1	
		2	Treatment Visit	5	
CES	TV				

and store it as a ".json" file (e.g. "TV_draft.json") in a folder of your choice.

When then later working on the TV dataset, use ""Existing Trial Design Dataset …"



○ New Trial Design Dataset

TA
TE
TV
TD
TI
TS
TM

□ Populate TS table with loaded TS Controlled Terminology

□ Populate TS table with FDA desired TS Parameters

● Existing Trial Design Dataset (CDISC Dataset-JSON format)

D:\temp\TV-draft.json

● Use metadata from currently loaded define.xml

○ Use metadata from an external define.xml file, containing a study-specific instance of the trial design domain selected.

○ define.xml v.1.0 ○ define.xml v.2.0 ● define.xml v.2.1

which will load the table again so that one can continue working on it.

Use "File - Save as SAS-XPT File" only in case you are sure your TV dataset is complete and you still need to do your submission in XPT format. Remark that the SDTM-ETL software has no capability to import XPT files, as the XPT format is mostly propriety.
In case your target regulatory is already accepting Dataset-JSON 1.1, you can just submit your completed Dataset-JSON file, e.g. as "TV.json".
Also remark that in this case, there is no need for "optimizing" your file "for size", as, unlike XPT format, Dataset-JSON is always already very compact and optimized.

Generate and merge RELREC datasets

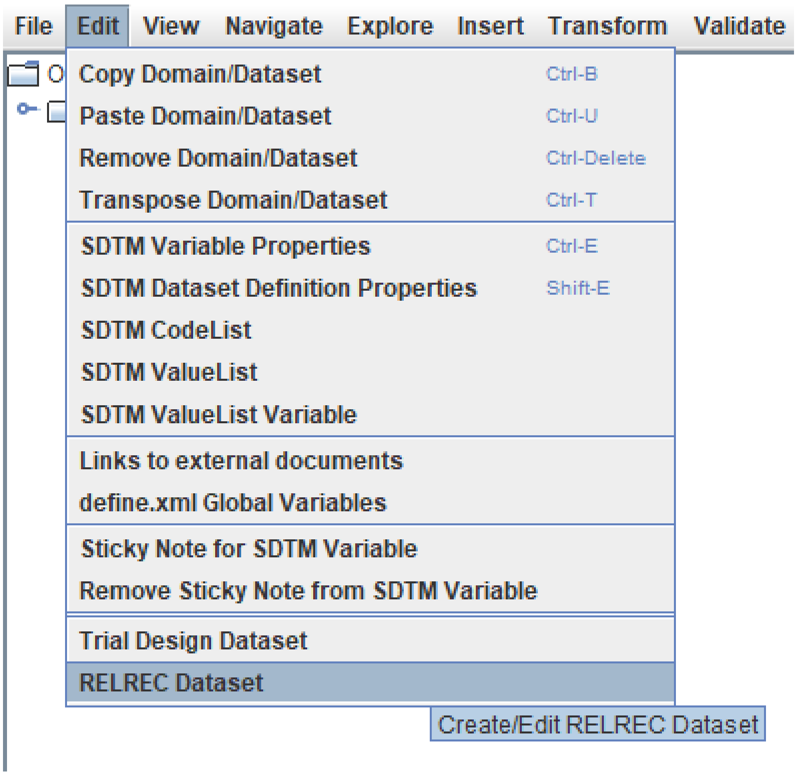
RELREC is a somewhat special domain, "pushing" relations between variables in a flat table[1]. Even more strange is that it mixes relations for individual records (e.g. between AE, CM, and LB) with those for "generic" dataset relations, so-called "one-many" relations.
This makes it rather difficult to handle by software when trying to automate.

Therefore we added a new feature to create RELREC datasets as a table and fill it manually (usually for generating the "one-many" dataset relations), and then having the possibility to

[1] We consider RELREC as a major design error, but yes, "if you only have SAS-XPT, everything is a table" ...

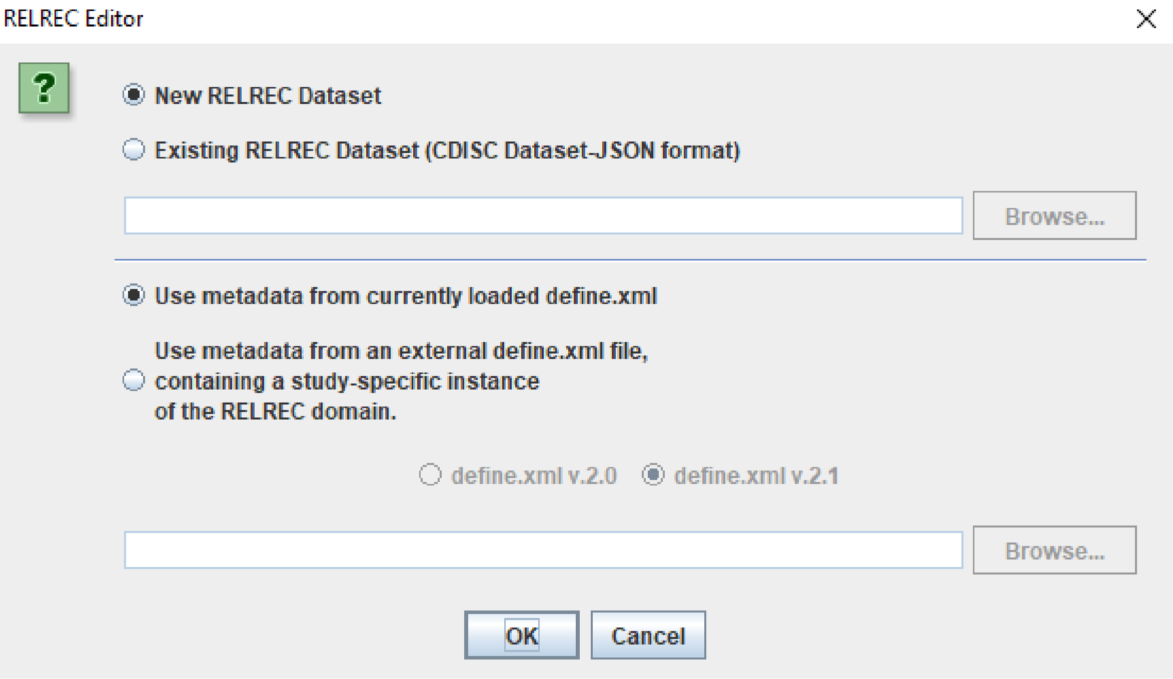
merge with an existing RELREC dataset (usually with record relations). The result can then be saved in the new CDISC Dataset-JSON-1.1 format, which is expected to be accepted soon by FDA, or the "classic" XPT format.
Even when saving into XPT format, it is also recommended by save the generated/merged RELREC into Dataset-JSON-1.1 format, as the latter will allow to make later changes.
For this, the RELREC editor allows to read existing Dataset-JSON-1.1 RELREC datasets, but not XPT datasets[2].

The new RELREC editor can be started using the menu "Edit - Edit RELREC dataset":


File
Edit
View
Navigate
Explore
Insert
Transform
Validate
Copy Domain/Dataset Ctrl-B
Paste Domain/Dataset Ctrl-U
Remove Domain/Dataset Ctrl-Delete
Transpose Domain/Dataset Ctrl-T
SDTM Variable Properties Ctrl-E
SDTM Dataset Definition Properties Shift-E
SDTM CodeList
SDTM ValueList
SDTM ValueList Variable
Links to external documents
define.xml Global Variables
Sticky Note for SDTM Variable
Remove Sticky Note from SDTM Variable
Trial Design Dataset
RELREC Dataset
Create/Edit RELREC Dataset

leading to a dialog:

[2] Reason is that SAS-XPT is essentially "semi-propriety format", so "not open". Also, we expect that regulatory authorities will soon start accepting Dataset-JSON format, and deprecate SAS-XPT. This would especially be beneficial for authorities where submissions can contain non-English information (Spanish, Japanese, Chinese, Korea, …)



RELREC Editor

● New RELREC Dataset

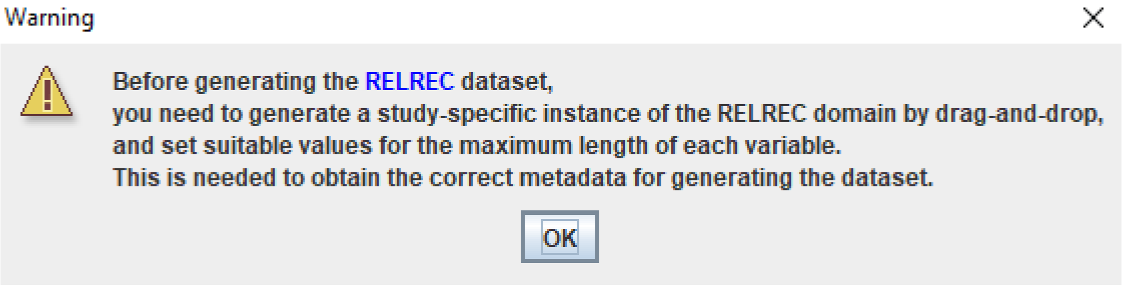
○ Existing RELREC Dataset (CDISC Dataset-JSON format)

● Use metadata from currently loaded define.xml

○ Use metadata from an external define.xml file, containing a study-specific instance of the RELREC domain.

○ define.xml v.2.0 ● define.xml v.2.1

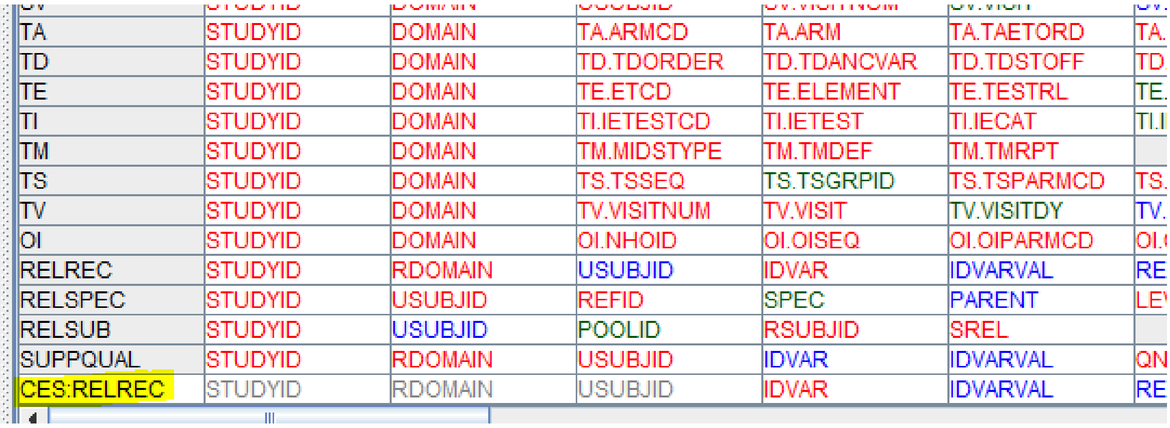
allowing to start a new RELREC dataset, or start from an existing one in Dataset-JSON-1.1 format. When using SDTM-ETL, one will usually also want to use the metadata from the currently loaded define". When no study-specific instance has been created yet for RELREC in the current session, continuing with "OK" will lead to a warning message:



Warning

Before generating the RELREC dataset,
you need to generate a study-specific instance of the RELREC domain by drag-and-drop,
and set suitable values for the maximum length of each variable.
This is needed to obtain the correct metadata for generating the dataset.

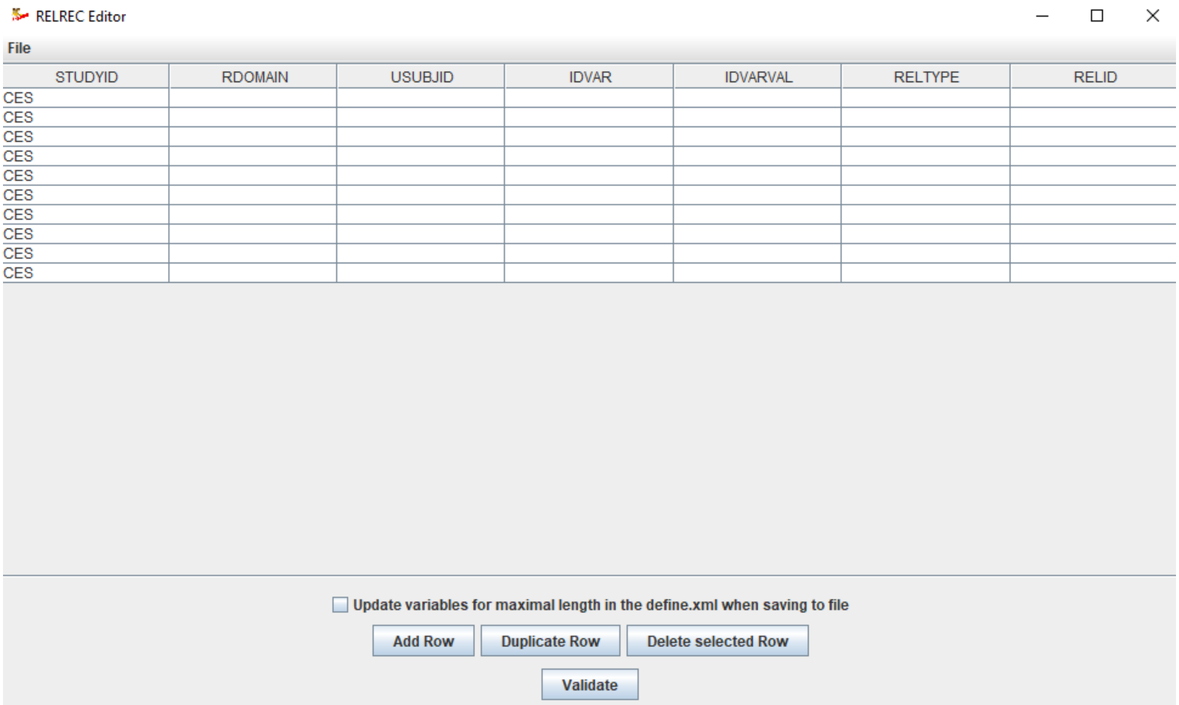
In case one would go back, and create a study-specific instance of RELREC simply by "drag-and-drop", e.g.:



SV	STUDYID	DOMAIN	USUBJID	SV.VISITNUM	SV.VISIT	SV.
TA	STUDYID	DOMAIN	TA.ARMCD	TA.ARM	TA.TAETORD	TA.
TD	STUDYID	DOMAIN	TD.TDORDER	TD.TDANCVAR	TD.TDSTOFF	TD.
TE	STUDYID	DOMAIN	TE.ETCD	TE.ELEMENT	TE.TESTRL	TE.
TI	STUDYID	DOMAIN	TI.IETESTCD	TI.IETEST	TI.IECAT	TI.I
TM	STUDYID	DOMAIN	TM.MIDSTYPE	TM.TMDEF	TM.TMRPT	
TS	STUDYID	DOMAIN	TS.TSSEQ	TS.TSGRPID	TS.TSPARMCD	TS.
TV	STUDYID	DOMAIN	TV.VISITNUM	TV.VISIT	TV.VISITDY	TV.
OI	STUDYID	DOMAIN	OI.NHOID	OI.OISEQ	OI.OIPARMCD	OI.
RELREC	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RE
RELSPEC	STUDYID	USUBJID	REFID	SPEC	PARENT	LE
RELSUB	STUDYID	USUBJID	POOLID	RSUBJID	SREL	
SUPPQUAL	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	QN
CES:RELREC	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RE

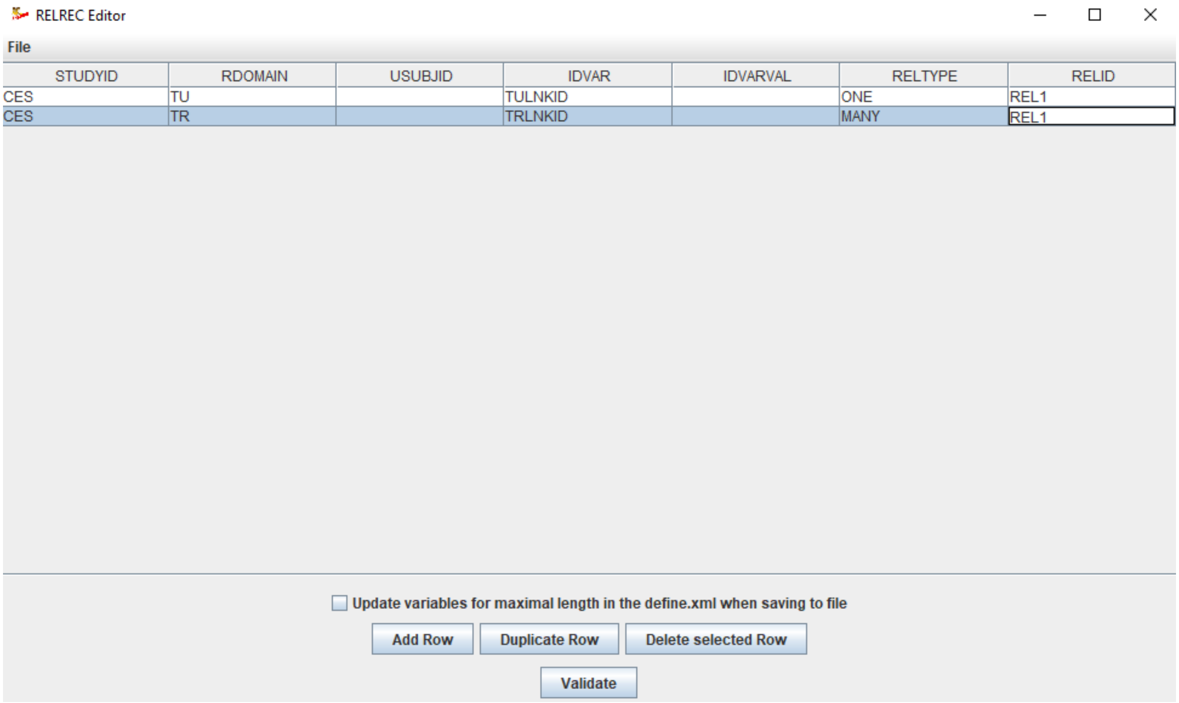
One can then edit the properties, like setting different lengths, adding comments, …

Once we have a study-specific instance of RELREC, the dialog with "New RELREC", after clicking "OK" will lead to a editable table:


RELREC Editor
File
☐ Update variables for maximal length in the define.xml when saving to file
Add Row
Duplicate Row
Delete selected Row
Validate

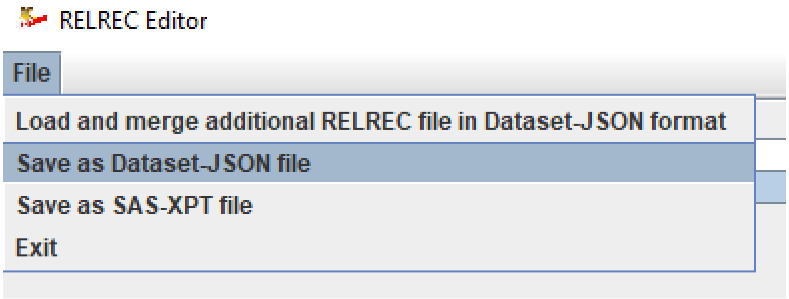
STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RELTYPE	RELID
CES						
CES						
CES						
CES						
CES						
CES						
CES						
CES						
CES						
CES						

which one can then start filling, usually with "one-many" domain relationships. For example:

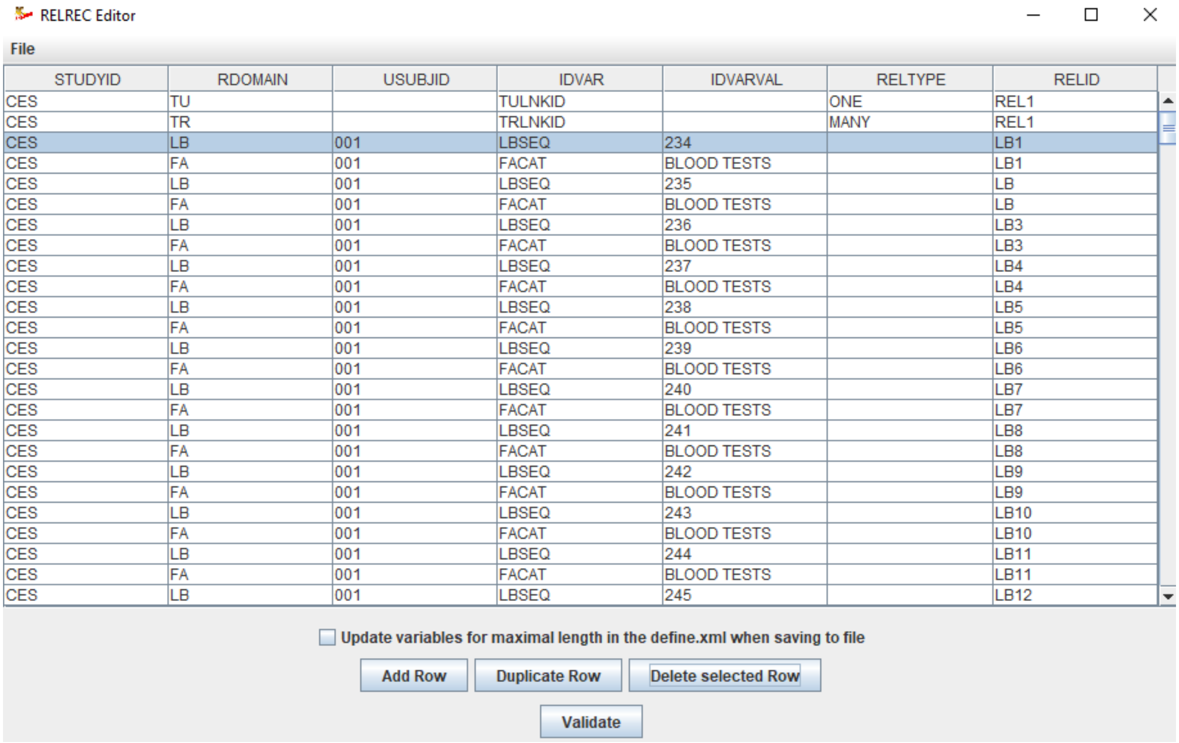

RELREC Editor
File
☐ Update variables for maximal length in the define.xml when saving to file
Add Row
Duplicate Row
Delete selected Row
Validate

STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RELTYPE	RELID
CES	TU		TULNKID		ONE	REL1
CES	TR		TRLNKID		MANY	REL1

If one then wants to continue working on the dataset later, or merge it with automatically generated "record" RELREC datasets (see the tutorial "Automatically creating related records (RELREC datasets)"), one can then save the dataset as Dataset-JSON-1.1.
Once final, one can also save the dataset in SAS-XPT format:


RELREC Editor
File
Load and merge additional RELREC file in Dataset-JSON format
Save as Dataset-JSON file
Save as SAS-XPT file
Exit

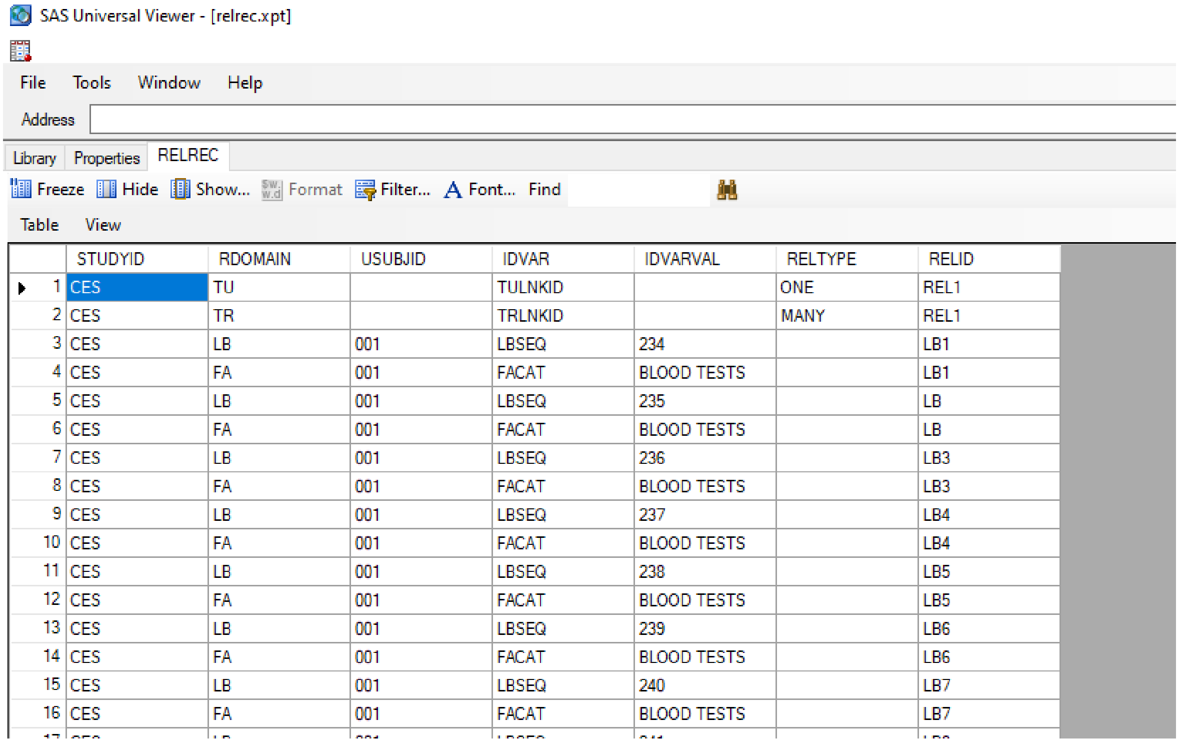
When then wants to merge with an automatically generated "record-based" RELREC dataset, one uses "Load and merge …", e.g. leading to:



STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RELTYPE	RELID
CES	TU		TULNKID		ONE	REL1
CES	TR		TRLNKID		MANY	REL1
CES	LB	001	LBSEQ	234		LB1
CES	FA	001	FACAT	BLOOD TESTS		LB1
CES	LB	001	LBSEQ	235		LB
CES	FA	001	FACAT	BLOOD TESTS		LB
CES	LB	001	LBSEQ	236		LB3
CES	FA	001	FACAT	BLOOD TESTS		LB3
CES	LB	001	LBSEQ	237		LB4
CES	FA	001	FACAT	BLOOD TESTS		LB4
CES	LB	001	LBSEQ	238		LB5
CES	FA	001	FACAT	BLOOD TESTS		LB5
CES	LB	001	LBSEQ	239		LB6
CES	FA	001	FACAT	BLOOD TESTS		LB6
CES	LB	001	LBSEQ	240		LB7
CES	FA	001	FACAT	BLOOD TESTS		LB7
CES	LB	001	LBSEQ	241		LB8
CES	FA	001	FACAT	BLOOD TESTS		LB8
CES	LB	001	LBSEQ	242		LB9
CES	FA	001	FACAT	BLOOD TESTS		LB9
CES	LB	001	LBSEQ	243		LB10
CES	FA	001	FACAT	BLOOD TESTS		LB10
CES	LB	001	LBSEQ	244		LB11
CES	FA	001	FACAT	BLOOD TESTS		LB11
CES	LB	001	LBSEQ	245		LB12

☐ Update variables for maximal length in the define.xml when saving to file

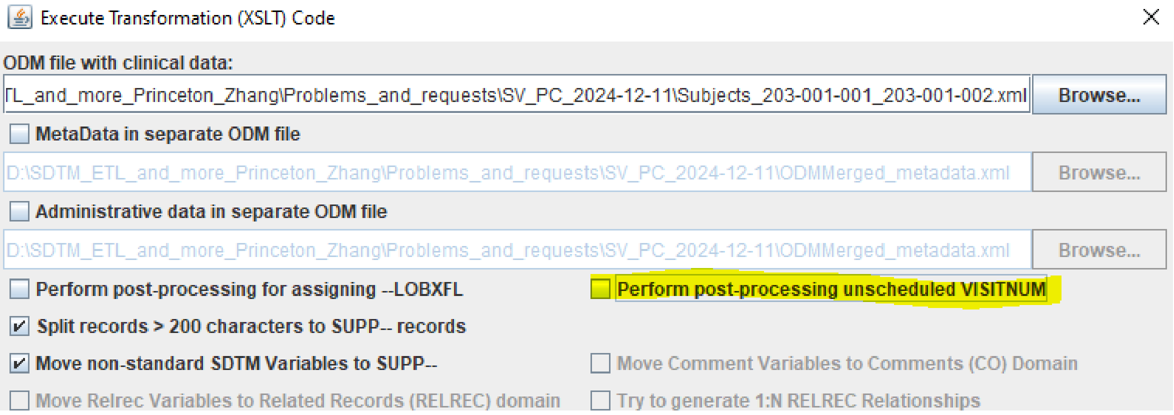
which can be saved as Dataset-JSON again for later editing or merging, or when final, as SAS-XPT, leading to the XPT file as:



	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RELTYPE	RELID
1	CES	TU		TULNKID		ONE	REL1
2	CES	TR		TRLNKID		MANY	REL1
3	CES	LB	001	LBSEQ	234		LB1
4	CES	FA	001	FACAT	BLOOD TESTS		LB1
5	CES	LB	001	LBSEQ	235		LB
6	CES	FA	001	FACAT	BLOOD TESTS		LB
7	CES	LB	001	LBSEQ	236		LB3
8	CES	FA	001	FACAT	BLOOD TESTS		LB3
9	CES	LB	001	LBSEQ	237		LB4
10	CES	FA	001	FACAT	BLOOD TESTS		LB4
11	CES	LB	001	LBSEQ	238		LB5
12	CES	FA	001	FACAT	BLOOD TESTS		LB5
13	CES	LB	001	LBSEQ	239		LB6
14	CES	FA	001	FACAT	BLOOD TESTS		LB6
15	CES	LB	001	LBSEQ	240		LB7
16	CES	FA	001	FACAT	BLOOD TESTS		LB7

More features for Unscheduled Visits calculations

For the case of "unscheduled" visits, one can either calculate the VISITNUM (which then is a fractional number) by adding a script for it (just like for any other variable), or by using the option "Perform post-processing unscheduled VISITNUM":



ODM file with clinical data:

TL_and_more_Princeton_Zhang\Problems_and_requests\SV_PC_2024-12-11\Subjects_203-001-001_203-001-002.xml

□ MetaData in separate ODM file

D:\SDTM_ETL_and_more_Princeton_Zhang\Problems_and_requests\SV_PC_2024-12-11\ODMMerged_metadata.xml

□ Administrative data in separate ODM file

D:\SDTM_ETL_and_more_Princeton_Zhang\Problems_and_requests\SV_PC_2024-12-11\ODMMerged_metadata.xml

□ Perform post-processing for assigning --LOBXFL

□ Perform post-processing unscheduled VISITNUM

☑ Split records > 200 characters to SUPP-- records

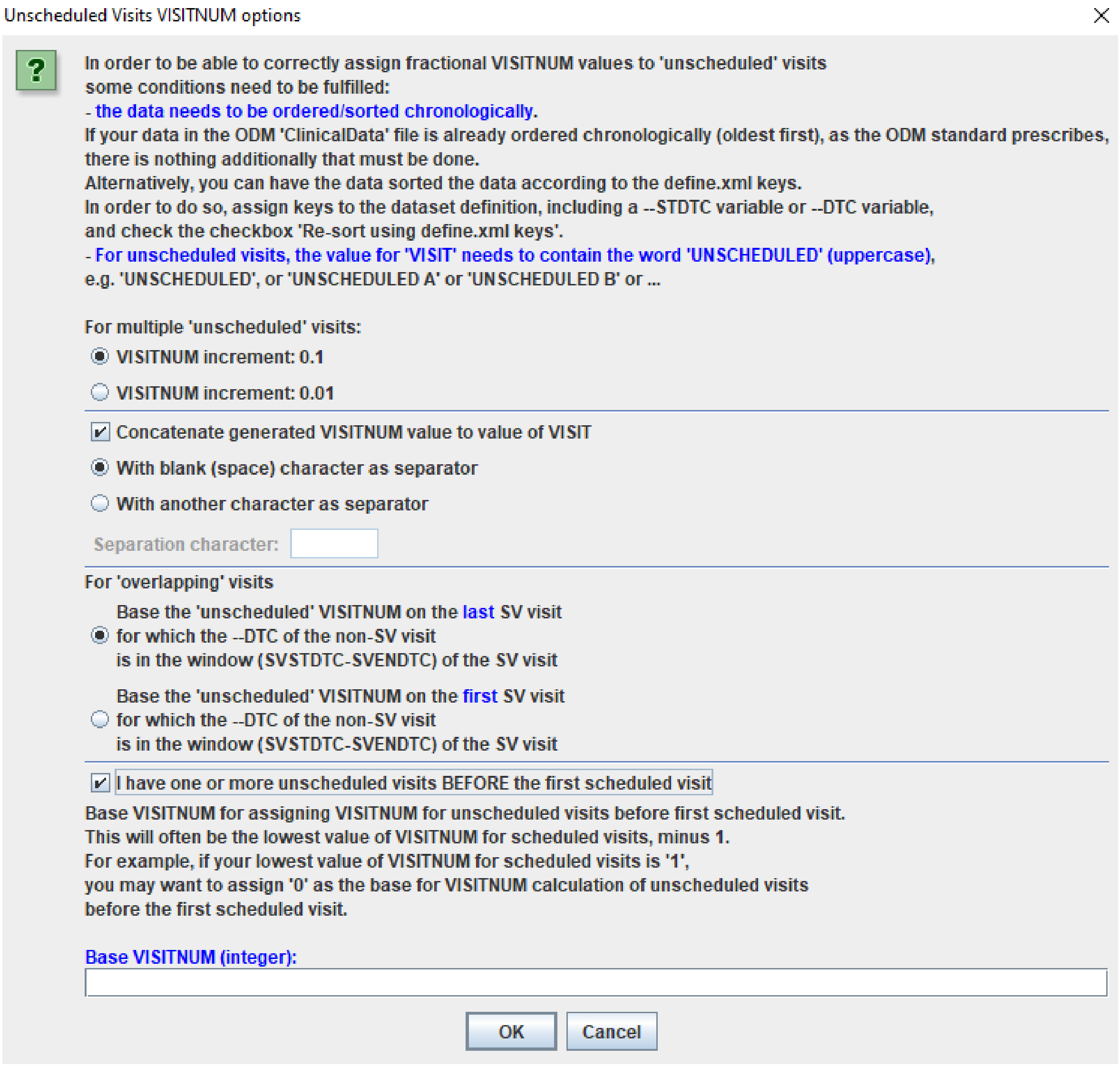
☑ Move non-standard SDTM Variables to SUPP--

□ Move Comment Variables to Comments (CO) Domain

□ Move Relrec Variables to Related Records (RELREC) domain

□ Try to generate 1:N RELREC Relationships

When one then checks the checkbox, the following (now extended) dialog is displayed:



Unscheduled Visits VISITNUM options

In order to be able to correctly assign fractional VISITNUM values to 'unscheduled' visits
some conditions need to be fulfilled:
- the data needs to be ordered/sorted chronologically.
If your data in the ODM 'ClinicalData' file is already ordered chronologically (oldest first), as the ODM standard prescribes,
there is nothing additionally that must be done.
Alternatively, you can have the data sorted the data according to the define.xml keys.
In order to do so, assign keys to the dataset definition, including a --STDTC variable or --DTC variable,
and check the checkbox 'Re-sort using define.xml keys'.
- For unscheduled visits, the value for 'VISIT' needs to contain the word 'UNSCHEDULED' (uppercase),
e.g. 'UNSCHEDULED', or 'UNSCHEDULED A' or 'UNSCHEDULED B' or ...

For multiple 'unscheduled' visits:

● VISITNUM increment: 0.1

○ VISITNUM increment: 0.01

☑ Concatenate generated VISITNUM value to value of VISIT

● With blank (space) character as separator

○ With another character as separator

Separation character:

For 'overlapping' visits

● Base the 'unscheduled' VISITNUM on the last SV visit
for which the --DTC of the non-SV visit
is in the window (SVSTDTC-SVENDTC) of the SV visit

○ Base the 'unscheduled' VISITNUM on the first SV visit
for which the --DTC of the non-SV visit
is in the window (SVSTDTC-SVENDTC) of the SV visit

☑ I have one or more unscheduled visits BEFORE the first scheduled visit

Base VISITNUM for assigning VISITNUM for unscheduled visits before first scheduled visit.
This will often be the lowest value of VISITNUM for scheduled visits, minus 1.
For example, if your lowest value of VISITNUM for scheduled visits is '1',
you may want to assign '0' as the base for VISITNUM calculation of unscheduled visits
before the first scheduled visit.

Base VISITNUM (integer):

New is the part for "overlapping" visits, i.e. where it cannot be exactly determined whether the unscheduled visit comes before or after scheduled visit (as in the SV dataset). This will often occur when no time part for either the scheduled or unscheduled visit has been collected.

Also new is the part for unscheduled visits that come before the first scheduled visit, and for which also a number for VISITNUM must be assigned. This situation should essentially never occur in well-designed and well-executed visits, but we have seen cases over and over again where this situation occurs.

CDISC CORE version 0.9.1

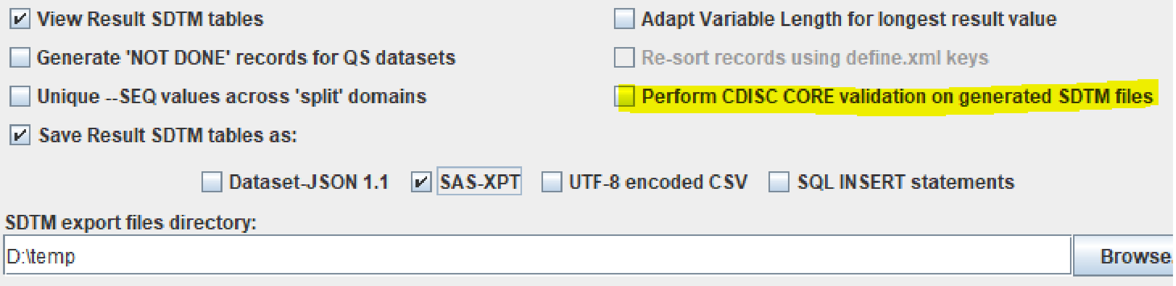
CDISC CORE (CDISC Open Rules Engine) is a very successful CDISC project, aiming to come to a really open and transparent set of validation rules for SDTM, SEND and ADaM submissions. Currently also the "FDA Business Rules" are being implemented, as well as rules for "TIG", the "Tobacco Implementation Guide" in strong cooperation with the FDA "Center for Tobacco Products".

Currently, the FDA is highly interested in CORE, so that we expect that the current validation software of the FDA will soon be replaced by CORE, or software products based on CORE.

Version 0.9.1 of CORE already contains a very good number of the "FDA Business Rules" besides hundreds of the classic CDISC validation rules for SDTM, SEND and ADaM, reason for which we decided to add this CORE version to the software.

Currently, the CORE implementation is limited to SAS Transport 5 as the dataset format, but work is currently being done to also support the new Dataset-JSON v.1.1 format, which will further contribute to the acceptance of Dataset-JSON as the submission format of the near future.

So, when selecting "SAS-XPT" as the output of choice, the checkbox "Perform CDISC CORE validation on generated SDTM files"[3] becomes enabled:



☑ View Result SDTM tables

□ Generate 'NOT DONE' records for QS datasets

□ Unique --SEQ values across 'split' domains

☑ Save Result SDTM tables as:

□ Dataset-JSON 1.1 ☑ SAS-XPT □ UTF-8 encoded CSV □ SQL INSERT statements

□ Adapt Variable Length for longest result value

□ Re-sort records using define.xml keys

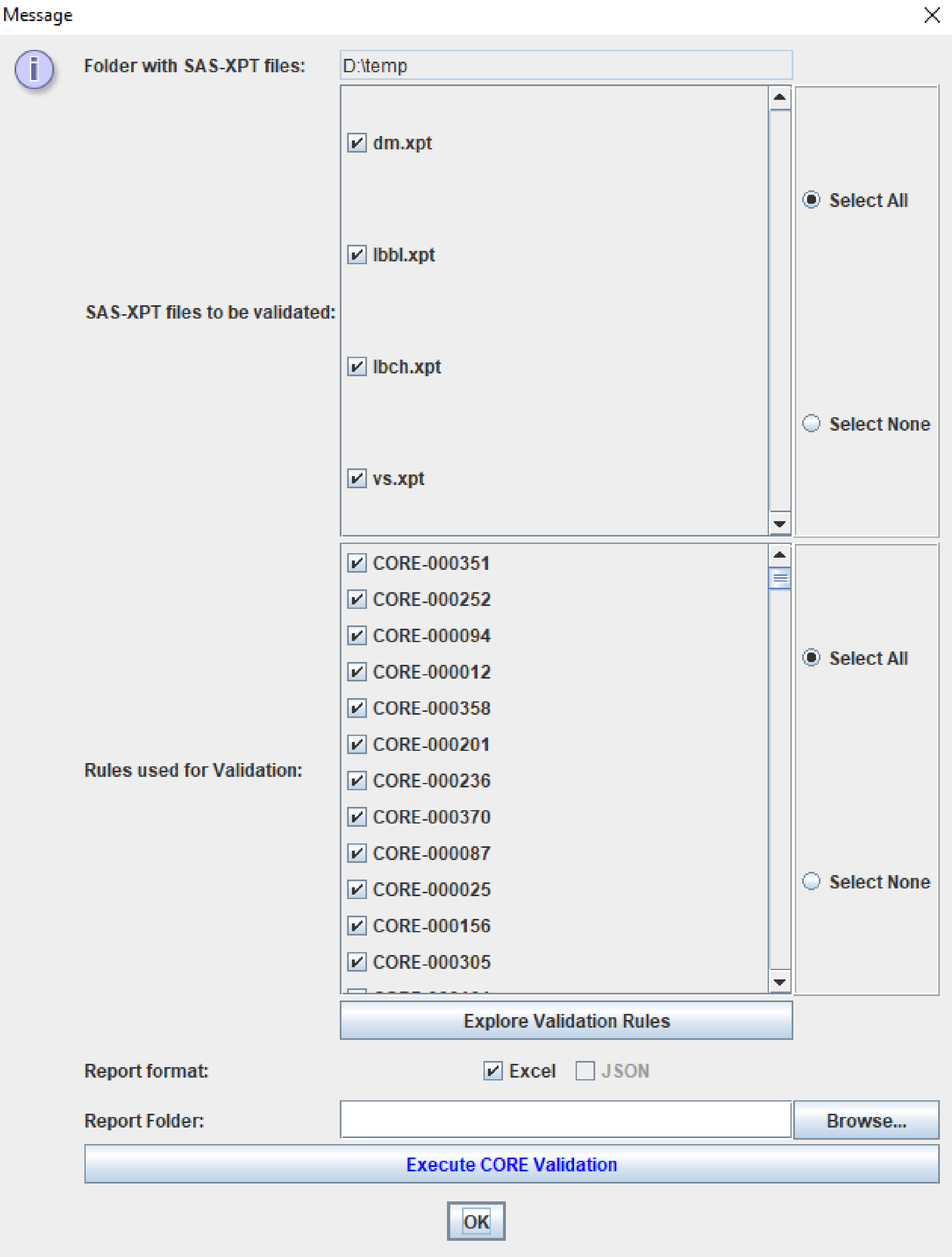
□ Perform CDISC CORE validation on generated SDTM files

SDTM export files directory:

D:\temp

When then checking that checkbox, at the end of the process, a new dialog is opened:

[3] In the case of SEND, the text will of course be "… on generated SEND files".



Message

Folder with SAS-XPT files: D:\temp

SAS-XPT files to be validated:

☑ dm.xpt

☑ lbbl.xpt

☑ lbch.xpt

☑ vs.xpt

● Select All ○ Select None

Rules used for Validation:

☑ CORE-000351

☑ CORE-000252

☑ CORE-000094

☑ CORE-000012

☑ CORE-000358

☑ CORE-000201

☑ CORE-000236

☑ CORE-000370

☑ CORE-000087

☑ CORE-000025

☑ CORE-000156

☑ CORE-000305

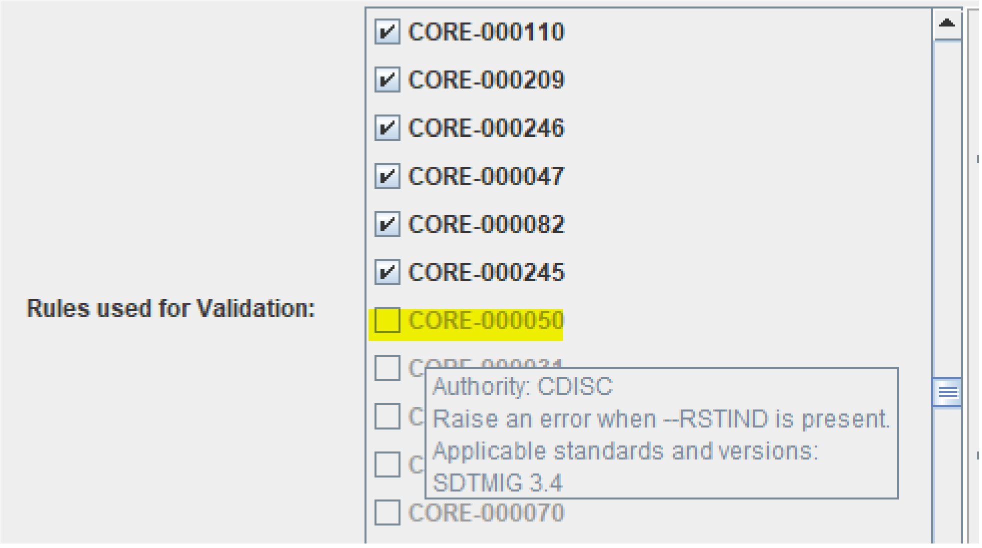
● Select All ○ Select None

Report format: ☑ Excel □ JSON

Report Folder:

allowing the user to select which generated datasets need to be validated, and to select (if wanted) which rules will be executed on the selected datasets.

Remark that checkboxes for the rules that do not apply to the selected submission standard and version are automatically disabled, e.g.:



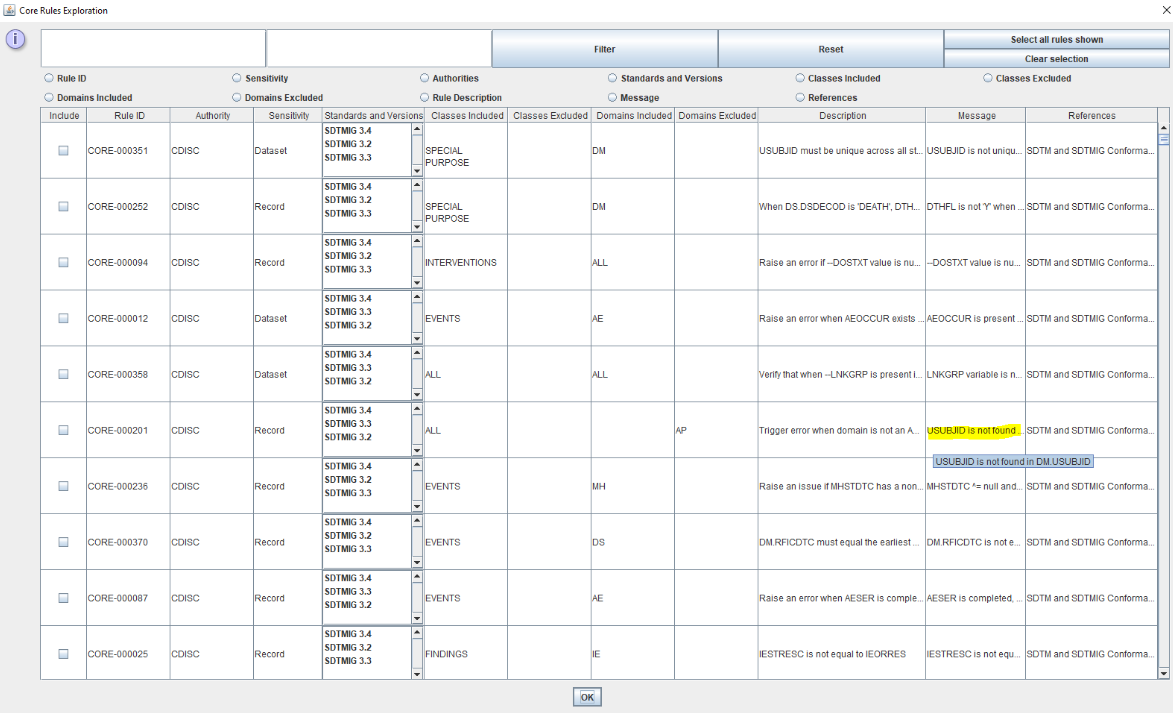
Rules used for Validation:

- ☑ CORE-000110
- ☑ CORE-000209
- ☑ CORE-000246
- ☑ CORE-000047
- ☑ CORE-000082
- ☑ CORE-000245
- □ CORE-000050
- □ CORE-000031
- □ C
- □ C
- □ CORE-000070

Authority: CDISC
Raise an error when --RSTIND is present.
Applicable standards and versions:
SDTMIG 3.4

Rule CORE-000050 is only applicable to SDTMIG version 3.4, but we are currently generating SDTMIG version 3.3 datasets, so that checkbox is disabled.

When clicking the "Explore Validation Rules", the following dialog is displayed:



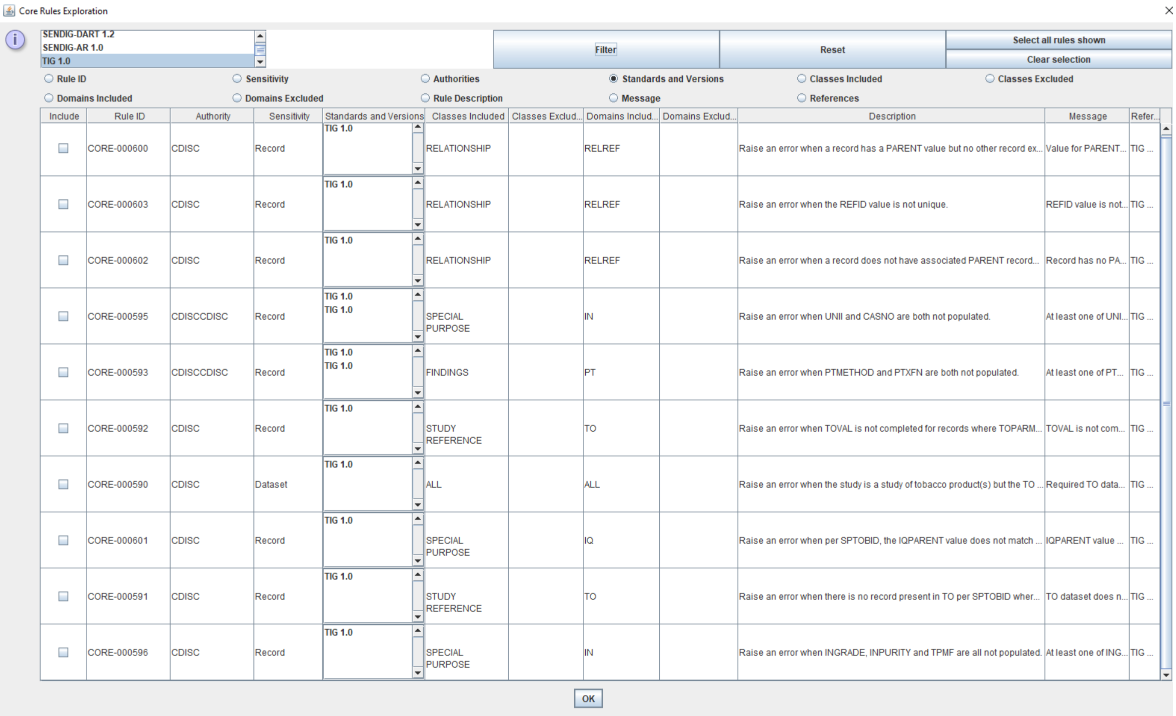
Core Rules Exploration

Include	Rule ID	Authority	Sensitivity	Standards and Versions	Classes Included	Classes Excluded	Domains Included	Domains Excluded	Description	Message	References
□	CORE-000351	CDISC	Dataset	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3	SPECIAL PURPOSE		DM		USUBJID must be unique across all st...	USUBJID is not uniqu...	SDTM and SDTMIG Conforma...
□	CORE-000252	CDISC	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3	SPECIAL PURPOSE		DM		When DS.DSDECOD is 'DEATH', DTH...	DTHFL is not 'Y' when ...	SDTM and SDTMIG Conforma...
□	CORE-000094	CDISC	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3	INTERVENTIONS		ALL		Raise an error if --DOSTXT value is nu...	--DOSTXT value is nu...	SDTM and SDTMIG Conforma...
□	CORE-000012	CDISC	Dataset	SDTMIG 3.4 SDTMIG 3.3 SDTMIG 3.2	EVENTS		AE		Raise an error when AEOCCUR exists ...	AEOCCUR is present ...	SDTM and SDTMIG Conforma...
□	CORE-000358	CDISC	Dataset	SDTMIG 3.4 SDTMIG 3.3 SDTMIG 3.2	ALL		ALL		Verify that when --LNKGRP is present i...	LNKGRP variable is n...	SDTM and SDTMIG Conforma...
□	CORE-000201	CDISC	Record	SDTMIG 3.4 SDTMIG 3.3 SDTMIG 3.2	ALL			AP	Trigger error when domain is not an A...	USUBJID is not found ...	SDTM and SDTMIG Conforma...
□	CORE-000236	CDISC	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3	EVENTS		MH		Raise an issue if MHSTDTC has a non...	MHSTDTC ^= null and...	SDTM and SDTMIG Conforma...
□	CORE-000370	CDISC	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3	EVENTS		DS		DM.RFICDTC must equal the earliest ...	DM.RFICDTC is not e...	SDTM and SDTMIG Conforma...
□	CORE-000087	CDISC	Record	SDTMIG 3.4 SDTMIG 3.3 SDTMIG 3.2	EVENTS		AE		Raise an error when AESER is comple...	AESER is completed, ...	SDTM and SDTMIG Conforma...
□	CORE-000025	CDISC	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3	FINDINGS		IE		IESTRESC is not equal to IEORRES	IESTRESC is not equ...	SDTM and SDTMIG Conforma...

USUBJID is not found in DM.USUBJID

allowing the user to select or filter validation rules to be included based on SDTMIG version, domain, SDTM/SEND class, authority (CDISC, FDA, …) and many other things.

For example, when one filters on "Standards and Versions" and selects "TIG", one sees:



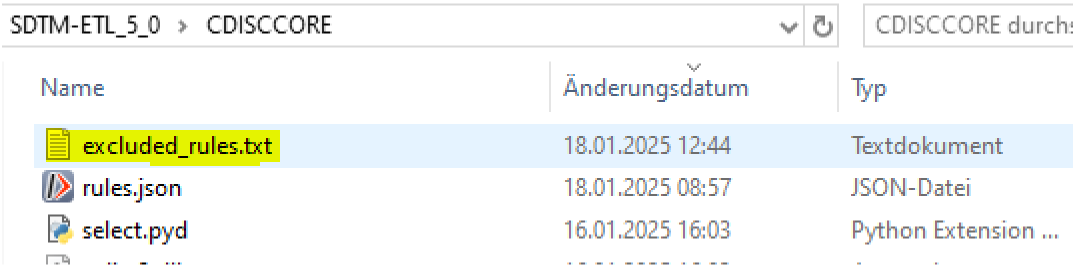
○ Rule ID ○ Sensitivity ○ Authorities ● Standards and Versions ○ Classes Included ○ Classes Excluded
○ Domains Included ○ Domains Excluded ○ Rule Description ○ Message ○ References

Include	Rule ID	Authority	Sensitivity	Standards and Versions	Classes Included	Classes Exclud...	Domains Includ...	Domains Exclud...	Description	Message	Refer...
□	CORE-000600	CDISC	Record	TIG 1.0	RELATIONSHIP		RELREF		Raise an error when a record has a PARENT value but no other record ex...	Value for PARENT...	TIG ...
□	CORE-000603	CDISC	Record	TIG 1.0	RELATIONSHIP		RELREF		Raise an error when the REFID value is not unique.	REFID value is not...	TIG ...
□	CORE-000602	CDISC	Record	TIG 1.0	RELATIONSHIP		RELREF		Raise an error when a record does not have associated PARENT record...	Record has no PA...	TIG ...
□	CORE-000595	CDISCCDISC	Record	TIG 1.0 TIG 1.0	SPECIAL PURPOSE		IN		Raise an error when UNII and CASNO are both not populated.	At least one of UNI...	TIG ...
□	CORE-000593	CDISCCDISC	Record	TIG 1.0 TIG 1.0	FINDINGS		PT		Raise an error when PTMETHOD and PTXFN are both not populated.	At least one of PT...	TIG ...
□	CORE-000592	CDISC	Record	TIG 1.0	STUDY REFERENCE		TO		Raise an error when TOVAL is not completed for records where TOPARM...	TOVAL is not com...	TIG ...
□	CORE-000590	CDISC	Dataset	TIG 1.0	ALL		ALL		Raise an error when the study is a study of tobacco product(s) but the TO ...	Required TO data...	TIG ...
□	CORE-000601	CDISC	Record	TIG 1.0	SPECIAL PURPOSE		IQ		Raise an error when per SPTOBID, the IQPARENT value does not match ...	IQPARENT value ...	TIG ...
□	CORE-000591	CDISC	Record	TIG 1.0	STUDY REFERENCE		TO		Raise an error when there is no record present in TO per SPTOBID wher...	TO dataset does n...	TIG ...
□	CORE-000596	CDISC	Record	TIG 1.0	SPECIAL PURPOSE		IN		Raise an error when INGRADE, INPURITY and TPMF are all not populated.	At least one of ING...	TIG ...

showing only the "TIG" (Tobacco Implementation Guide") rules, and can select them by clicking check boxes, or use the button "Select all rules shown".

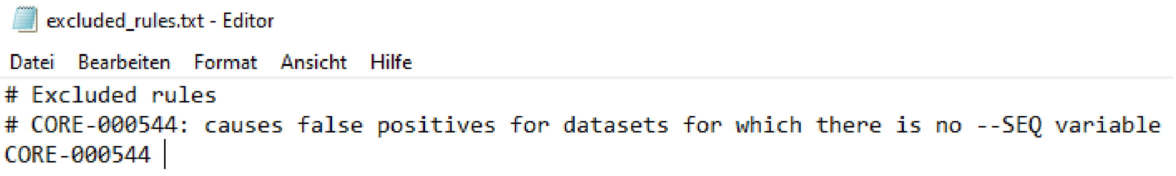
Selecting a set of all rules, instead of all the rules, allows to avoid a mass of violations reported when still in the development of the datasets, which is currently not possible with the validation software of a company that is also used by FDA. We however expect FDA to soon stop using that software, and move to CDISC CORE.

Another new feature is that the user can keep a list of rules always to be excluded in the file "excluded_rules.txt" that is found in the folder "CDISCCORE":



Name	Änderungsdatum	Typ
excluded_rules.txt	18.01.2025 12:44	Textdokument
rules.json	18.01.2025 08:57	JSON-Datei
select.pyd	16.01.2025 16:03	Python Extension ...

e.g. with the content:

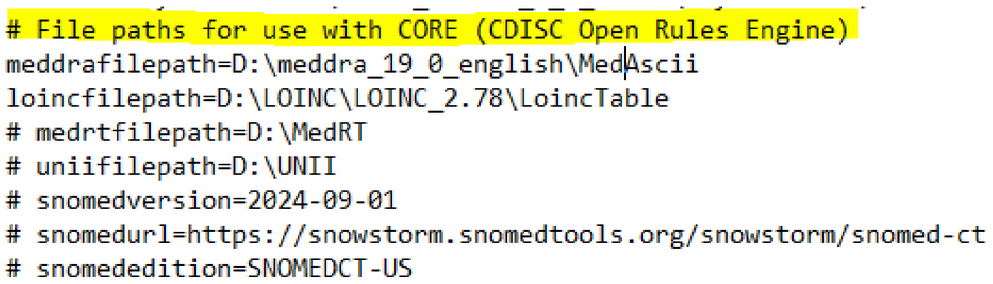


```
# Excluded rules
# CORE-000544: causes false positives for datasets for which there is no --SEQ variable
CORE-000544
```

where lines started by a "#" are comment lines.
This mechanism allows to disable rules that are known to cause problems, such as rule CORE-000544 in version 0.9.1 that causes false positives for datasets for which there is no -

-SEQ variable, such as SV (Subject Visits).
When then a new, updated set of the rules is released by CDISC, or a new version of the CORE engine, the contents of the file can be adapted.

As CORE is now also using external dictionaries (such as MedDRA, WHO-drug, …) for validation, the paths to the files for these should be added in order to being able to use such rules. These paths must be added to the "properties.dat" file that is located in the folder where the SDTM-ETL software is installed, and e.g. contains the following content:



```
# File paths for use with CORE (CDISC Open Rules Engine)
meddrafilepath=D:\meddra_19_0_english\MedAscii
loincfilepath=D:\LOINC\LOINC_2.78\LoincTable
# medrtfilepath=D:\MedRT
# uniifilepath=D:\UNII
# snomedversion=2024-09-01
# snomedurl=https://snowstorm.snomedtools.org/snowstorm/snomed-ct
# snomededition=SNOMEDCT-US
```

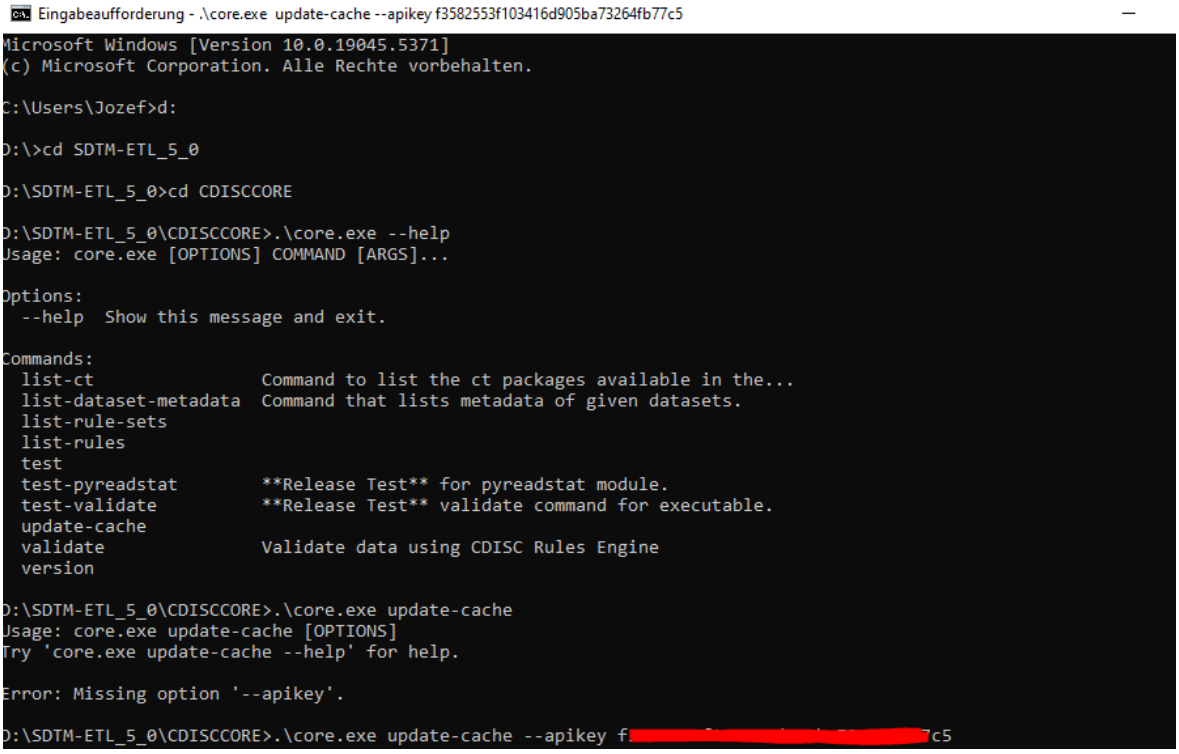
Updating the CORE engine and rules set

Essentially the implementation of the CORE engine that comes with SDTM-ETL is independent of the software itself.
This means that the CORE engine can be replaced by a newer one without needing a new version of SDTM-ETL itself: it suffices to download the Windows version of the core engine from the CORE download site, click on "Assets", and then look for and download "core-windows.zip". Then unzip the file and copy the contents including all subfolder to the "CDISCCORE" folder. The update should then work "out of the box".
If you have installed SDTM-ETL on Linux or Mac, please contact us for further details.

The rules themselves are stored in the files "rules.json". If you want to update them, you can run the command:

.\core.exe update-cache --apikey xxxxx

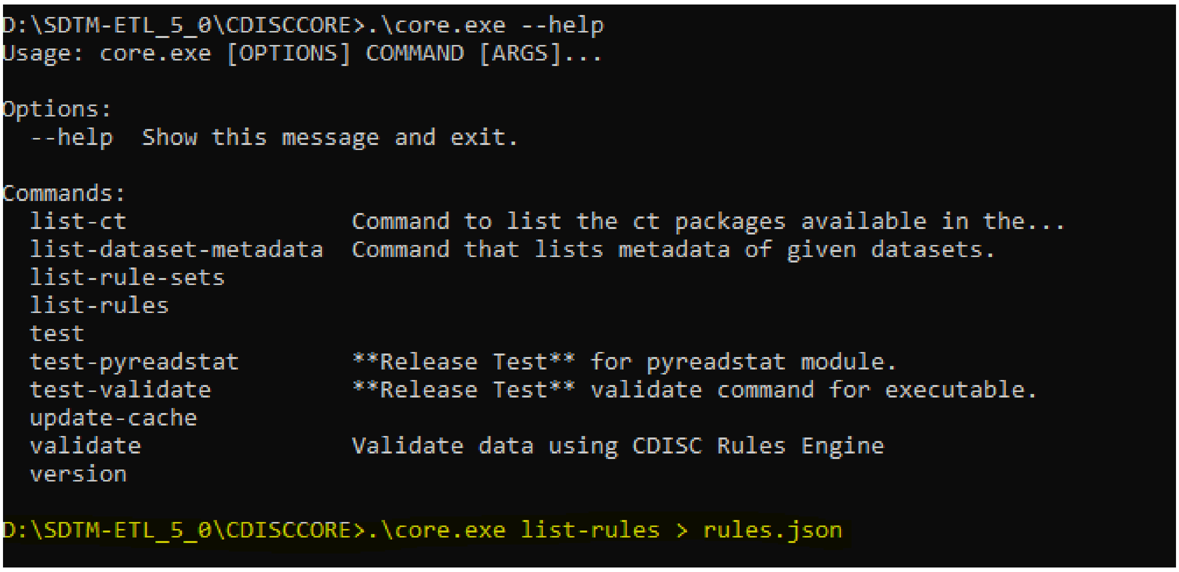
where "xxxxx" is your CDISC-Library API key. For example:


Eingabeaufforderung - .\core.exe update-cache --apikey f3582553f103416d905ba73264fb77c5

```
Microsoft Windows [Version 10.0.19045.5371]
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Jozef>d:

D:\>cd SDTM-ETL_5_0

D:\SDTM-ETL_5_0>cd CDISCCORE

D:\SDTM-ETL_5_0\CDISCCORE>.\core.exe --help
Usage: core.exe [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  list-ct                Command to list the ct packages available in the...
  list-dataset-metadata  Command that lists metadata of given datasets.
  list-rule-sets
  list-rules
  test
  test-pyreadstat        **Release Test** for pyreadstat module.
  test-validate          **Release Test** validate command for executable.
  update-cache
  validate               Validate data using CDISC Rules Engine
  version

D:\SDTM-ETL_5_0\CDISCCORE>.\core.exe update-cache
Usage: core.exe update-cache [OPTIONS]
Try 'core.exe update-cache --help' for help.

Error: Missing option '--apikey'.

D:\SDTM-ETL_5_0\CDISCCORE>.\core.exe update-cache --apikey f[redacted]c5
```

The "cache" with the rules will then be updated (a lot of information messages will scroll by).

When finished, then run the command ".\core.exe --list-rules > rules.json"



```
D:\SDTM-ETL_5_0\CDISCCORE>.\core.exe --help
Usage: core.exe [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  list-ct                Command to list the ct packages available in the...
  list-dataset-metadata  Command that lists metadata of given datasets.
  list-rule-sets
  list-rules
  test
  test-pyreadstat        **Release Test** for pyreadstat module.
  test-validate          **Release Test** validate command for executable.
  update-cache
  validate               Validate data using CDISC Rules Engine
  version

D:\SDTM-ETL_5_0\CDISCCORE>.\core.exe list-rules > rules.json
```

to update the file "rules.json"

In the near future, we will add a button in the SDTM-ETL software itself to update the set of rules and the "rules.json" file.

Remark that in order to update the set of rules, you will need a "CDISC Library API key", which you can obtain at no cost from the "CDISC Library API" site:

https://api.developer.library.cdisc.org/

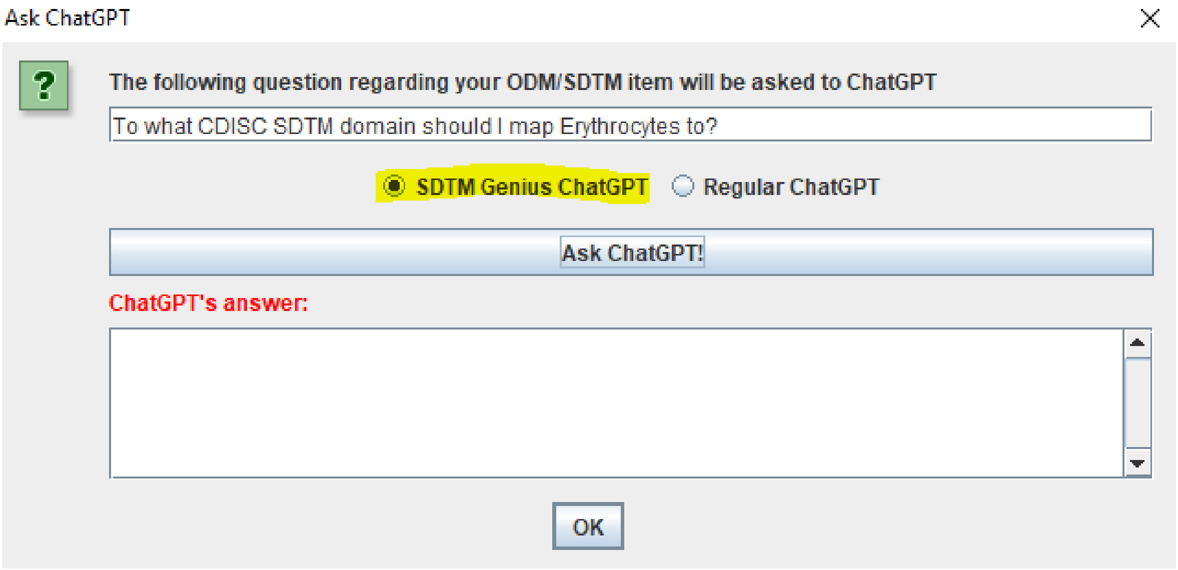
For running the CORE validation engine, no API key is needed at all.

Limitations

Currently, generating the CORE validation report in JSON format (for use in other applications) is being disable, due to a bug in CORE-0.9.1.
We hope to be able to add it again very soon.

Choice between "SDTM Genius" and regular ChatGPT for mapping advice

An interesting evolution is that people can now develop "specialized additions" to ChatGPT. For CDISC-SDTM, such an extension has been made available by Sai Krishna Reddy of BMS, named "Genius SDTM", which usually behaves much better than "regular" ChatGPT. So, when using the menu "Explore - Ask ChatGPT for mapping suggestions", the user can now choose between "SDTM Genius ChatGPT" and "regular" ChatGPT:



Ask ChatGPT

The following question regarding your ODM/SDTM item will be asked to ChatGPT

To what CDISC SDTM domain should I map Erythrocytes to?

● SDTM Genius ChatGPT ○ Regular ChatGPT

ChatGPT's answer:

Use of Oracle-SQL for generating and populating relational databases (as of v.4.6)

SDTM-ETL comes with features for generating (SQL-based) databases with sets of SDTM/SEND tables. Unfortunately, there are still many "dialects" and "specializations" of SQL. The one that is implemented by default will work for most users, and is e.g. compatible with MySQL.
We do now also support Oracle-SQL. If you want to use Oracle-SQL, just add a line:

```
useoraclesql=true
```

to the properties.dat file, and the software will generate Oracle-compatible SQL commands when using one of the SQL features of the software.

Stopping support for Define-XML v.1.0

As most regulatory authorities do not accept Define-XML 1.0 anymore, we decided to stop supporting Define-XML 1.0 anymore, and thus removed the choice in the software and the folder "define_1_0" containing the template files for this version.
Instead we encourage users to start using Define-XML v.2.1, which is the newest version and accepted by almost all regulatory authorities.

Updated template files, generated from the CDISC-Library

Many of our Define-XML templates were still generated by carefully inspecting the SDTM- and SENDIGs and using copy-paste. Now that the CDISC Library with its API allows to automate the generation of these template files (we developed software for that), we replaced all template define.xml files for both Define-XML 2.0 as 2.1 by template files generated from the CDISC Library using the API.
This leads to a higher quality (although it was already very high), ensuring e.g. that all dataset and variable labels and data types are correct.

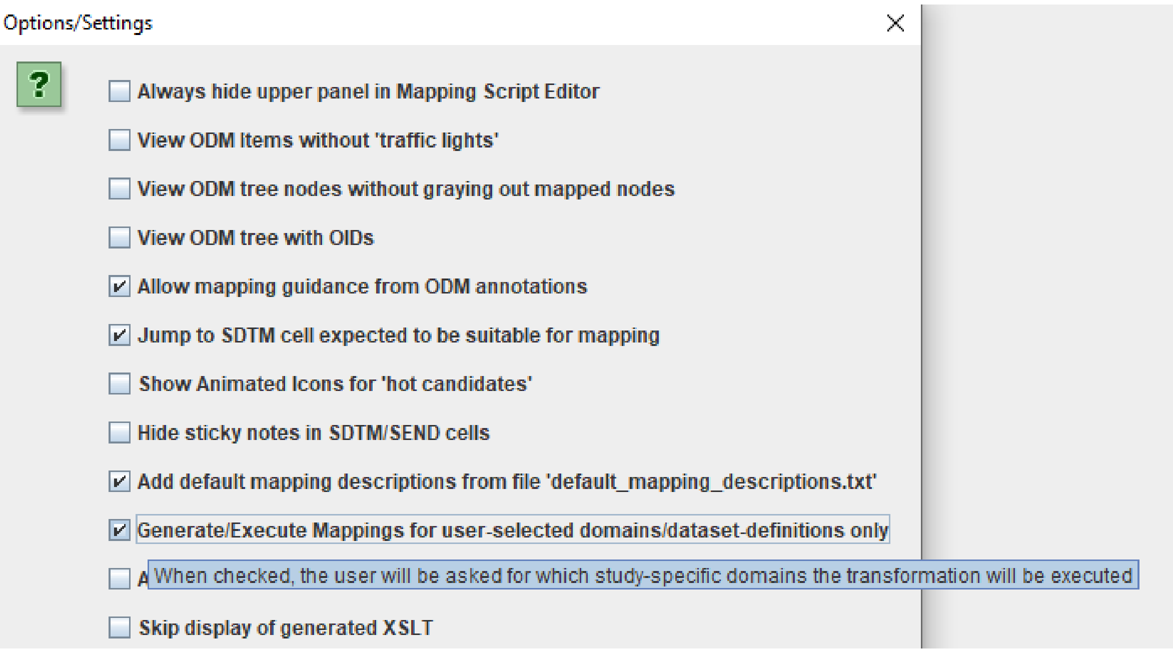
New templates for SENDIG-DART, SENDIG-AR and SENDIG-GENETOX

This also allowed us to generate templates for SEND DART-1.1, SENDIG-AR-1.0 and SENDIG-GENETOX-1.0. The latter is a combined template for SENDIG-3.1.1 together with the specific GENETOX domains such as GV (Genetic Toxicology - In Vivo) and updates to TS (Trial Summary).

Generate/Execute Mappings for user-selected domains/dataset-definitions only

In some cases one may have that one has set up or loaded mappings for several domain/dataset-definitions, but may want to test these for only one or two of these, e.g. to save time in the case of large datasets. Until now, the usual way was then to delete the rows in the mapping table for those domains/datasets for which one did not want to have the mappings executed, and later, when all updates made to the domain/dataset definitions that one is testing, remerge with the original dataset definitions.
This is of course time-intensive in the case of a large number of domain/dataset-definitions, and can also lead to errors when remerging.

In SDTM-ETL 5.0 it is now possible to only execute the mappings for only a selection of the loaded domain/dataset-definitions. In order to enable this feature, use the menu "Options - Settings" and check the checkbox "Generate/Execute Mappings for user-selected domains/dataset-definitions only":



Options/Settings

□ Always hide upper panel in Mapping Script Editor

□ View ODM Items without 'traffic lights'

□ View ODM tree nodes without graying out mapped nodes

□ View ODM tree with OIDs

☑ Allow mapping guidance from ODM annotations

☑ Jump to SDTM cell expected to be suitable for mapping

□ Show Animated Icons for 'hot candidates'

□ Hide sticky notes in SDTM/SEND cells

☑ Add default mapping descriptions from file 'default_mapping_descriptions.txt'

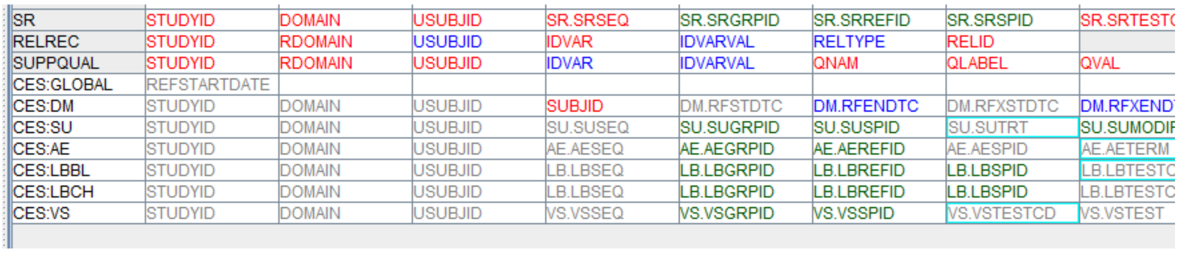
☑ Generate/Execute Mappings for user-selected domains/dataset-definitions only

□ A

When checked, the user will be asked for which study-specific domains the transformation will be executed

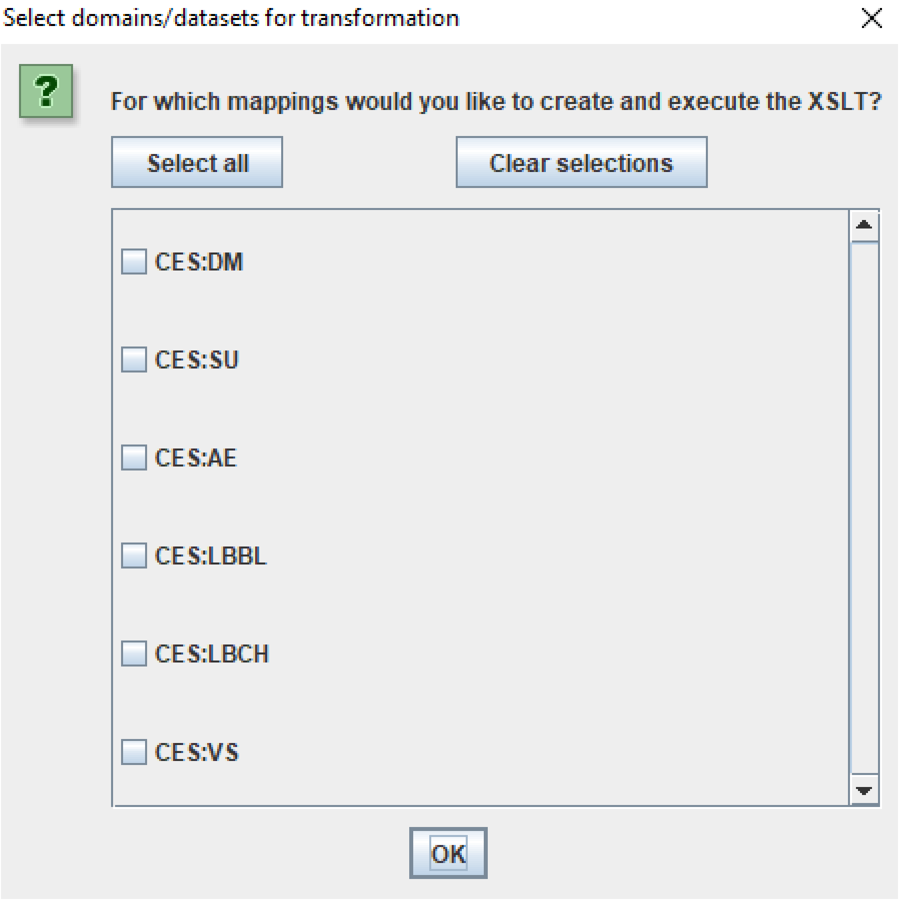
□ Skip display of generated XSLT

In case we have e.g. loaded:



SR	STUDYID	DOMAIN	USUBJID	SR.SRSEQ	SR.SRGRPID	SR.SRREFID	SR.SRSPID	SR.SRTESTC
RELREC	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RELTYPE	RELID	
SUPPQUAL	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	QNAM	QLABEL	QVAL
CES:GLOBAL	REFSTARTDATE							
CES:DM	STUDYID	DOMAIN	USUBJID	SUBJID	DM.RFSTDTC	DM.RFENDTC	DM.RFXSTDTC	DM.RFXEND
CES:SU	STUDYID	DOMAIN	USUBJID	SU.SUSEQ	SU.SUGRPID	SU.SUSPID	SU.SUTRT	SU.SUMODIF
CES:AE	STUDYID	DOMAIN	USUBJID	AE.AESEQ	AE.AEGRPID	AE.AEREFID	AE.AESPID	AE.AETERM
CES:LBBL	STUDYID	DOMAIN	USUBJID	LB.LBSEQ	LB.LBGRPID	LB.LBREFID	LB.LBSPID	LB.LBTESTC
CES:LBCH	STUDYID	DOMAIN	USUBJID	LB.LBSEQ	LB.LBGRPID	LB.LBREFID	LB.LBSPID	LB.LBTESTC
CES:VS	STUDYID	DOMAIN	USUBJID	VS.VSSEQ	VS.VSGRPID	VS.VSSPID	VS.VSTESTCD	VS.VSTEST

and when then executing the mappings using the "Transform" menu, the following dialog will appear:



Select domains/datasets for transformation

For which mappings would you like to create and execute the XSLT?

□ CES:DM

□ CES:SU

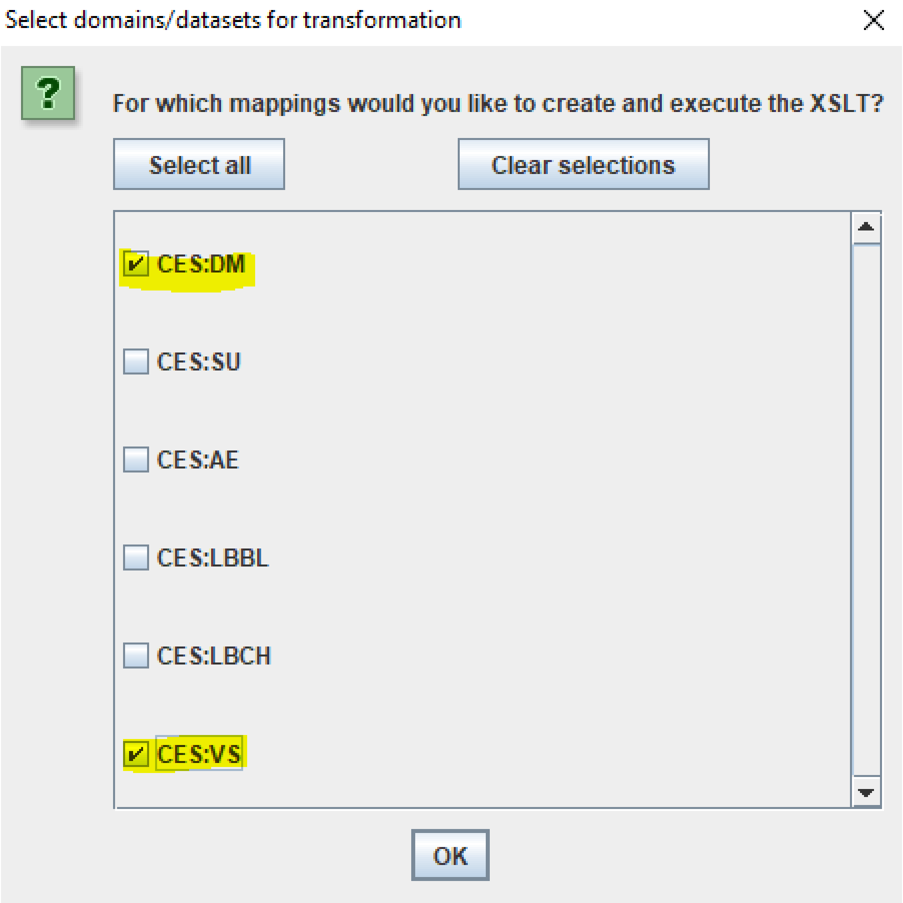
□ CES:AE

□ CES:LBBL

□ CES:LBCH

□ CES:VS

If we only want to execute the loaded mappings for e.g. "DM" and "VS", we can select these by the checkboxes:



For which mappings would you like to create and execute the XSLT?

☑ CES:DM

□ CES:SU

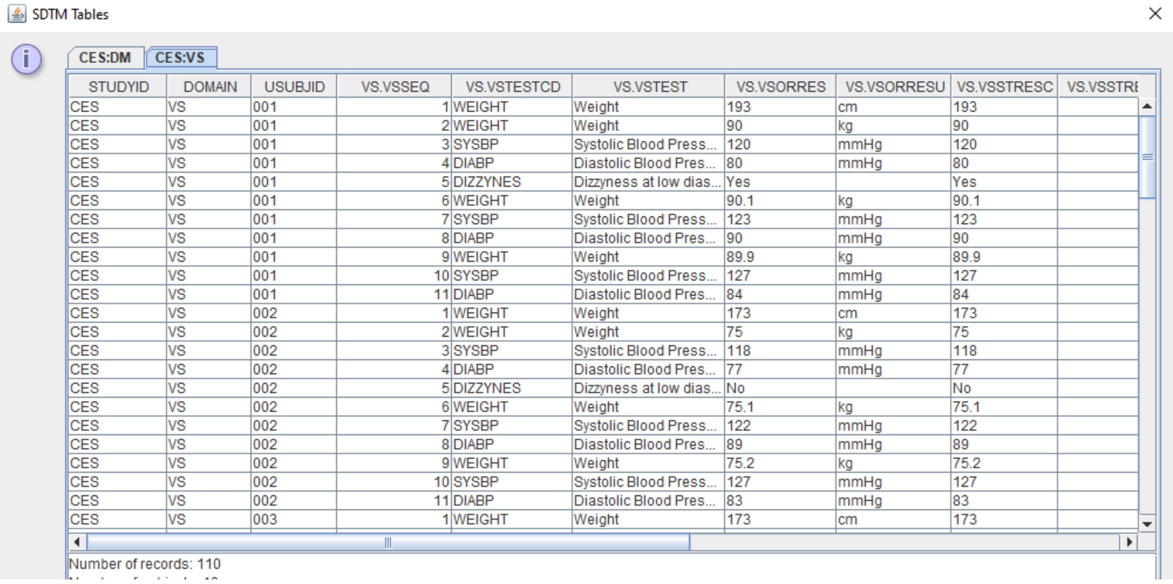
□ CES:AE

□ CES:LBBL

□ CES:LBCH

☑ CES:VS

finally leading to the result:

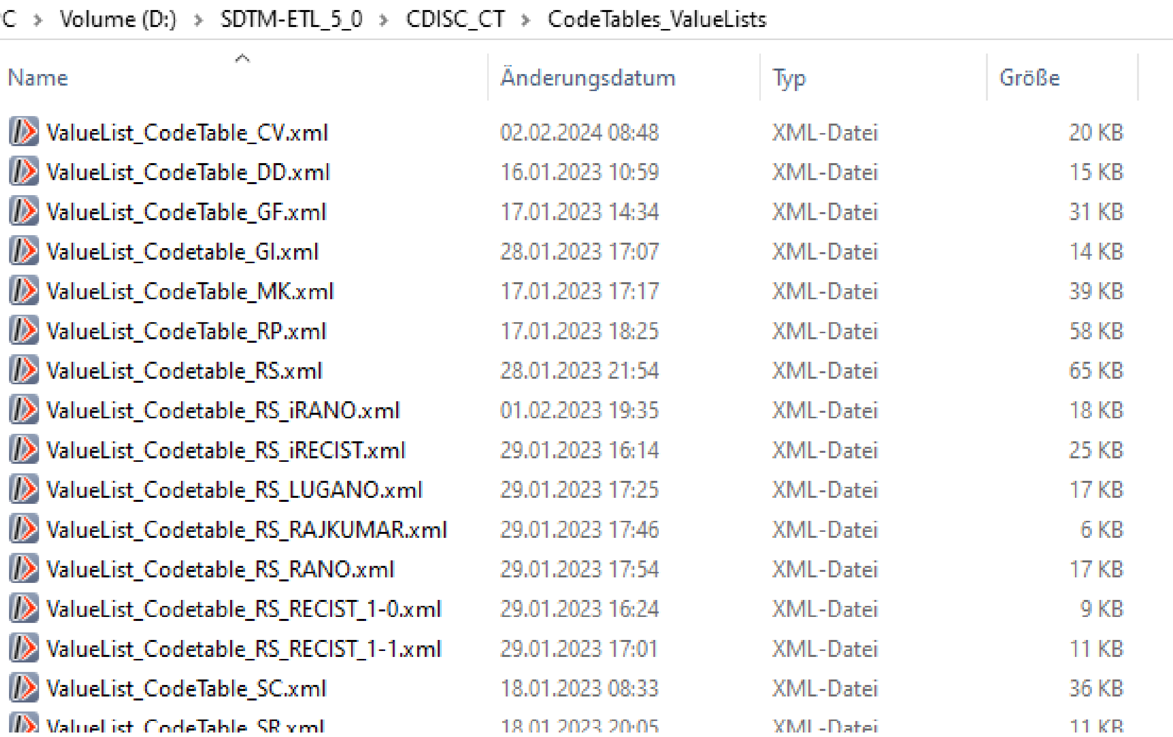


STUDYID	DOMAIN	USUBJID	VS.VSSEQ	VS.VSTESTCD	VS.VSTEST	VS.VSORRES	VS.VSORRESU	VS.VSSTRESC	VS.VSSTRE
CES	VS	001	1	WEIGHT	Weight	193	cm	193	
CES	VS	001	2	WEIGHT	Weight	90	kg	90	
CES	VS	001	3	SYSBP	Systolic Blood Press...	120	mmHg	120	
CES	VS	001	4	DIABP	Diastolic Blood Pres...	80	mmHg	80	
CES	VS	001	5	DIZZYNES	Dizzyness at low dias...	Yes		Yes	
CES	VS	001	6	WEIGHT	Weight	90.1	kg	90.1	
CES	VS	001	7	SYSBP	Systolic Blood Press...	123	mmHg	123	
CES	VS	001	8	DIABP	Diastolic Blood Pres...	90	mmHg	90	
CES	VS	001	9	WEIGHT	Weight	89.9	kg	89.9	
CES	VS	001	10	SYSBP	Systolic Blood Press...	127	mmHg	127	
CES	VS	001	11	DIABP	Diastolic Blood Pres...	84	mmHg	84	
CES	VS	002	1	WEIGHT	Weight	173	cm	173	
CES	VS	002	2	WEIGHT	Weight	75	kg	75	
CES	VS	002	3	SYSBP	Systolic Blood Press...	118	mmHg	118	
CES	VS	002	4	DIABP	Diastolic Blood Pres...	77	mmHg	77	
CES	VS	002	5	DIZZYNES	Dizzyness at low dias...	No		No	
CES	VS	002	6	WEIGHT	Weight	75.1	kg	75.1	
CES	VS	002	7	SYSBP	Systolic Blood Press...	122	mmHg	122	
CES	VS	002	8	DIABP	Diastolic Blood Pres...	89	mmHg	89	
CES	VS	002	9	WEIGHT	Weight	75.2	kg	75.2	
CES	VS	002	10	SYSBP	Systolic Blood Press...	127	mmHg	127	
CES	VS	002	11	DIABP	Diastolic Blood Pres...	83	mmHg	83	
CES	VS	003	1	WEIGHT	Weight	173	cm	173	

Number of records: 110

Using pre-defined ValueLists from CDISC "Code Tables"

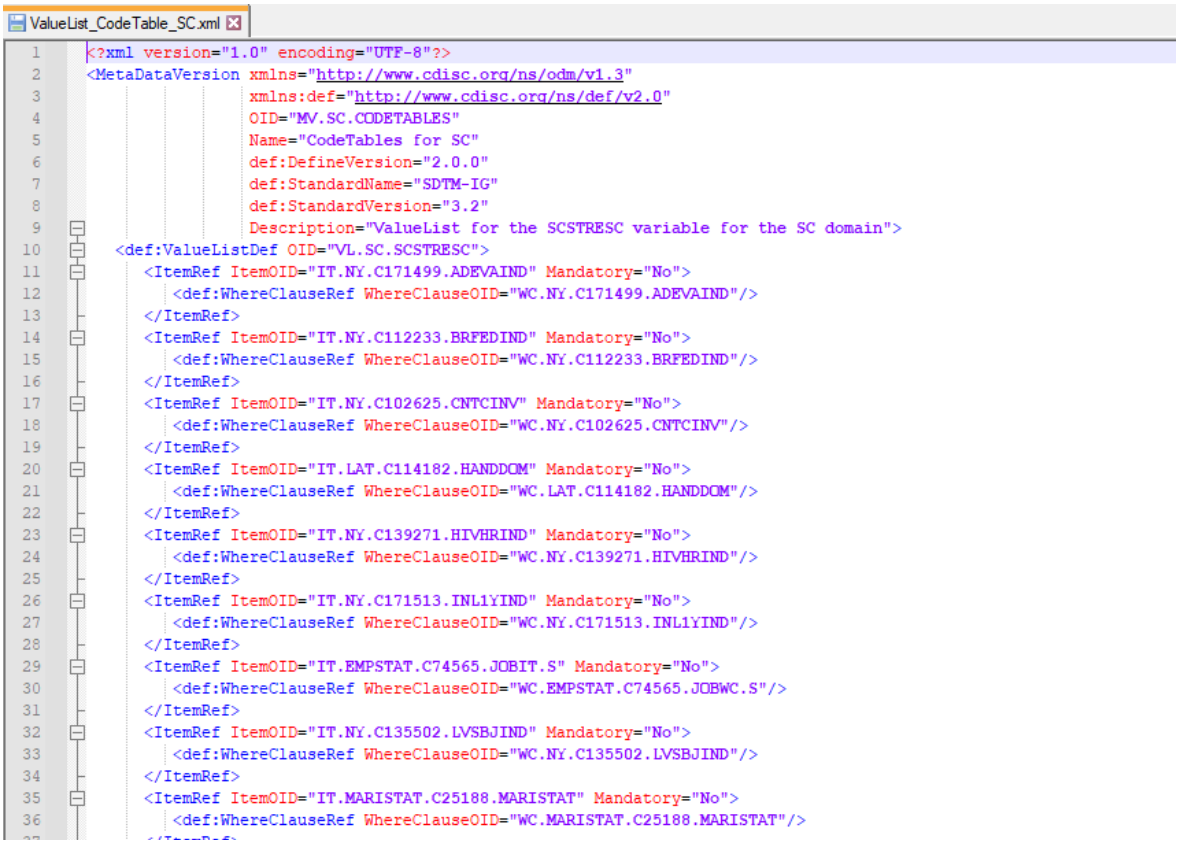
The software now comes with a set of pre-defined "ValueList" files that have been generated from the by CDISC published "Code Tables". These files can be found in the folder "CDISC_CT/CodeTables_ValueLists":



C › Volume (D:) › SDTM-ETL_5_0 › CDISC_CT › CodeTables_ValueLists

Name	Änderungsdatum	Typ	Größe
ValueList_CodeTable_CV.xml	02.02.2024 08:48	XML-Datei	20 KB
ValueList_CodeTable_DD.xml	16.01.2023 10:59	XML-Datei	15 KB
ValueList_CodeTable_GF.xml	17.01.2023 14:34	XML-Datei	31 KB
ValueList_Codetable_GI.xml	28.01.2023 17:07	XML-Datei	14 KB
ValueList_CodeTable_MK.xml	17.01.2023 17:17	XML-Datei	39 KB
ValueList_CodeTable_RP.xml	17.01.2023 18:25	XML-Datei	58 KB
ValueList_Codetable_RS.xml	28.01.2023 21:54	XML-Datei	65 KB
ValueList_Codetable_RS_iRANO.xml	01.02.2023 19:35	XML-Datei	18 KB
ValueList_Codetable_RS_iRECIST.xml	29.01.2023 16:14	XML-Datei	25 KB
ValueList_Codetable_RS_LUGANO.xml	29.01.2023 17:25	XML-Datei	17 KB
ValueList_Codetable_RS_RAJKUMAR.xml	29.01.2023 17:46	XML-Datei	6 KB
ValueList_Codetable_RS_RANO.xml	29.01.2023 17:54	XML-Datei	17 KB
ValueList_Codetable_RS_RECIST_1-0.xml	29.01.2023 16:24	XML-Datei	9 KB
ValueList_Codetable_RS_RECIST_1-1.xml	29.01.2023 17:01	XML-Datei	11 KB
ValueList_CodeTable_SC.xml	18.01.2023 08:33	XML-Datei	36 KB
ValueList_CodeTable_SR.xml	18.01.2023 20:05	XML-Datei	11 KB

When one e.g. opens the file "ValueList_CodeTable_SC" (for the "Subject Characteristics" domain) in NotePad++, one finds:

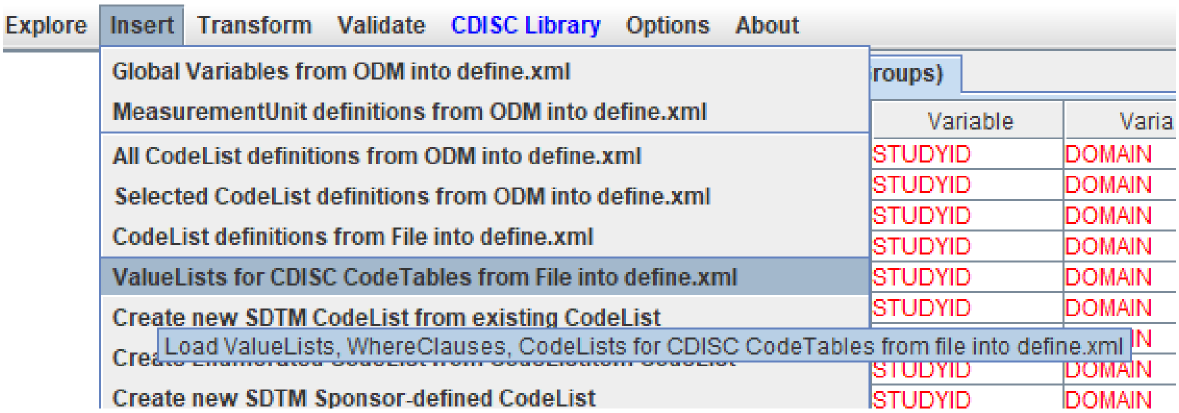


ValueList_CodeTable_SC.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<MetaDataVersion xmlns="http://www.cdisc.org/ns/odm/v1.3"
                 xmlns:def="http://www.cdisc.org/ns/def/v2.0"
                 OID="MV.SC.CODETABLES"
                 Name="CodeTables for SC"
                 def:DefineVersion="2.0.0"
                 def:StandardName="SDTM-IG"
                 def:StandardVersion="3.2"
                 Description="ValueList for the SCSTRESC variable for the SC domain">
   <def:ValueListDef OID="VL.SC.SCSTRESC">
      <ItemRef ItemOID="IT.NY.C171499.ADEVAIND" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.NY.C171499.ADEVAIND"/>
      </ItemRef>
      <ItemRef ItemOID="IT.NY.C112233.BRFEDIND" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.NY.C112233.BRFEDIND"/>
      </ItemRef>
      <ItemRef ItemOID="IT.NY.C102625.CNTCINV" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.NY.C102625.CNTCINV"/>
      </ItemRef>
      <ItemRef ItemOID="IT.LAT.C114182.HANDDOM" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.LAT.C114182.HANDDOM"/>
      </ItemRef>
      <ItemRef ItemOID="IT.NY.C139271.HIVHRIND" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.NY.C139271.HIVHRIND"/>
      </ItemRef>
      <ItemRef ItemOID="IT.NY.C171513.INL1YIND" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.NY.C171513.INL1YIND"/>
      </ItemRef>
      <ItemRef ItemOID="IT.EMPSTAT.C74565.JOBIT.S" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.EMPSTAT.C74565.JOBWC.S"/>
      </ItemRef>
      <ItemRef ItemOID="IT.NY.C135502.LVSBJIND" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.NY.C135502.LVSBJIND"/>
      </ItemRef>
      <ItemRef ItemOID="IT.MARISTAT.C25188.MARISTAT" Mandatory="No">
         <def:WhereClauseRef WhereClauseOID="WC.MARISTAT.C25188.MARISTAT"/>
```

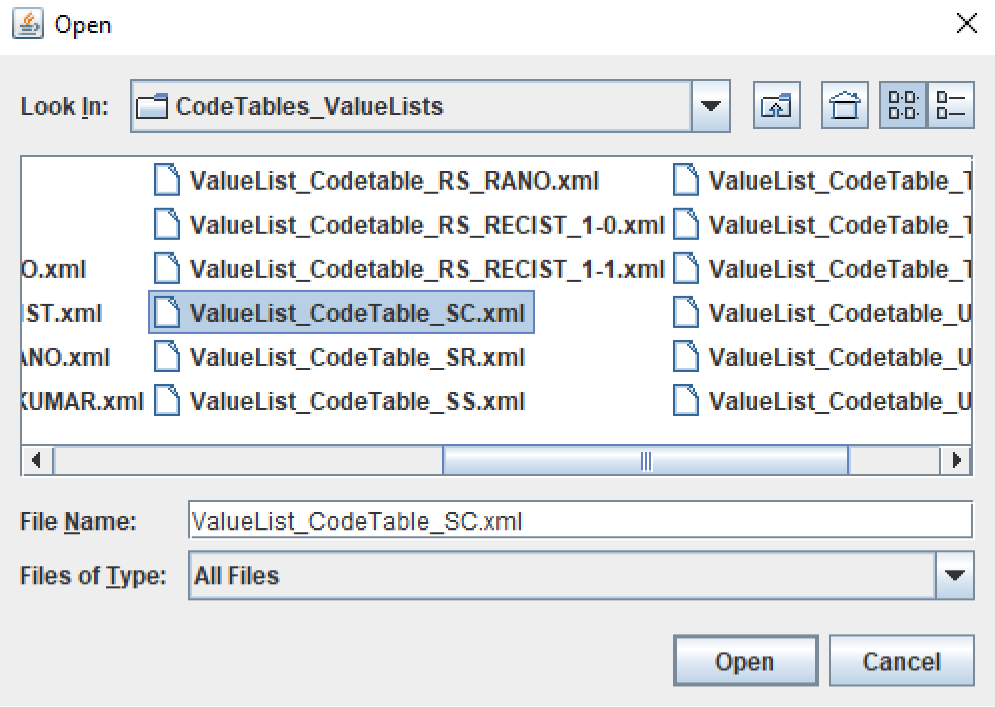
with all the corresponding "WhereClause-s" and "ValueList-ItemDefs".

This pre-defined ValueList can then be used for e.g. SCSTRESC (Character Result/Finding in Std Format) by using the menu "Insert - ValueLists for CDISC CodeTables from File into define.xml":


Explore
Insert
Transform
Validate
CDISC Library
Options
About
Global Variables from ODM into define.xml
MeasurementUnit definitions from ODM into define.xml
All CodeList definitions from ODM into define.xml
Selected CodeList definitions from ODM into define.xml
CodeList definitions from File into define.xml
ValueLists for CDISC CodeTables from File into define.xml
Create new SDTM CodeList from existing CodeList
Crea
Create new SDTM Sponsor-defined CodeList
roups)
Variable
Varia
STUDYID
DOMAIN

Load ValueLists, WhereClauses, CodeLists for CDISC CodeTables from file into define.xml

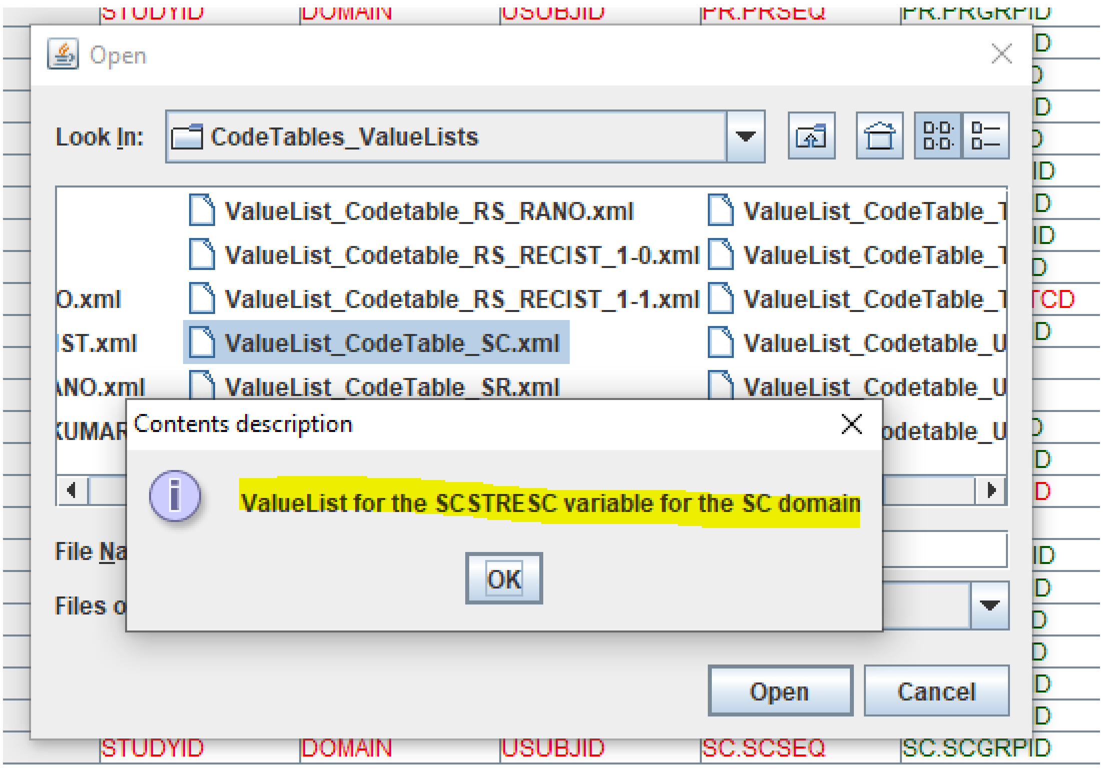
and then selecting e.g. "ValueList_CodeTable_SC":


Open
Look In: CodeTables_ValueLists
ValueList_Codetable_RS_RANO.xml
ValueList_Codetable_RS_RECIST_1-0.xml
ValueList_Codetable_RS_RECIST_1-1.xml
ValueList_CodeTable_SC.xml
ValueList_CodeTable_SR.xml
ValueList_CodeTable_SS.xml
ValueList_CodeTable_T
ValueList_CodeTable_T
ValueList_CodeTable_T
ValueList_Codetable_U
ValueList_Codetable_U
ValueList_Codetable_U
O.xml
ST.xml
NO.xml
UMAR.xml
Open
Cancel

File Name: ValueList_CodeTable_SC.xml

Files of Type: All Files

which will pop up an explanation message about what the ValueList is about:

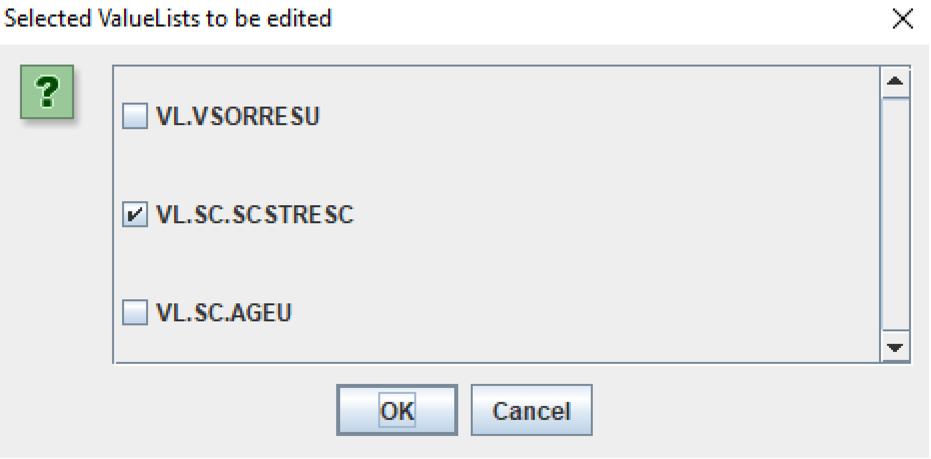


Open
Look In: CodeTables_ValueLists
ValueList_Codetable_RS_RANO.xml
ValueList_Codetable_RS_RECIST_1-0.xml
ValueList_Codetable_RS_RECIST_1-1.xml
ValueList_CodeTable_SC.xml
ValueList_CodeTable_SR.xml
File Na
Files o
Open
Cancel

Contents description

ValueList for the SCSTRESC variable for the SC domain

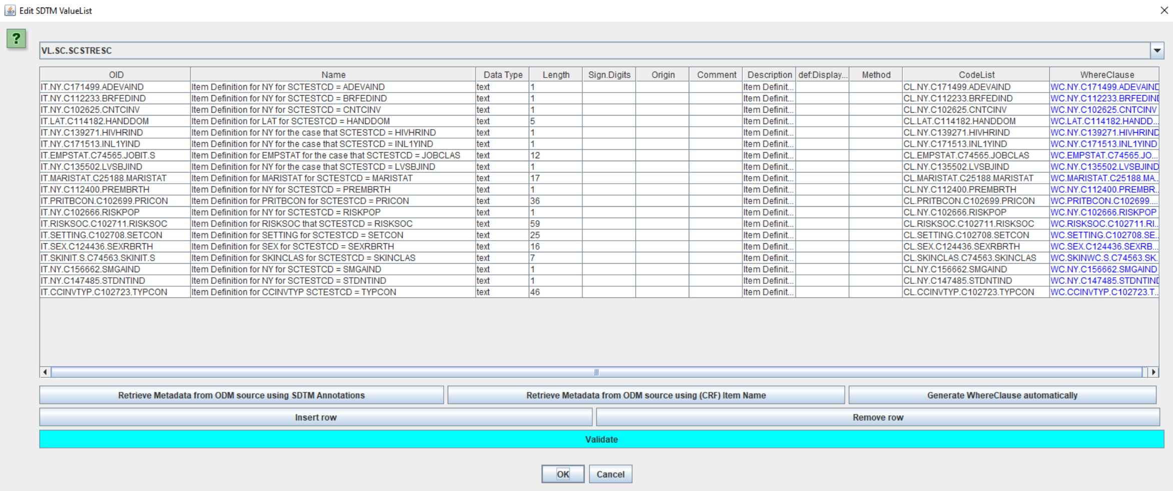
when than confirming the choice, the ValueList is loaded, and the system proposes it to have it edited by the user:



Selected ValueLists to be edited

- ☐ VL.VSORRESU
- ☑ VL.SC.SCSTRESC
- ☐ VL.SC.AGEU

when then clicking "OK", the "ValueList Wizard" opens up (this will take a few moments …):



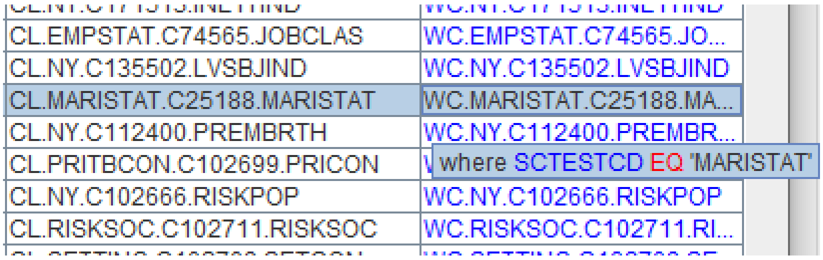
Edit SDTM ValueList

VL.SC.SCSTRESC

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:Display...	Method	CodeList	WhereClause
IT.NY.C171499.ADEVAIND	Item Definition for NY for SCTESTCD = ADEVAIND	text	1				Item Definit...			CL.NY.C171499.ADEVAIND	WC.NY.C171499.ADEVAIND
IT.NY.C112233.BRFEDIND	Item Definition for NY for SCTESTCD = BRFEDIND	text	1				Item Definit...			CL.NY.C112233.BRFEDIND	WC.NY.C112233.BRFEDIND
IT.NY.C102625.CNTCINV	Item Definition for NY for SCTESTCD = CNTCINV	text	1				Item Definit...			CL.NY.C102625.CNTCINV	WC.NY.C102625.CNTCINV
IT.LAT.C114182.HANDDOM	Item Definition for LAT for SCTESTCD = HANDDOM	text	5				Item Definit...			CL.LAT.C114182.HANDDOM	WC.LAT.C114182.HANDD...
IT.NY.C139271.HIVHRIND	Item Definition for NY for the case that SCTESTCD = HIVHRIND	text	1				Item Definit...			CL.NY.C139271.HIVHRIND	WC.NY.C139271.HIVHRIND
IT.NY.C171513.INL1YIND	Item Definition for NY for the case that SCTESTCD = INL1YIND	text	1				Item Definit...			CL.NY.C171513.INL1YIND	WC.NY.C171513.INL1YIND
IT.EMPSTAT.C74565.JOBIT.S	Item Definition for EMPSTAT for the case that SCTESTCD = JOBCLAS	text	12				Item Definit...			CL.EMPSTAT.C74565.JOBCLAS	WC.EMPSTAT.C74565.JO...
IT.NY.C135502.LVSBJIND	Item Definition for NY for the case that SCTESTCD = LVSBJIND	text	1				Item Definit...			CL.NY.C135502.LVSBJIND	WC.NY.C135502.LVSBJIND
IT.MARISTAT.C25188.MARISTAT	Item Definition for MARISTAT for SCTESTCD = MARISTAT	text	17				Item Definit...			CL.MARISTAT.C25188.MARISTAT	WC.MARISTAT.C25188.MA...
IT.NY.C112400.PREMBRTH	Item Definition for NY for SCTESTCD = PREMBRTH	text	1				Item Definit...			CL.NY.C112400.PREMBRTH	WC.NY.C112400.PREMBR...
IT.PRITBCON.C102699.PRICON	Item Definition for PRITBCON for SCTESTCD = PRICON	text	36				Item Definit...			CL.PRITBCON.C102699.PRICON	WC.PRITBCON.C102699....
IT.NY.C102666.RISKPOP	Item Definition for NY for SCTESTCD = RISKPOP	text	1				Item Definit...			CL.NY.C102666.RISKPOP	WC.NY.C102666.RISKPOP
IT.RISKSOC.C102711.RISKSOC	Item Definition for RISKSOC that SCTESTCD = RISKSOC	text	59				Item Definit...			CL.RISKSOC.C102711.RISKSOC	WC.RISKSOC.C102711.RI...
IT.SETTING.C102708.SETCON	Item Definition for SETTING for SCTESTCD = SETCON	text	25				Item Definit...			CL.SETTING.C102708.SETCON	WC.SETTING.C102708.SE...
IT.SEX.C124436.SEXRBRTH	Item Definition for SEX for SCTESTCD = SEXRBRTH	text	16				Item Definit...			CL.SEX.C124436.SEXRBRTH	WC.SEX.C124436.SEXRB...
IT.SKINIT.S.C74563.SKINIT.S	Item Definition for SKINCLAS for SCTESTCD = SKINCLAS	text	7				Item Definit...			CL.SKINCLAS.C74563.SKINCLAS	WC.SKINWC.S.C74563.SK...
IT.NY.C156662.SMGAIND	Item Definition for NY for SCTESTCD = SMGAIND	text	1				Item Definit...			CL.NY.C156662.SMGAIND	WC.NY.C156662.SMGAIND
IT.NY.C147485.STDNTIND	Item Definition for NY for SCTESTCD = STDNTIND	text	1				Item Definit...			CL.NY.C147485.STDNTIND	WC.NY.C147485.STDNTIND
IT.CCINVTYP.C102723.TYPCON	Item Definition for CCINVTYP SCTESTCD = TYPCON	text	46				Item Definit...			CL.CCINVTYP.C102723.TYPCON	WC.CCINVTYP.C102723.T...

Retrieve Metadata from ODM source using SDTM Annotations
Retrieve Metadata from ODM source using (CRF) Item Name
Generate WhereClause automatically
Insert row
Remove row
Validate
OK
Cancel

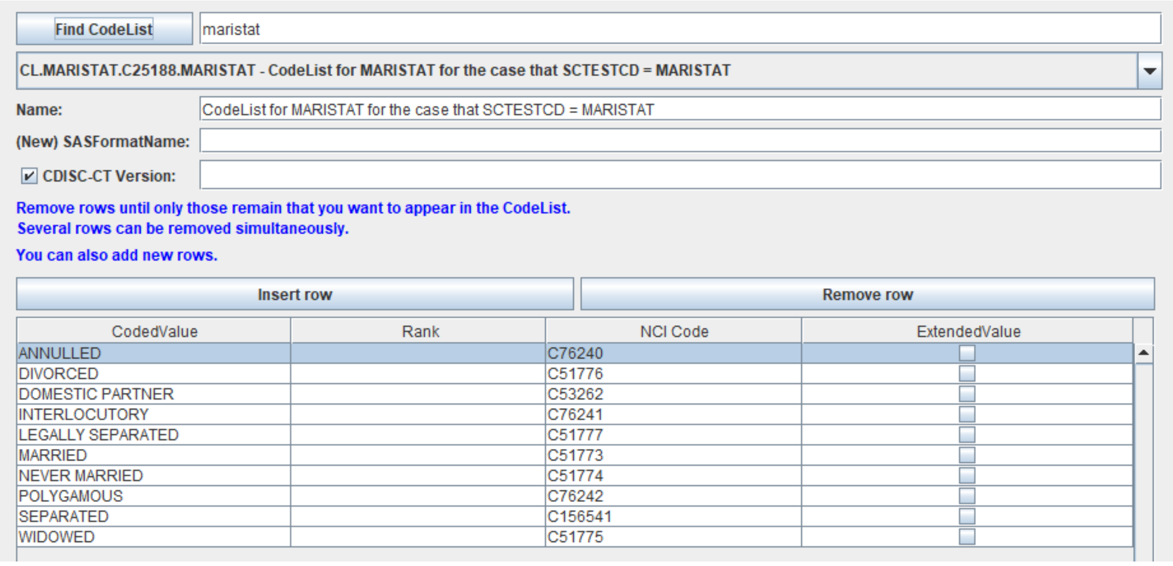
with all the "WhereClause-s" already predefined. For example, for SCTESTCD=MARISTAT (Marital Status):



CL.NY.C171513.INL1YIND	WC.NY.C171513.INL1YIND
CL.EMPSTAT.C74565.JOBCLAS	WC.EMPSTAT.C74565.JO...
CL.NY.C135502.LVSBJIND	WC.NY.C135502.LVSBJIND
CL.MARISTAT.C25188.MARISTAT	WC.MARISTAT.C25188.MA...
CL.NY.C112400.PREMBRTH	WC.NY.C112400.PREMBR...
CL.PRITBCON.C102699.PRICON	W
CL.NY.C102666.RISKPOP	WC.NY.C102666.RISKPOP
CL.RISKSOC.C102711.RISKSOC	WC.RISKSOC.C102711.RI...
CL.SETTING.C102708.SETCON	WC.SETTING.C102708.SE

where SCTESTCD EQ 'MARISTAT'

and also the associated codelists already attached, e.g. for "Marital Status" (CDISC codelist C25188):



Find CodeList: maristat

CL.MARISTAT.C25188.MARISTAT - CodeList for MARISTAT for the case that SCTESTCD = MARISTAT

Name: CodeList for MARISTAT for the case that SCTESTCD = MARISTAT

(New) SASFormatName:

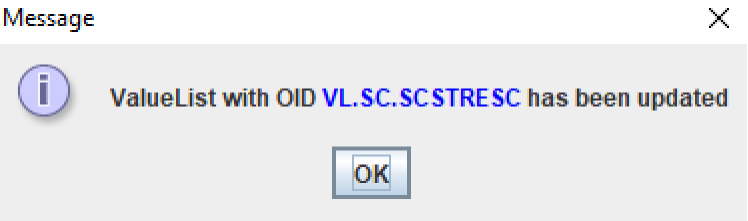
☑ CDISC-CT Version:

Remove rows until only those remain that you want to appear in the CodeList.
Several rows can be removed simultaneously.

You can also add new rows.

CodedValue	Rank	NCI Code	ExtendedValue
ANNULLED		C76240	☐
DIVORCED		C51776	☐
DOMESTIC PARTNER		C53262	☐
INTERLOCUTORY		C76241	☐
LEGALLY SEPARATED		C51777	☐
MARRIED		C51773	☐
NEVER MARRIED		C51774	☐
POLYGAMOUS		C76242	☐
SEPARATED		C156541	☐
WIDOWED		C51775	☐

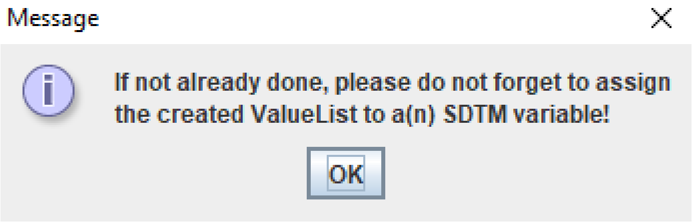
One can then remove rows that one does not need for the current status, and only keep the ones that correspond to tests that have really been used in the study.
One can also add additional rows with new "cases", but this will be seldomly the case.
When then clicking "OK", a message is displayed:



Message

ValueList with OID VL.SC.SCSTRESC has been updated

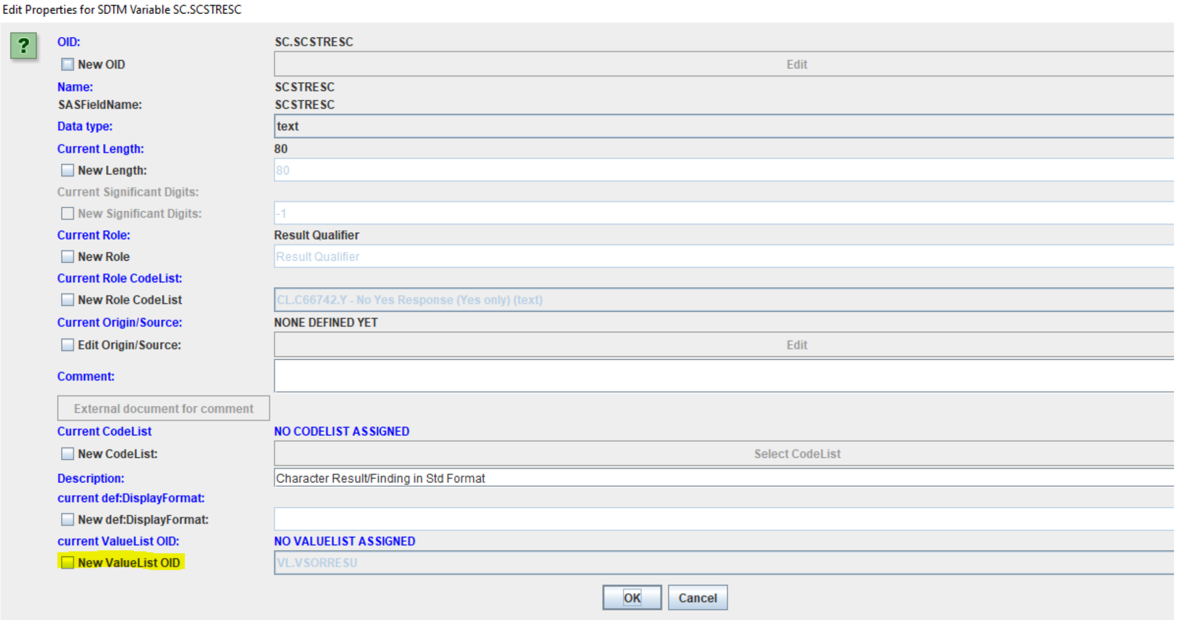
followed by the message:



Message

If not already done, please do not forget to assign the created ValueList to a(n) SDTM variable!

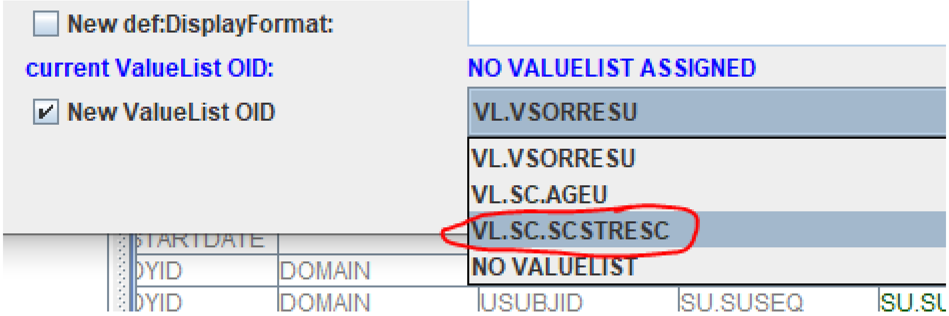
We can now already decide to assign the (probably adapted) ValueList from CDISC "Code Table" to SCSTRESC already now, or do it later. In the former case, select "SCSTRESC" and use the menu "Edit - SDTM Variable Properties" (Ctrl-E):



Edit Properties for SDTM Variable SC.SCSTRESC

OID:	SC.SCSTRESC
□ New OID	Edit
Name:	SCSTRESC
SASFieldName:	SCSTRESC
Data type:	text
Current Length:	80
□ New Length:	80
Current Significant Digits:	
□ New Significant Digits:	-1
Current Role:	Result Qualifier
□ New Role	Result Qualifier
Current Role CodeList:	
□ New Role CodeList	CL.C66742.Y - No Yes Response (Yes only) (text)
Current Origin/Source:	NONE DEFINED YET
□ Edit Origin/Source:	Edit
Comment:	
External document for comment	
Current CodeList	NO CODELIST ASSIGNED
□ New CodeList:	Select CodeList
Description:	Character Result/Finding in Std Format
current def:DisplayFormat:	
□ New def:DisplayFormat:	
current ValueList OID:	NO VALUELIST ASSIGNED
□ New ValueList OID	VL.VSORRESU

and look for "New ValueList OID", and then select the one just generated from the CDISC "Code Table":

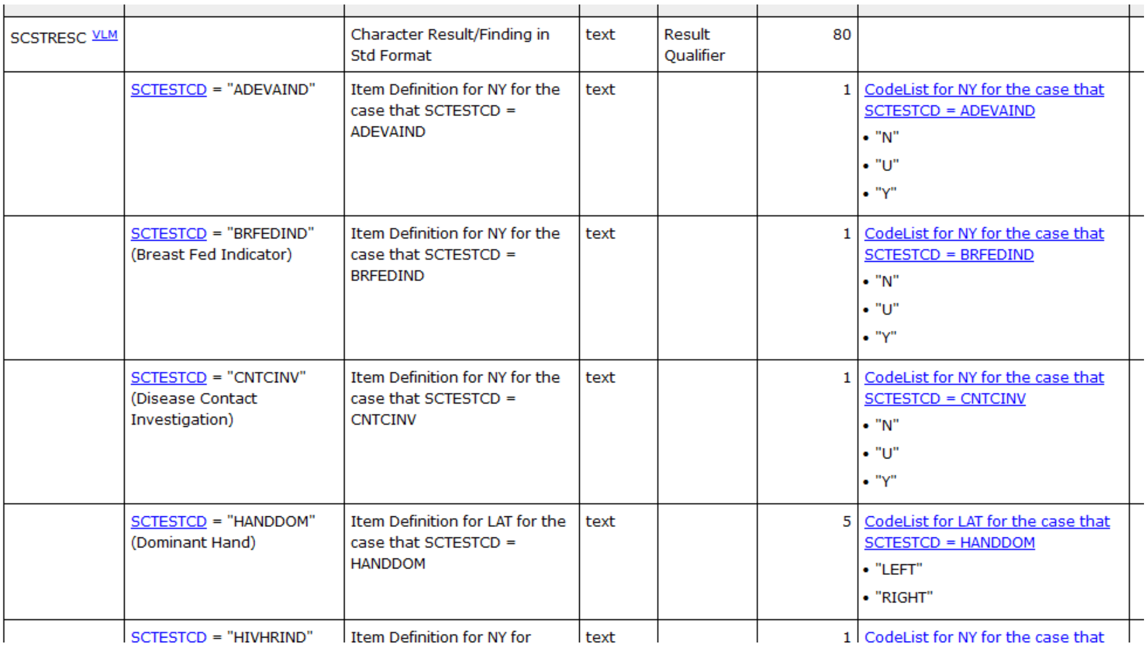


□ New def:DisplayFormat:

current ValueList OID: NO VALUELIST ASSIGNED

☑ New ValueList OID VL.VSORRESU

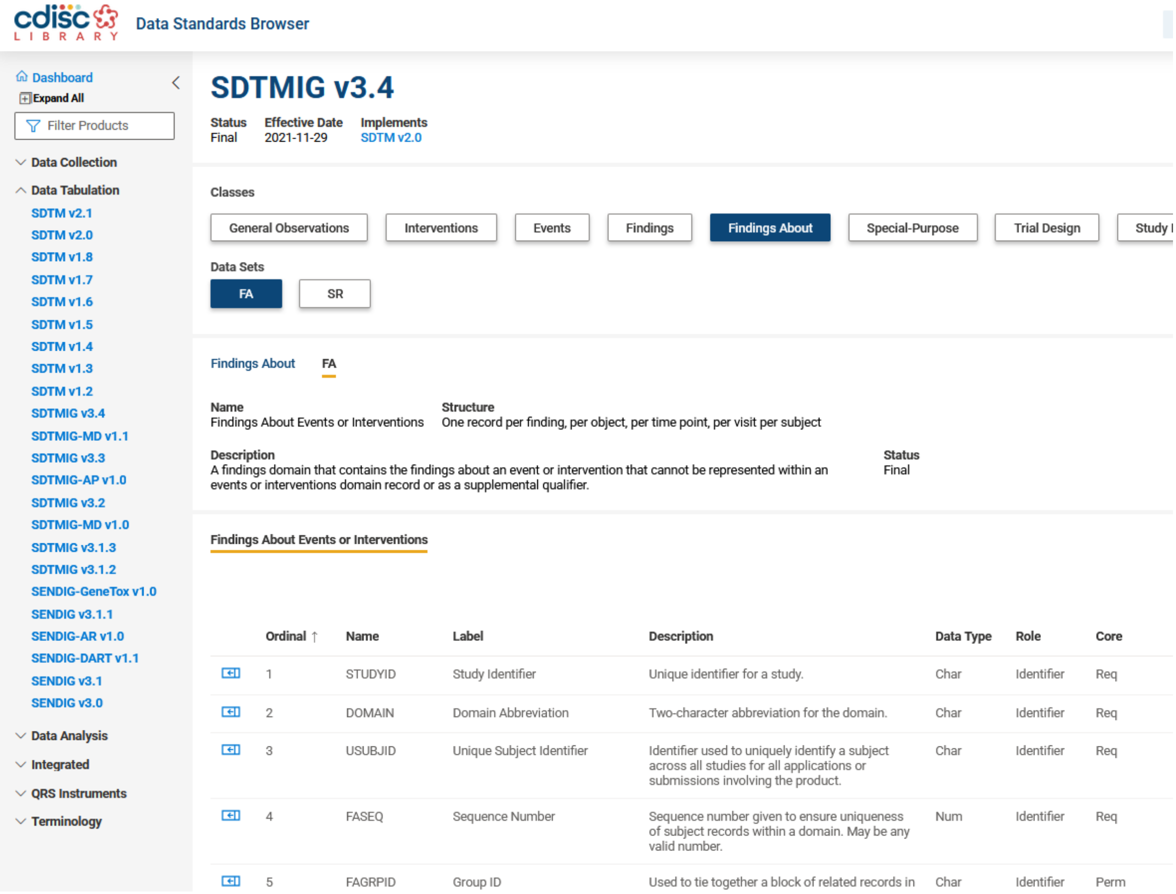
Using "View - View define.xml in browser", one then e.g. finds:



SCSTRESC VLM		Character Result/Finding in Std Format	text	Result Qualifier	80	
	SCTESTCD = "ADEVAIND"	Item Definition for NY for the case that SCTESTCD = ADEVAIND	text		1	CodeList for NY for the case that SCTESTCD = ADEVAIND • "N" • "U" • "Y"
	SCTESTCD = "BRFEDIND" (Breast Fed Indicator)	Item Definition for NY for the case that SCTESTCD = BRFEDIND	text		1	CodeList for NY for the case that SCTESTCD = BRFEDIND • "N" • "U" • "Y"
	SCTESTCD = "CNTCINV" (Disease Contact Investigation)	Item Definition for NY for the case that SCTESTCD = CNTCINV	text		1	CodeList for NY for the case that SCTESTCD = CNTCINV • "N" • "U" • "Y"
	SCTESTCD = "HANDDOM" (Dominant Hand)	Item Definition for LAT for the case that SCTESTCD = HANDDOM	text		5	CodeList for LAT for the case that SCTESTCD = HANDDOM • "LEFT" • "RIGHT"
	SCTESTCD = "HIVHRIND"	Item Definition for NY for	text		1	CodeList for NY for the case that

Using FA (Findings About) Specializations

One of the most curious SDTM domains surely is FA (Findings About Events or Interventions). When looking it up in the "CDISC Library Browser", we find:



SDTMIG v3.4

Status: Final
Effective Date: 2021-11-29
Implements: SDTM v2.0

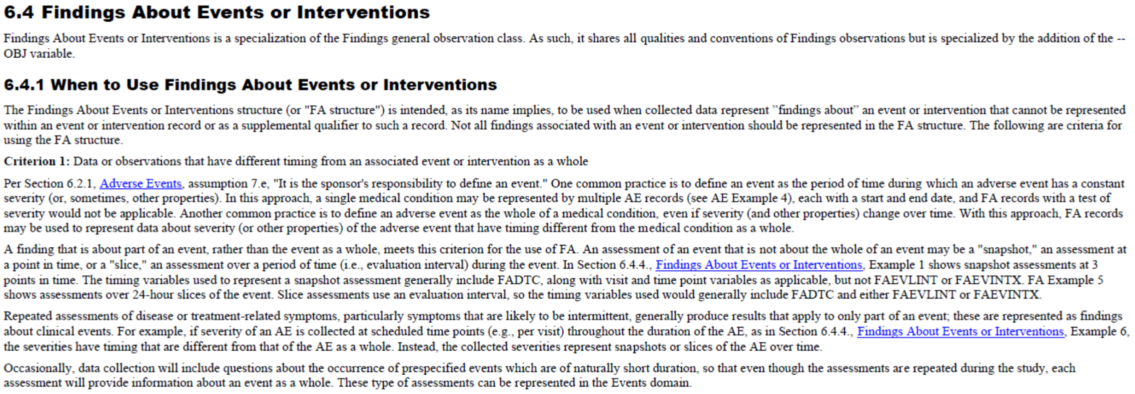
Name: Findings About Events or Interventions
Structure: One record per finding, per object, per time point, per visit per subject

Description: A findings domain that contains the findings about an event or intervention that cannot be represented within an events or interventions domain record or as a supplemental qualifier.

Status: Final

Ordinal ↑	Name	Label	Description	Data Type	Role	Core
1	STUDYID	Study Identifier	Unique identifier for a study.	Char	Identifier	Req
2	DOMAIN	Domain Abbreviation	Two-character abbreviation for the domain.	Char	Identifier	Req
3	USUBJID	Unique Subject Identifier	Identifier used to uniquely identify a subject across all studies for all applications or submissions involving the product.	Char	Identifier	Req
4	FASEQ	Sequence Number	Sequence number given to ensure uniqueness of subject records within a domain. May be any valid number.	Num	Identifier	Req
5	FAGRPID	Group ID	Used to tie together a block of related records in	Char	Identifier	Perm

and in the SDTMIG we read:



6.4 Findings About Events or Interventions

Findings About Events or Interventions is a specialization of the Findings general observation class. As such, it shares all qualities and conventions of Findings observations but is specialized by the addition of the --OBJ variable.

6.4.1 When to Use Findings About Events or Interventions

The Findings About Events or Interventions structure (or "FA structure") is intended, as its name implies, to be used when collected data represent "findings about" an event or intervention that cannot be represented within an event or intervention record or as a supplemental qualifier to such a record. Not all findings associated with an event or intervention should be represented in the FA structure. The following are criteria for using the FA structure.

Criterion 1: Data or observations that have different timing from an associated event or intervention as a whole

Per Section 6.2.1, Adverse Events, assumption 7.e, "It is the sponsor's responsibility to define an event." One common practice is to define an event as the period of time during which an adverse event has a constant severity (or, sometimes, other properties). In this approach, a single medical condition may be represented by multiple AE records (see AE Example 4), each with a start and end date, and FA records with a test of severity would not be applicable. Another common practice is to define an adverse event as the whole of a medical condition, even if severity (and other properties) change over time. With this approach, FA records may be used to represent data about severity (or other properties) of the adverse event that have timing different from the medical condition as a whole.

A finding that is about part of an event, rather than the event as a whole, meets this criterion for the use of FA. An assessment of an event that is not about the whole of an event may be a "snapshot," an assessment at a point in time, or a "slice," an assessment over a period of time (i.e., evaluation interval) during the event. In Section 6.4.4., Findings About Events or Interventions, Example 1 shows snapshot assessments at 3 points in time. The timing variables used to represent a snapshot assessment generally include FADTC, along with visit and time point variables as applicable, but not FAEVLINT or FAEVINTX. FA Example 5 shows assessments over 24-hour slices of the event. Slice assessments use an evaluation interval, so the timing variables used would generally include FADTC and either FAEVLINT or FAEVINTX.

Repeated assessments of disease or treatment-related symptoms, particularly symptoms that are likely to be intermittent, generally produce results that apply to only part of an event; these are represented as findings about clinical events. For example, if severity of an AE is collected at scheduled time points (e.g., per visit) throughout the duration of the AE, as in Section 6.4.4., Findings About Events or Interventions, Example 6, the severities have timing that are different from that of the AE as a whole. Instead, the collected severities represent snapshots or slices of the AE over time.

Occasionally, data collection will include questions about the occurrence of prespecified events which are of naturally short duration, so that even though the assessments are repeated during the study, each assessment will provide information about an event as a whole. These type of assessments can be represented in the Events domain.

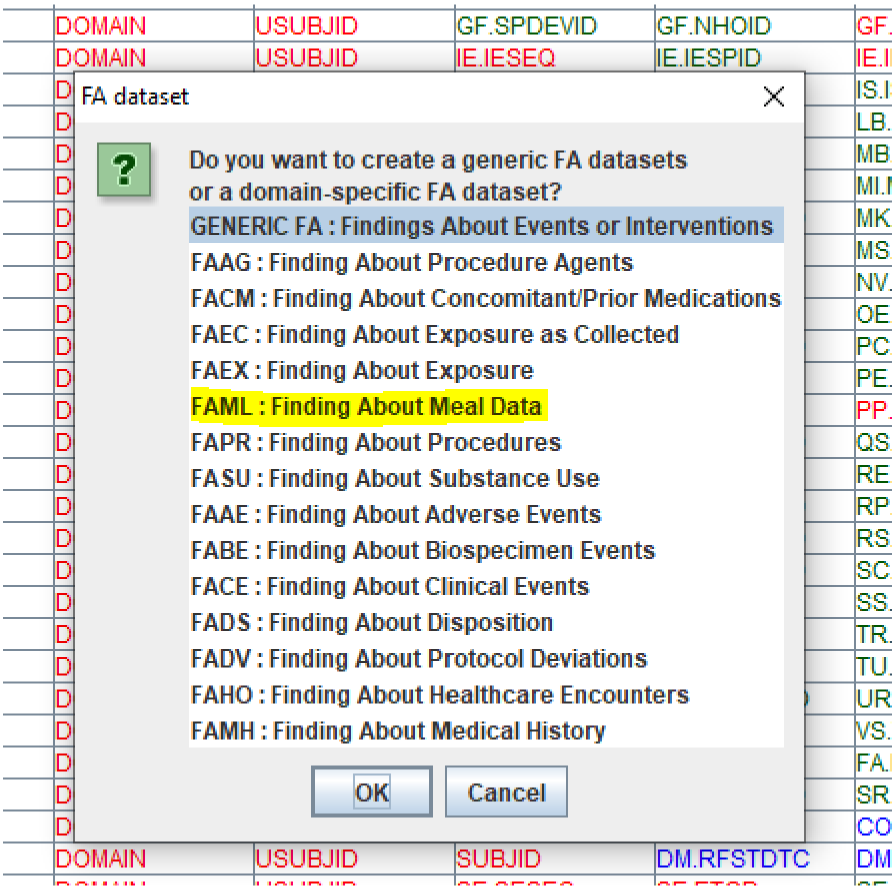
Essentially, it is yet another way (besides Non-standard variables (NSVs) and RELREC) to connect observations or test results to either events (such as AEs) or interventions (such as effects of concomitant medications) when the user believes there is no other way to do so[4].

So, when using "FA", one will usually decide to have "specialized" instances, such as FAAE (Findings About Adverse Events), FAML (Findings About Meal Data), etc..
For each of these, different codelists will apply for many of the variables, which can be hard to find out.
Fortunately, there are the CDISC "CT-Relations", which help to decide which codelists etc to assign for a specific FA "instance".

For example, suppose we want to set up mappings for FAML (Findings About Meal Data), where ML is an "Interventions" domain.

When doing "drag-and-drop" from the template of the FA row towards the bottom, the system reacts with:

[4] It is very regrettable that there are so many different ways to define relations between data points in SDTM. This surely is related the the idea that "everything is a table", due to the mandated use of SAS Transport files, and the fact that SDTM tables cannot be regarded as "relational tables", using "foreign keys" to define the relations.

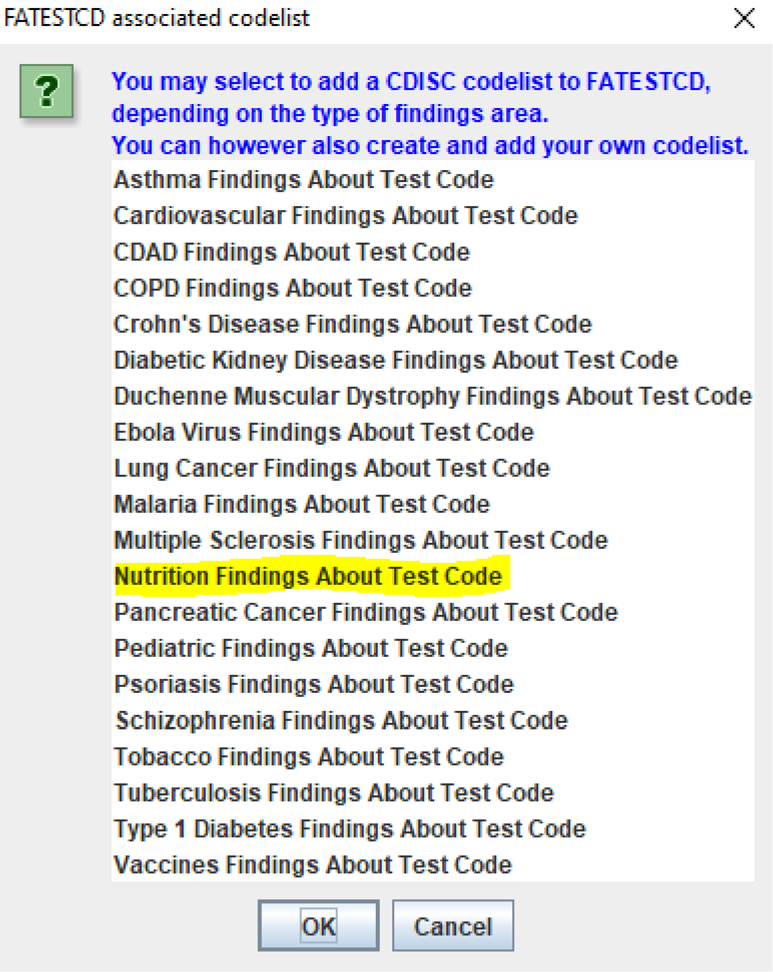


FA dataset

Do you want to create a generic FA datasets
or a domain-specific FA dataset?

GENERIC FA : Findings About Events or Interventions
FAAG : Finding About Procedure Agents
FACM : Finding About Concomitant/Prior Medications
FAEC : Finding About Exposure as Collected
FAEX : Finding About Exposure
FAML : Finding About Meal Data
FAPR : Finding About Procedures
FASU : Finding About Substance Use
FAAE : Finding About Adverse Events
FABE : Finding About Biospecimen Events
FACE : Finding About Clinical Events
FADS : Finding About Disposition
FADV : Finding About Protocol Deviations
FAHO : Finding About Healthcare Encounters
FAMH : Finding About Medical History

allowing us to opt either for a "generic" FA dataset definition, or a "specific" FA instance. When then selecting "FAML: Findings About Meal Data" and click "OK", we get after some time (a "please wait" progress bar" is displayed):

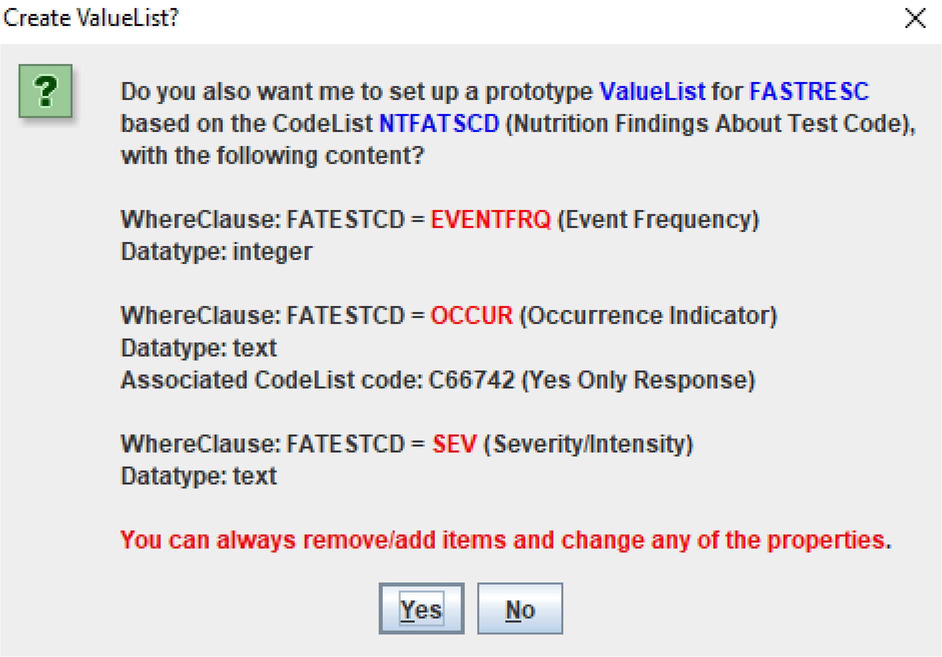


FATESTCD associated codelist

You may select to add a CDISC codelist to FATESTCD, depending on the type of findings area.
You can however also create and add your own codelist.

Asthma Findings About Test Code
Cardiovascular Findings About Test Code
CDAD Findings About Test Code
COPD Findings About Test Code
Crohn's Disease Findings About Test Code
Diabetic Kidney Disease Findings About Test Code
Duchenne Muscular Dystrophy Findings About Test Code
Ebola Virus Findings About Test Code
Lung Cancer Findings About Test Code
Malaria Findings About Test Code
Multiple Sclerosis Findings About Test Code
Nutrition Findings About Test Code
Pancreatic Cancer Findings About Test Code
Pediatric Findings About Test Code
Psoriasis Findings About Test Code
Schizophrenia Findings About Test Code
Tobacco Findings About Test Code
Tuberculosis Findings About Test Code
Type 1 Diabetes Findings About Test Code
Vaccines Findings About Test Code

allowing us to assign a codelist to FATESTCD. In the current case, an obvious candidate is the "Nutrition Findings About Test Code". When selecting it, and then clicking OK, this codelist is assigned to FATESTCD, and the system is asking whether it would automatically generate a ValueList for FASTRESC (standardized result):



Create ValueList?

Do you also want me to set up a prototype ValueList for FASTRESC based on the CodeList NTFATSCD (Nutrition Findings About Test Code), with the following content?

WhereClause: FATESTCD = EVENTFRQ (Event Frequency)
Datatype: integer

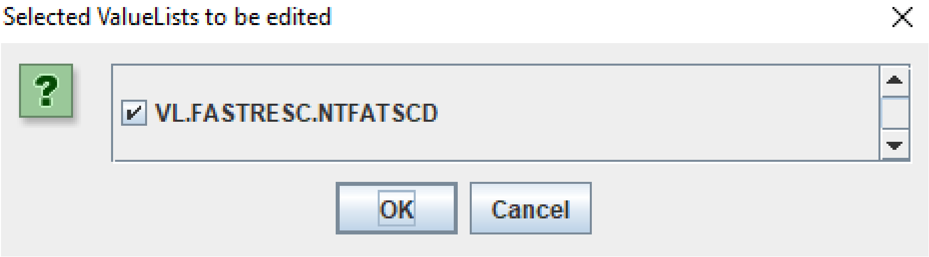
WhereClause: FATESTCD = OCCUR (Occurrence Indicator)
Datatype: text
Associated CodeList code: C66742 (Yes Only Response)

WhereClause: FATESTCD = SEV (Severity/Intensity)
Datatype: text

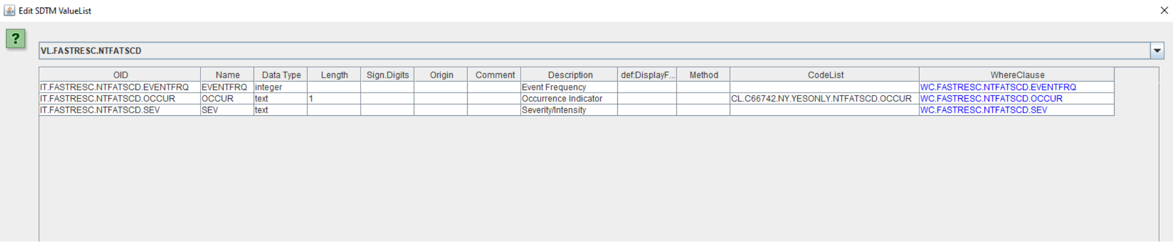
You can always remove/add items and change any of the properties.

meaning that different data types, lengths and codelists will be assigned to FASTRESC dependent on the value of FATESTCD. For example, for FATESTCD=EVENTFRQ (Event Frequency), the data type of ValueList-FASTRESC will be "integer" whereas for the case of "OCCUR" (Occurrence Indicator), a "Yes-only" codelist will be associated.

When then clicking OK, this leads to:

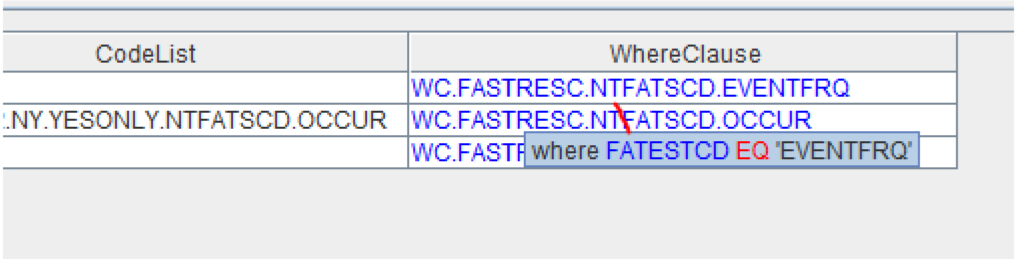


proposing to further adapt the ValueList. When clicking "OK", after a few moments the "ValueList Editor" is provided, allowing to make further adjustments when necessary:



OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:DisplayF...	Method	CodeList	WhereClause
IT.FASTRESC.NTFATSCD.EVENTFRQ	EVENTFRQ	integer					Event Frequency				WC.FASTRESC.NTFATSCD.EVENTFRQ
IT.FASTRESC.NTFATSCD.OCCUR	OCCUR	text	1				Occurrence Indicator			CL.C66742.NY.YESONLY.NTFATSCD.OCCUR	WC.FASTRESC.NTFATSCD.OCCUR
IT.FASTRESC.NTFATSCD.SEV	SEV	text					Severity/Intensity				WC.FASTRESC.NTFATSCD.SEV

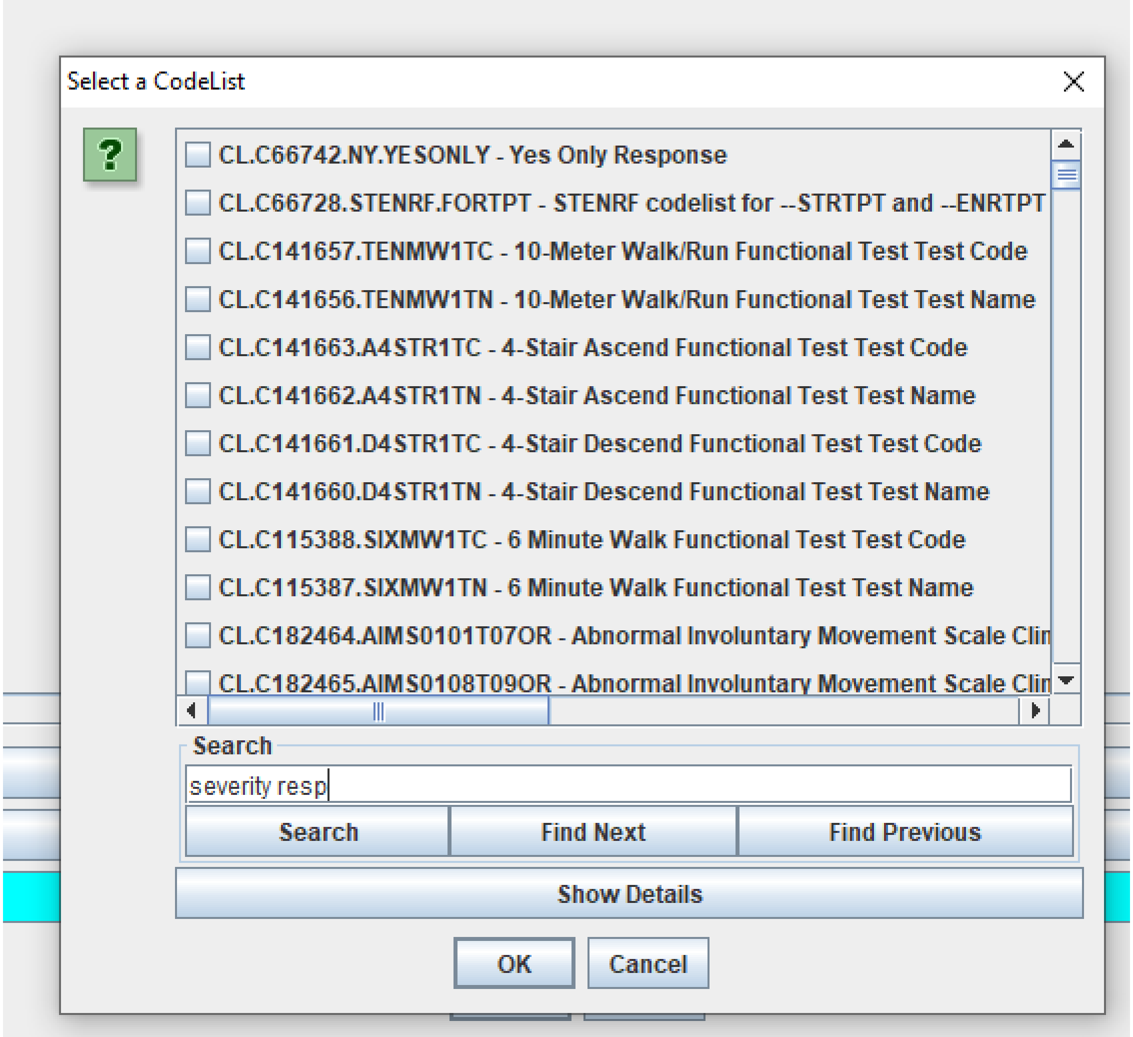
even with pre-defined "WhereClause-s", such as:



CodeList	WhereClause
	WC.FASTRESC.NTFATSCD.EVENTFRQ
.NY.YESONLY.NTFATSCD.OCCUR	WC.FASTRESC.NTFATSCD.OCCUR
	WC.FASTF

where FATESTCD EQ 'EVENTFRQ'

Remark that the "Length" has not been assigned yet to the items that do not have a codelist (EVENTFRQ, SEV). One can also add a codelist to "SEV" (Severity) as the values for it are often enumerated. So clicking on the cell "CodeList" for "SEV" will open a new dialog:



□ CL.C66742.NY.YESONLY - Yes Only Response

□ CL.C66728.STENRF.FORTPT - STENRF codelist for --STRTPT and --ENRTPT

□ CL.C141657.TENMW1TC - 10-Meter Walk/Run Functional Test Test Code

□ CL.C141656.TENMW1TN - 10-Meter Walk/Run Functional Test Test Name

□ CL.C141663.A4STR1TC - 4-Stair Ascend Functional Test Test Code

□ CL.C141662.A4STR1TN - 4-Stair Ascend Functional Test Test Name

□ CL.C141661.D4STR1TC - 4-Stair Descend Functional Test Test Code

□ CL.C141660.D4STR1TN - 4-Stair Descend Functional Test Test Name

□ CL.C115388.SIXMW1TC - 6 Minute Walk Functional Test Test Code

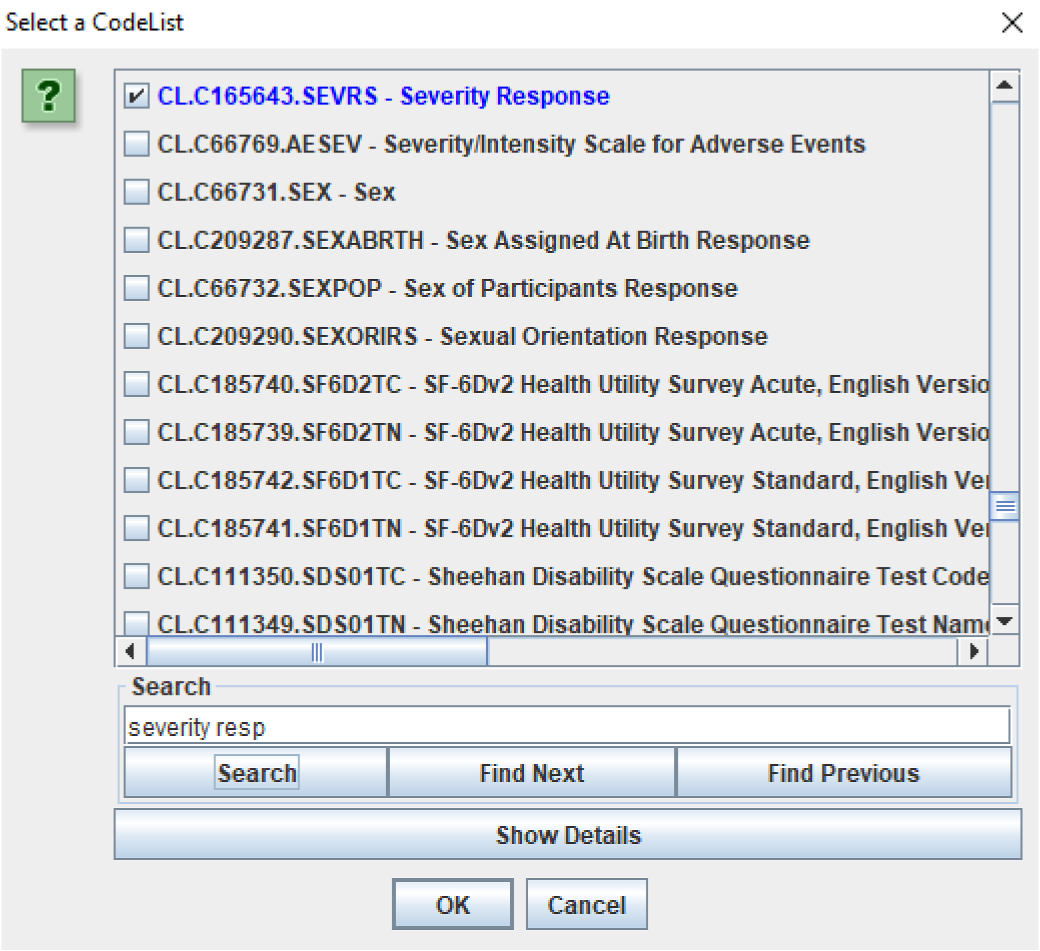
□ CL.C115387.SIXMW1TN - 6 Minute Walk Functional Test Test Name

□ CL.C182464.AIMS0101T07OR - Abnormal Involuntary Movement Scale Clin

□ CL.C182465.AIMS0108T09OR - Abnormal Involuntary Movement Scale Clin

Search: severity resp

and can search for a codelist using the "Search" button, e.g. "severity resp", which will show:



☑ CL.C165643.SEVRS - Severity Response

□ CL.C66769.AESEV - Severity/Intensity Scale for Adverse Events

□ CL.C66731.SEX - Sex

□ CL.C209287.SEXABRTH - Sex Assigned At Birth Response

□ CL.C66732.SEXPOP - Sex of Participants Response

□ CL.C209290.SEXORIRS - Sexual Orientation Response

□ CL.C185740.SF6D2TC - SF-6Dv2 Health Utility Survey Acute, English Versio

□ CL.C185739.SF6D2TN - SF-6Dv2 Health Utility Survey Acute, English Versio

□ CL.C185742.SF6D1TC - SF-6Dv2 Health Utility Survey Standard, English Ve

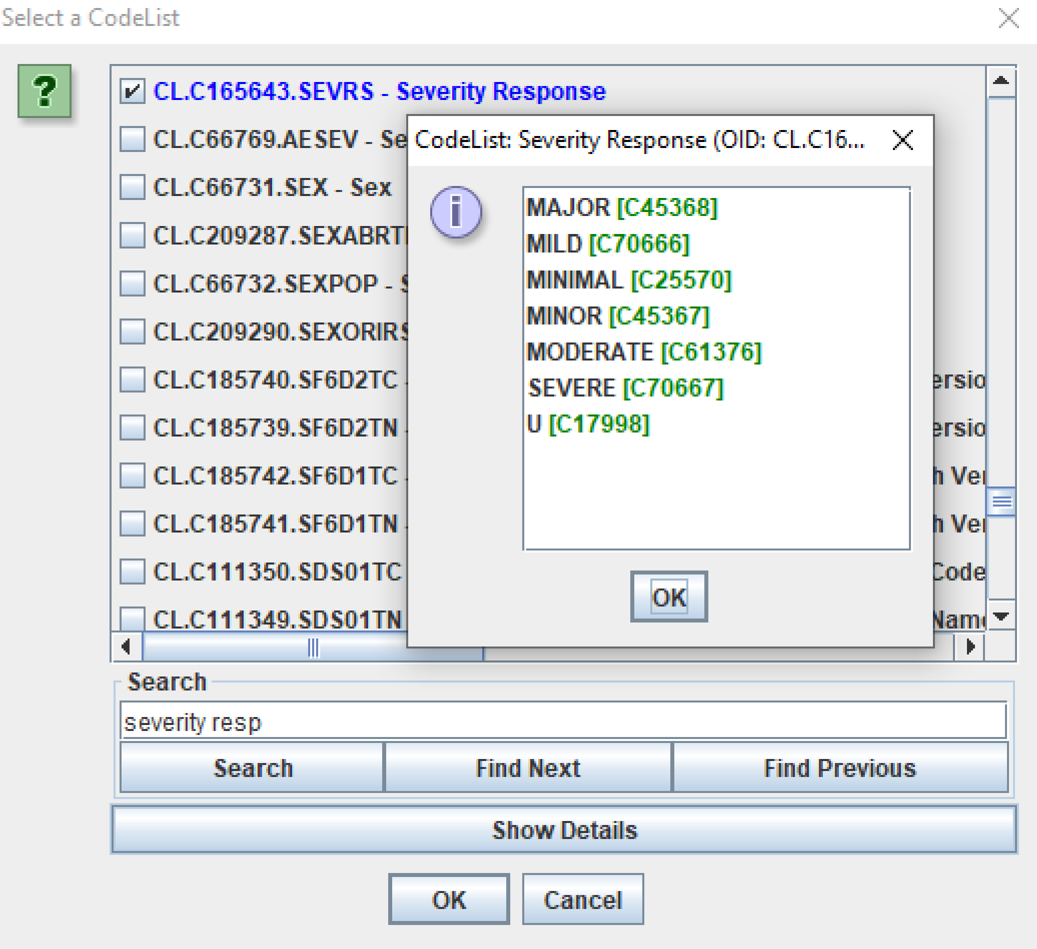
□ CL.C185741.SF6D1TN - SF-6Dv2 Health Utility Survey Standard, English Ve

□ CL.C111350.SDS01TC - Sheehan Disability Scale Questionnaire Test Code

□ CL.C111349.SDS01TN - Sheehan Disability Scale Questionnaire Test Nam

Search: severity resp

Details of this codelist can be found by clicking "Show Details", leading to:



Select a CodeList

☑ CL.C165643.SEVRS - Severity Response

□ CL.C66769.AESEV - Se

□ CL.C66731.SEX - Sex

□ CL.C209287.SEXABRT

□ CL.C66732.SEXPOP - S

□ CL.C209290.SEXORIRS

□ CL.C185740.SF6D2TC

□ CL.C185739.SF6D2TN

□ CL.C185742.SF6D1TC

□ CL.C185741.SF6D1TN

□ CL.C111350.SDS01TC

□ CL.C111349.SDS01TN

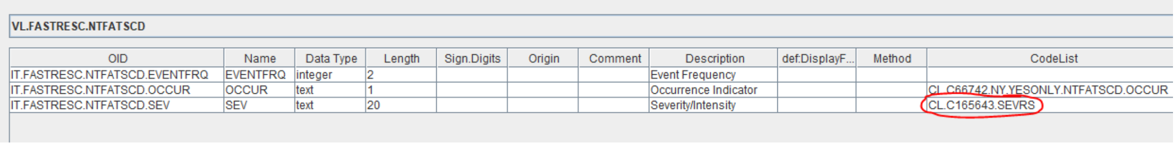
CodeList: Severity Response (OID: CL.C16...

MAJOR [C45368]
MILD [C70666]
MINIMAL [C25570]
MINOR [C45367]
MODERATE [C61376]
SEVERE [C70667]
U [C17998]

Search: severity resp

If one wants to use this codelist for the case of "SEV" (Severity), one clicks OK twice, which will assign this codelist to the ValueList-variable "SEV" for FASTRESC.
Remark that one can later still always subset this codelist, or remove or add items.

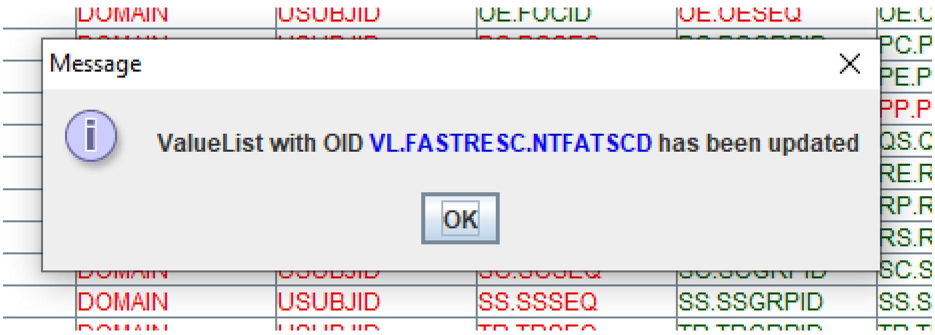
The result in the table then is:



VL.FASTRESC.NTFATSCD

OID	Name	Data Type	Length	Sign.Digits	Origin	Comment	Description	def:DisplayF...	Method	CodeList
IT.FASTRESC.NTFATSCD.EVENTFRQ	EVENTFRQ	integer	2				Event Frequency			
IT.FASTRESC.NTFATSCD.OCCUR	OCCUR	text	1				Occurrence Indicator			CL.C66742.NY.YESONLY.NTFATSCD.OCCUR
IT.FASTRESC.NTFATSCD.SEV	SEV	text	20				Severity/Intensity			CL.C165643.SEVRS

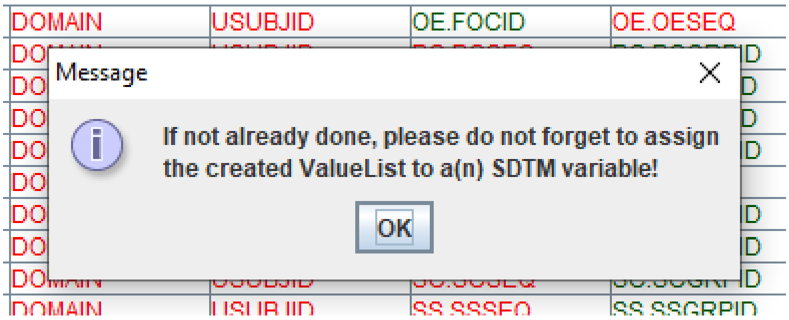
After having made all changes (or additions, removals, …) clicking "OK" leads to:



Message

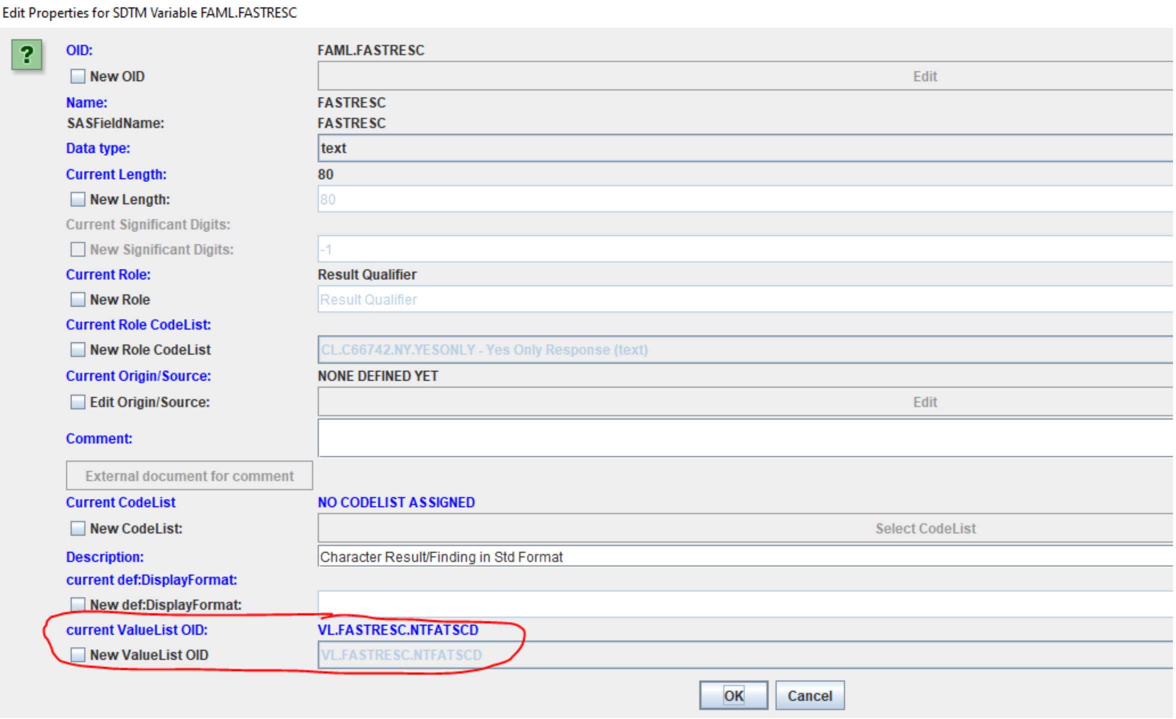
ValueList with OID VL.FASTRESC.NTFATSCD has been updated

followed by:



If not already done, please do not forget to assign the created ValueList to a(n) SDTM variable!

One one than uses the menu "Edit - SDTM Variable Properties" (Ctrl-E), and look for "Current ValueList OID", one sees that it has already been assigned.


Edit Properties for SDTM Variable FAML.FASTRESC

OID:	FAML.FASTRESC
□ New OID	Edit
Name:	FASTRESC
SASFieldName:	FASTRESC
Data type:	text
Current Length:	80
□ New Length:	80
Current Significant Digits:	
□ New Significant Digits:	-1
Current Role:	Result Qualifier
□ New Role	Result Qualifier
Current Role CodeList:	
□ New Role CodeList	CL.C66742.NY.YESONLY - Yes Only Response (text)
Current Origin/Source:	NONE DEFINED YET
□ Edit Origin/Source:	Edit
Comment:	
External document for comment	
Current CodeList	NO CODELIST ASSIGNED
□ New CodeList:	Select CodeList
Description:	Character Result/Finding in Std Format
current def:DisplayFormat:	
□ New def:DisplayFormat:	
current ValueList OID:	VL.FASTRESC.NTFATSCD
□ New ValueList OID	VL.FASTRESC.NTFATSCD

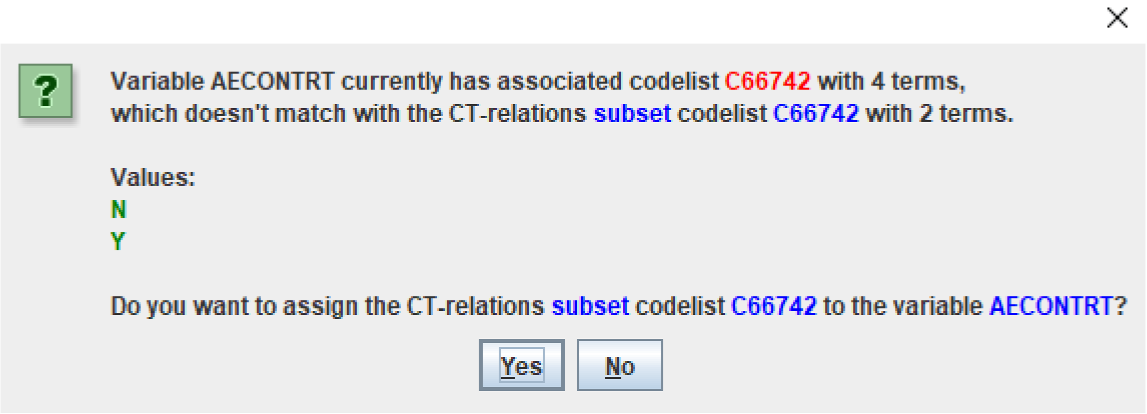
Of course it is always a good idea to check.

A movie about this feature can also be found on YouTube.

More "CT-Relationships"

In the last versions of the SDTM-ETL 4.x series, we started implementing the by CDISC published "Controlled Terminology Relationships" (CT-Relationships), especially for the creation of subset-codelists and valuelists. These "CT-Relationships" are especially for SDTMIG versions 3.3 and 3.4. They e.g. do not apply to SDTMIG-3.2 or SEND.

For example, when starting a mapping for the variable AECONTRT (Concomitant or Additional Trtmnt Given), the system will recognize that there is "CT-Relationship" for it, and display a message:

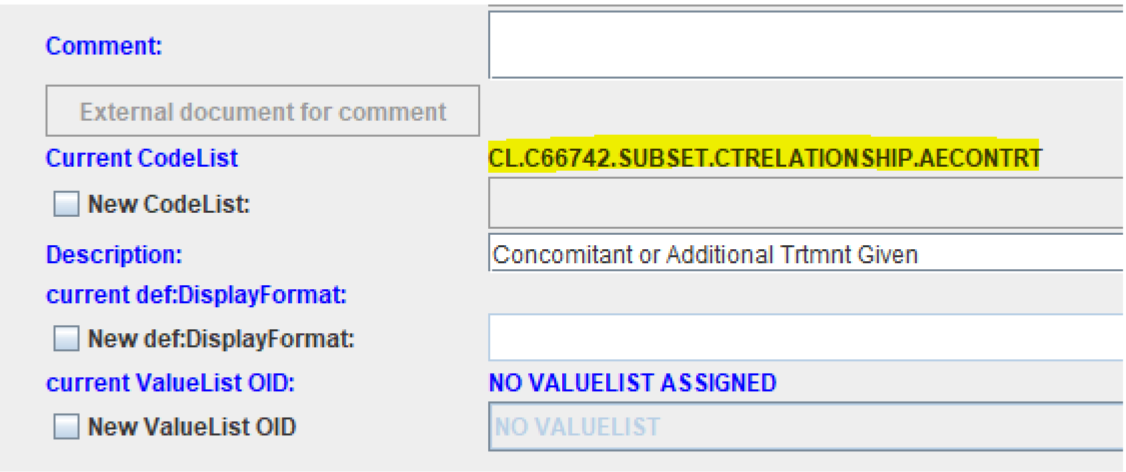


Variable AECONTRT currently has associated codelist C66742 with 4 terms,
which doesn't match with the CT-relations subset codelist C66742 with 2 terms.

Values:
N
Y

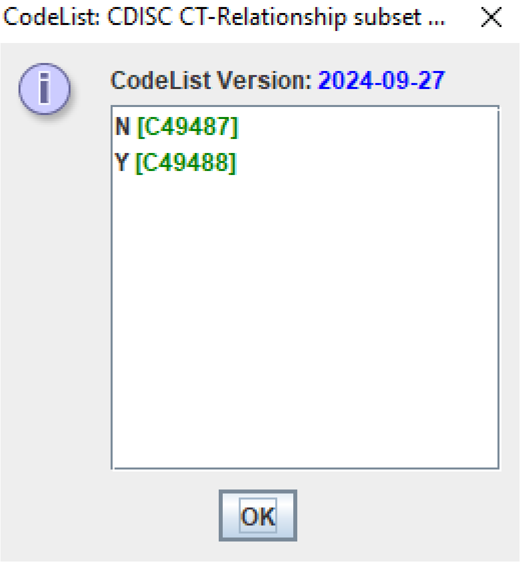
Do you want to assign the CT-relations subset codelist C66742 to the variable AECONTRT?

stating that essentially a subset-codelist of the C66742 codelist applies, with the values "N" and "Y", i.e. that the values "U" (Unknown) and "NA" (Not Applicable) are not allowed to be used. If the user then clicks "Yes" for question whether a subset-codelist should be generated and assigned to AECONTRT, such a codelist is automatically generated.
After having worked on the mapping, this can then be checked by using the menu "Edit - SDTM Variable":



Comment:	
External document for comment	
Current CodeList	CL.C66742.SUBSET.CTRELATIONSHIP.AECONTRT
□ New CodeList:	
Description:	Concomitant or Additional Trtmnt Given
current def:DisplayFormat:	
□ New def:DisplayFormat:	
current ValueList OID:	NO VALUELIST ASSIGNED
□ New ValueList OID	NO VALUELIST

or using the menu "View - SDTM associated CodeList":

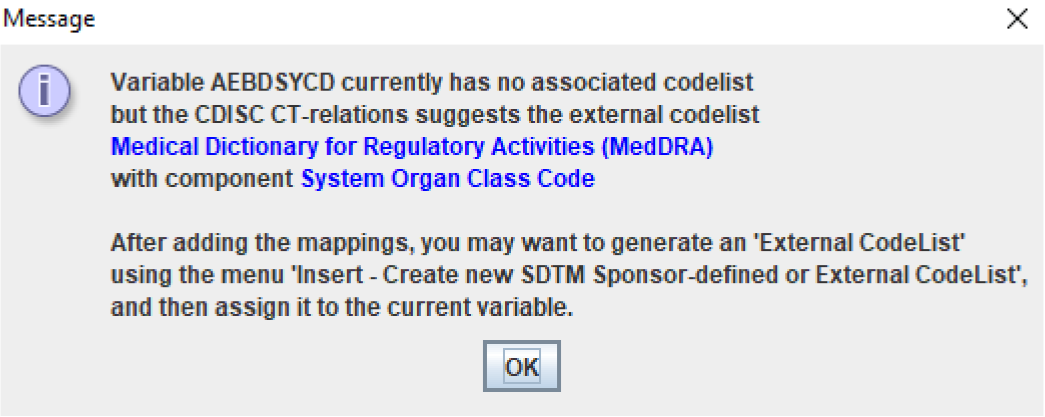


CodeList: CDISC CT-Relationship subset ...

CodeList Version: 2024-09-27

N [C49487]
Y [C49488]

New in v.5.0 is that also the information from the "CT-Relationships" regarding external codelists has been implemented. For example, when starting working on the mapping for AEBDSYCD (Body System or Organ Class Code), the system knows there is "CT-Relationship" information for it, and reacts with:

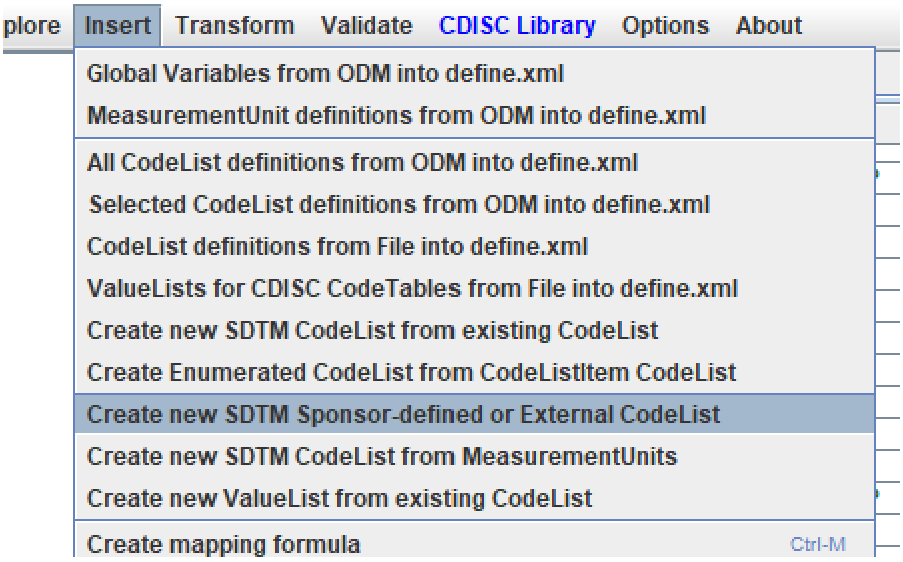


Variable AEBDSYCD currently has no associated codelist
but the CDISC CT-relations suggests the external codelist
Medical Dictionary for Regulatory Activities (MedDRA)
with component System Organ Class Code

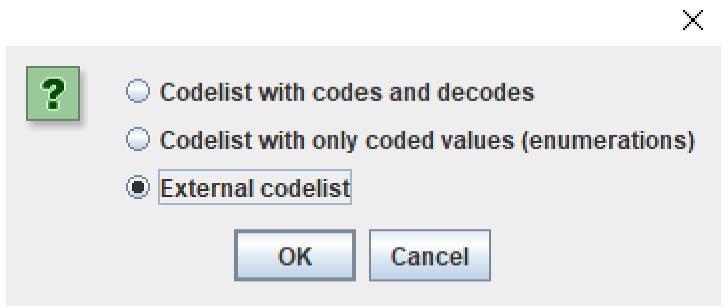
After adding the mappings, you may want to generate an 'External CodeList'
using the menu 'Insert - Create new SDTM Sponsor-defined or External CodeList',
and then assign it to the current variable.

Better mechanism for adding external codelist information

We also added better support for defining and assigning "external" codelists (e.g. MedDRA, WHODrug, LOINC, …), by extending the menu "Insert - Create new SDTM Sponsor-defined CodeList", which has been changed into "Create new SDTM Sponsored-defined or External CodeList":


plore
Insert
Transform
Validate
CDISC Library
Options
About
Global Variables from ODM into define.xml
MeasurementUnit definitions from ODM into define.xml
All CodeList definitions from ODM into define.xml
Selected CodeList definitions from ODM into define.xml
CodeList definitions from File into define.xml
ValueLists for CDISC CodeTables from File into define.xml
Create new SDTM CodeList from existing CodeList
Create Enumerated CodeList from CodeListItem CodeList
Create new SDTM Sponsor-defined or External CodeList
Create new SDTM CodeList from MeasurementUnits
Create new ValueList from existing CodeList
Create mapping formula
Ctrl-M

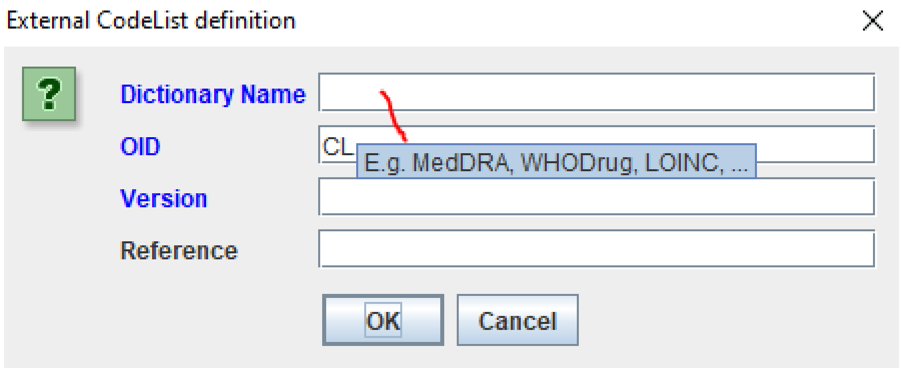
The dialog that then appears now has an additional choice "External CodeList":


○ Codelist with codes and decodes

○ Codelist with only coded values (enumerations)

● External codelist

when selected, this leads to a new dialog:


External CodeList definition

Dictionary Name:

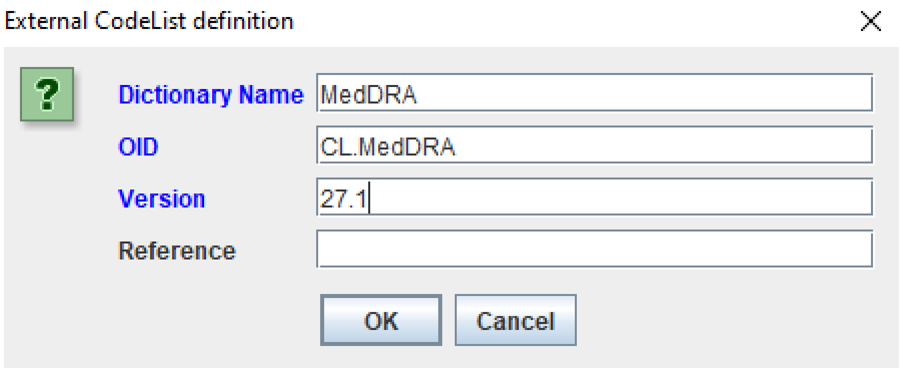
OID: CL

E.g. MedDRA, WHODrug, LOINC, ...

Version:

Reference:

with the fields with a blue label being "mandatory".
When filling the field with "Dictionary Name", the field for OID will automatically be filled (but one can still change it).
The result of filling may e.g. be:


External CodeList definition

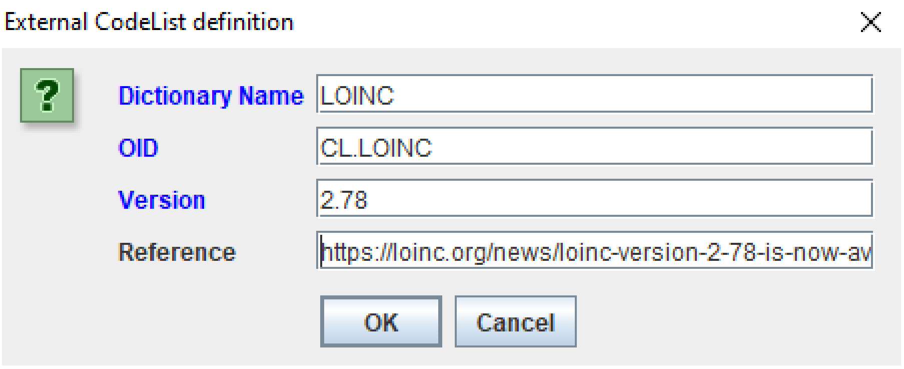
Dictionary Name: MedDRA

OID: CL.MedDRA

Version: 27.1

Reference:

for MedDRA, or for LOINC:


External CodeList definition

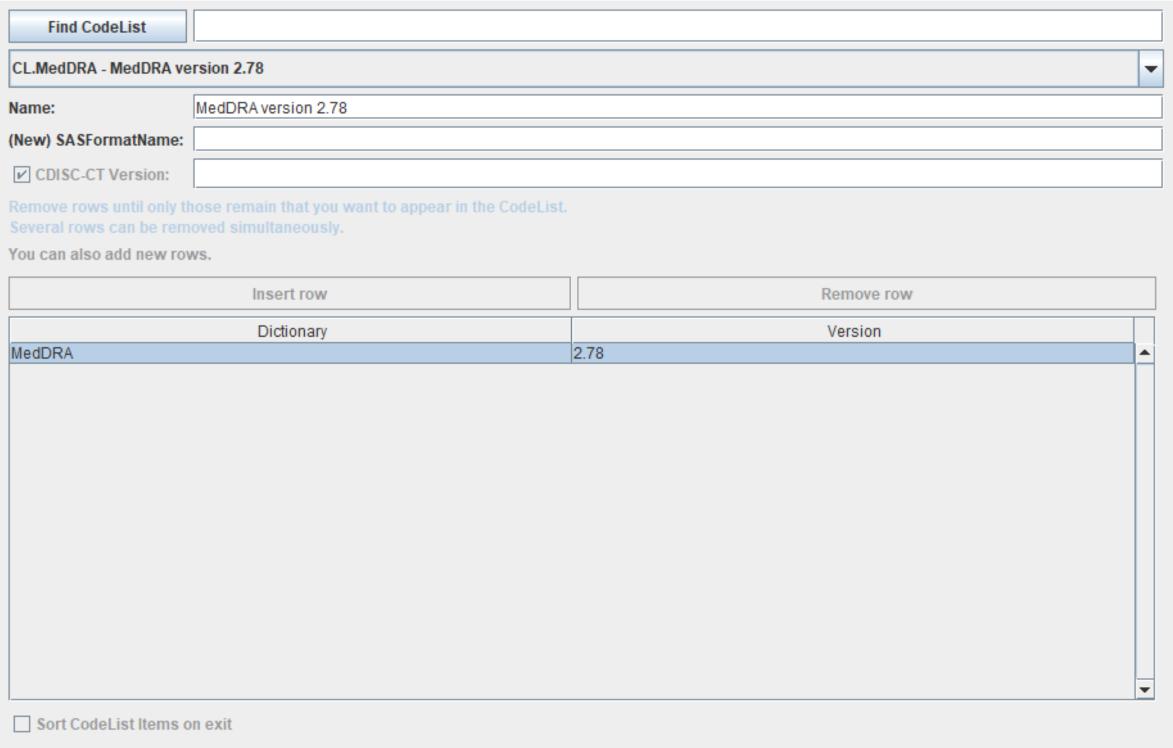
Dictionary Name: LOINC

OID: CL.LOINC

Version: 2.78

Reference: https://loinc.org/news/loinc-version-2-78-is-now-av

After finalizing, one can later still edit the information using the menu "Edit - SDTM CodeList", leading e.g. to:



CL.MedDRA - MedDRA version 2.78

Name: MedDRA version 2.78

(New) SASFormatName:

☑ CDISC-CT Version:

Remove rows until only those remain that you want to appear in the CodeList.
Several rows can be removed simultaneously.

You can also add new rows.

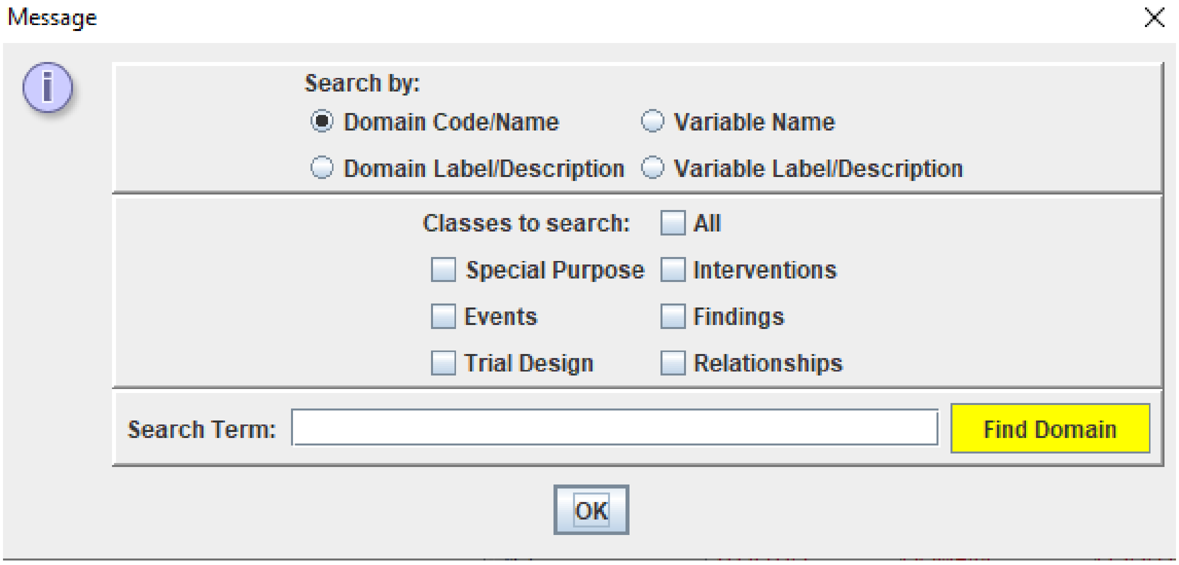
Dictionary	Version
MedDRA	2.78

☐ Sort CodeList Items on exit

If, when finalizing the "external codelist", and a applicable cell (e.g. AEBDSYCD) is selected, the system will also ask whether it should assign the "external codelist" to that variable.

Updates for "Navigate - Find SDTM/SEND Domain" and "Navigate - Find SDTM/SEND Variable"

Some improvements have been made to the dialogs to find a specific domain or variable. For finding an SDTM or SEND domain, the dialog now shows as:



Message

Search by:

● Domain Code/Name ○ Variable Name

○ Domain Label/Description ○ Variable Label/Description

Classes to search: □ All

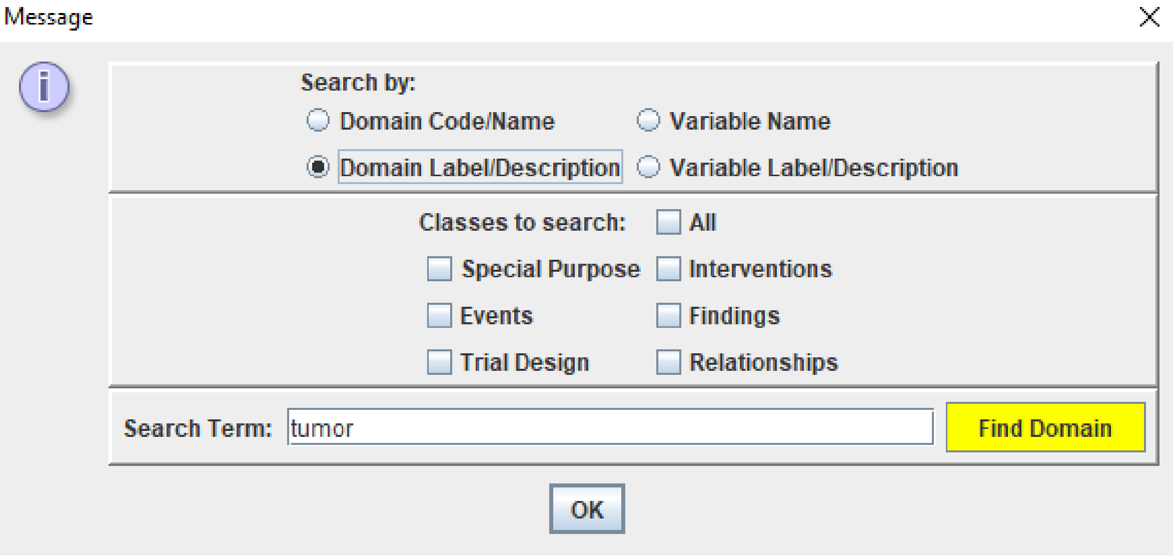
□ Special Purpose □ Interventions

□ Events □ Findings

□ Trial Design □ Relationships

Search Term:

For example, when filling in "tumor" and wanting to look for the word in the "label", we use:



Message

Search by:

○ Domain Code/Name ○ Variable Name

● Domain Label/Description ○ Variable Label/Description

Classes to search: □ All

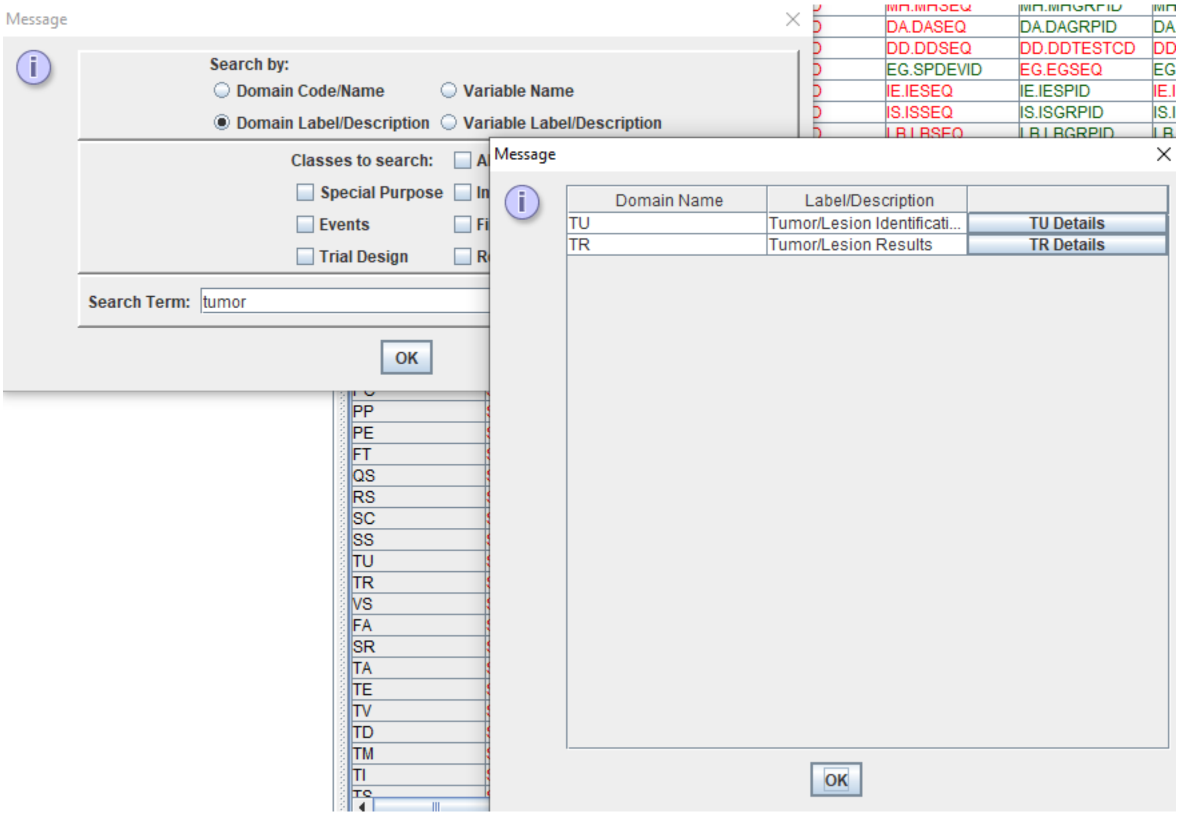
□ Special Purpose □ Interventions

□ Events □ Findings

□ Trial Design □ Relationships

Search Term: tumor

the result after clicking "OK" is:



Message

Search by:
○ Domain Code/Name ○ Variable Name
● Domain Label/Description ○ Variable Label/Description

Classes to search: □ A
□ Special Purpose □ In
□ Events □ Fi
□ Trial Design □ R

Search Term: tumor

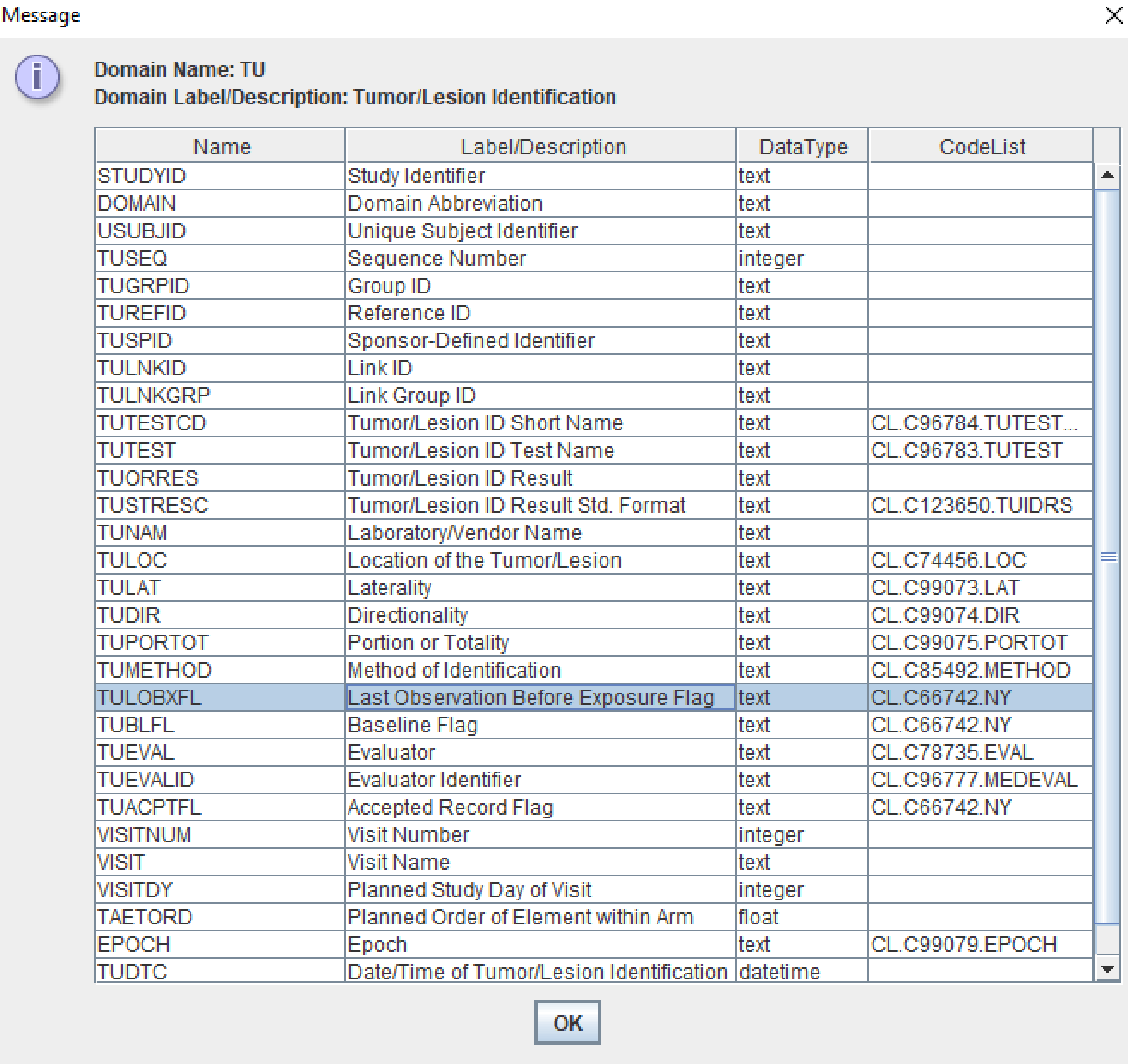
OK

Message

Domain Name	Label/Description	
TU	Tumor/Lesion Identificati...	TU Details
TR	Tumor/Lesion Results	TR Details

OK

One can get more details by clicking one of the "xx Details" buttons, e.g. "TU Details" leading to:

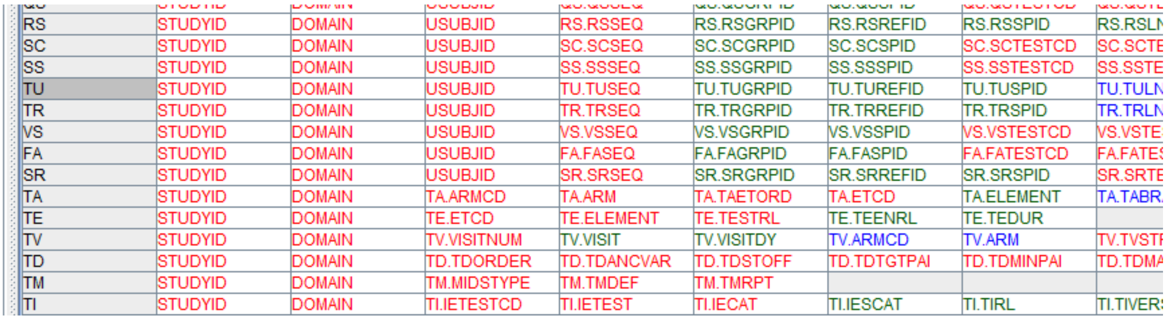


Message

Domain Name: TU
Domain Label/Description: Tumor/Lesion Identification

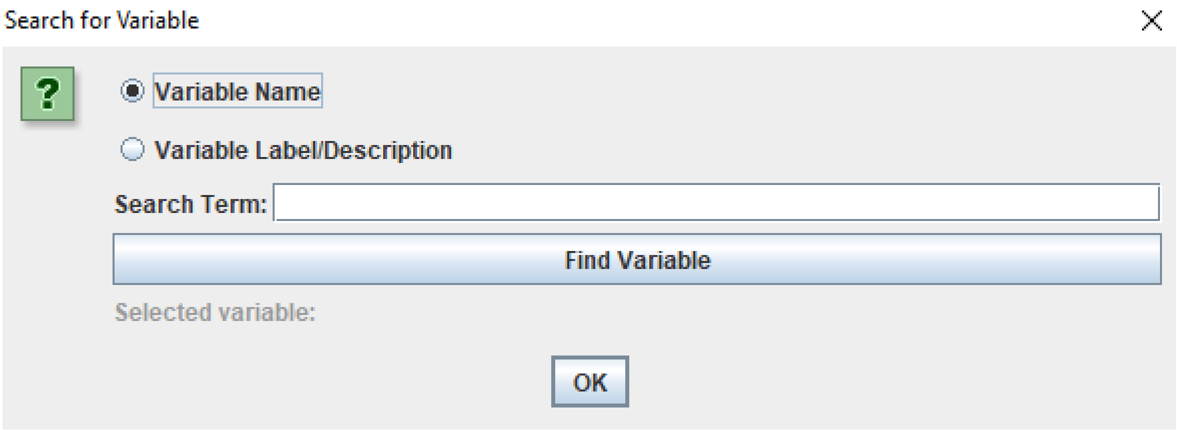
Name	Label/Description	DataType	CodeList
STUDYID	Study Identifier	text	
DOMAIN	Domain Abbreviation	text	
USUBJID	Unique Subject Identifier	text	
TUSEQ	Sequence Number	integer	
TUGRPID	Group ID	text	
TUREFID	Reference ID	text	
TUSPID	Sponsor-Defined Identifier	text	
TULNKID	Link ID	text	
TULNKGRP	Link Group ID	text	
TUTESTCD	Tumor/Lesion ID Short Name	text	CL.C96784.TUTEST...
TUTEST	Tumor/Lesion ID Test Name	text	CL.C96783.TUTEST
TUORRES	Tumor/Lesion ID Result	text	
TUSTRESC	Tumor/Lesion ID Result Std. Format	text	CL.C123650.TUIDRS
TUNAM	Laboratory/Vendor Name	text	
TULOC	Location of the Tumor/Lesion	text	CL.C74456.LOC
TULAT	Laterality	text	CL.C99073.LAT
TUDIR	Directionality	text	CL.C99074.DIR
TUPORTOT	Portion or Totality	text	CL.C99075.PORTOT
TUMETHOD	Method of Identification	text	CL.C85492.METHOD
TULOBXFL	Last Observation Before Exposure Flag	text	CL.C66742.NY
TUBLFL	Baseline Flag	text	CL.C66742.NY
TUEVAL	Evaluator	text	CL.C78735.EVAL
TUEVALID	Evaluator Identifier	text	CL.C96777.MEDEVAL
TUACPTFL	Accepted Record Flag	text	CL.C66742.NY
VISITNUM	Visit Number	integer	
VISIT	Visit Name	text	
VISITDY	Planned Study Day of Visit	integer	
TAETORD	Planned Order of Element within Arm	float	
EPOCH	Epoch	text	CL.C99079.EPOCH
TUDTC	Date/Time of Tumor/Lesion Identification	datetime	

showing all the variables and information about them for that domain. When having selected a domain, e.g. "TU" and then clicking "OK", the row for "TU" is highlighted in the table, so that it can immediately "instantiated" (by drag-and-drop) for use with our study:



RS	STUDYID	DOMAIN	USUBJID	RS.RSSEQ	RS.RSGRPID	RS.RSREFID	RS.RSSPID	RS.RSLN
SC	STUDYID	DOMAIN	USUBJID	SC.SCSEQ	SC.SCGRPID	SC.SCSPID	SC.SCTESTCD	SC.SCTE
SS	STUDYID	DOMAIN	USUBJID	SS.SSSEQ	SS.SSGRPID	SS.SSSPID	SS.SSTESTCD	SS.SSTE
TU	STUDYID	DOMAIN	USUBJID	TU.TUSEQ	TU.TUGRPID	TU.TUREFID	TU.TUSPID	TU.TULN
TR	STUDYID	DOMAIN	USUBJID	TR.TRSEQ	TR.TRGRPID	TR.TRREFID	TR.TRSPID	TR.TRLN
VS	STUDYID	DOMAIN	USUBJID	VS.VSSEQ	VS.VSGRPID	VS.VSSPID	VS.VSTESTCD	VS.VSTE
FA	STUDYID	DOMAIN	USUBJID	FA.FASEQ	FA.FAGRPID	FA.FASPID	FA.FATESTCD	FA.FATES
SR	STUDYID	DOMAIN	USUBJID	SR.SRSEQ	SR.SRGRPID	SR.SRREFID	SR.SRSPID	SR.SRTE
TA	STUDYID	DOMAIN	TA.ARMCD	TA.ARM	TA.TAETORD	TA.ETCD	TA.ELEMENT	TA.TABR
TE	STUDYID	DOMAIN	TE.ETCD	TE.ELEMENT	TE.TESTRL	TE.TEENRL	TE.TEDUR	
TV	STUDYID	DOMAIN	TV.VISITNUM	TV.VISIT	TV.VISITDY	TV.ARMCD	TV.ARM	TV.TVSTI
TD	STUDYID	DOMAIN	TD.TDORDER	TD.TDANCVAR	TD.TDSTOFF	TD.TDTGTPAI	TD.TDMINPAI	TD.TDMA
TM	STUDYID	DOMAIN	TM.MIDSTYPE	TM.TMDEF	TM.TMRPT			
TI	STUDYID	DOMAIN	TI.IETESTCD	TI.IETEST	TI.IECAT	TI.IESCAT	TI.TIRL	TI.TIVERS

Similarly for the case we are looking for a specific variable, the dialog displayed is:



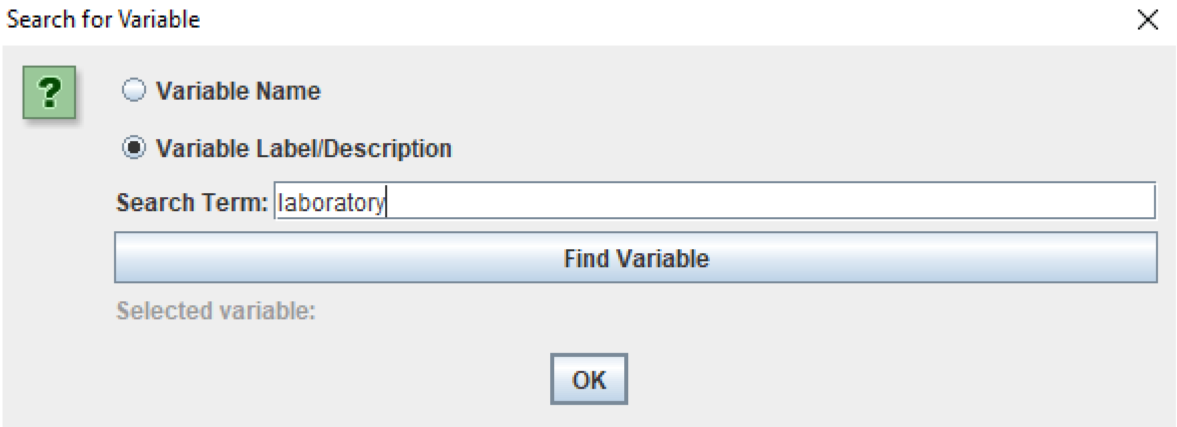
● Variable Name

○ Variable Label/Description

Search Term:

Selected variable:

and we e.g. look for the word "laboratory" in any of the labels:



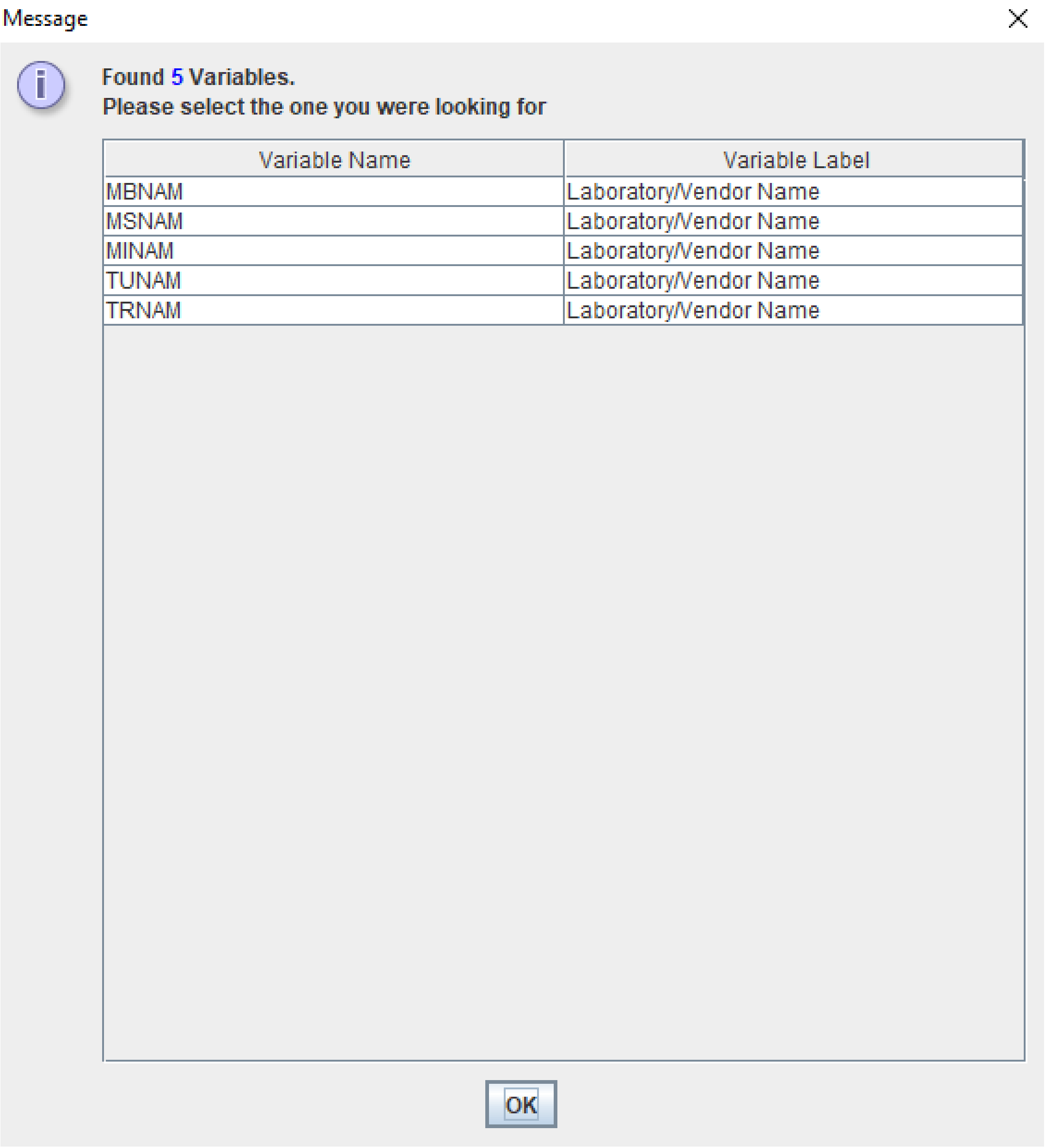
○ Variable Name

● Variable Label/Description

Search Term: laboratory

Selected variable:

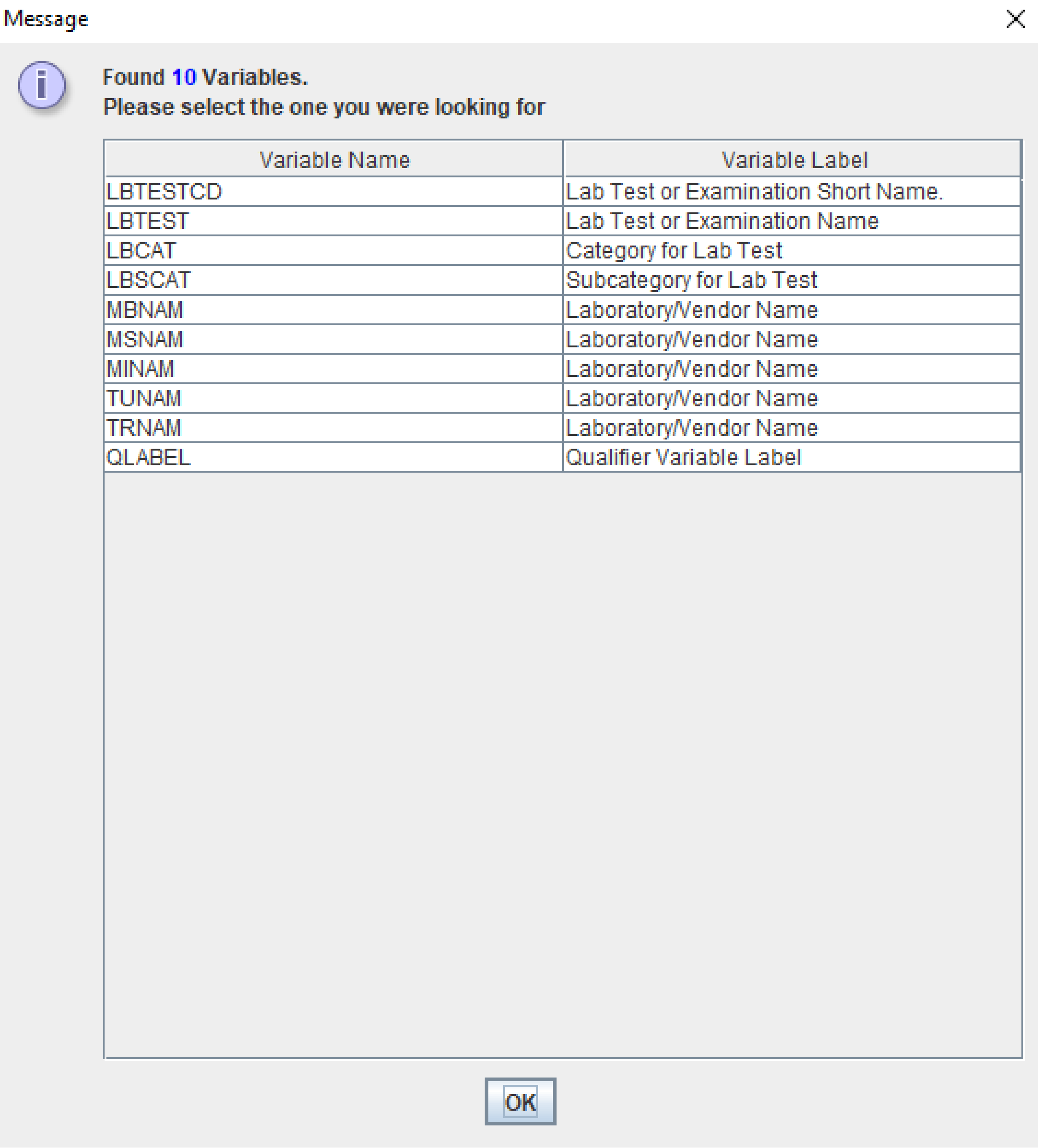
The result after clicking "Find Variable" is:



Found 5 Variables.
Please select the one you were looking for

Variable Name	Variable Label
MBNAM	Laboratory/Vendor Name
MSNAM	Laboratory/Vendor Name
MINAM	Laboratory/Vendor Name
TUNAM	Laboratory/Vendor Name
TRNAM	Laboratory/Vendor Name

When we search for "Lab" however, the result is:

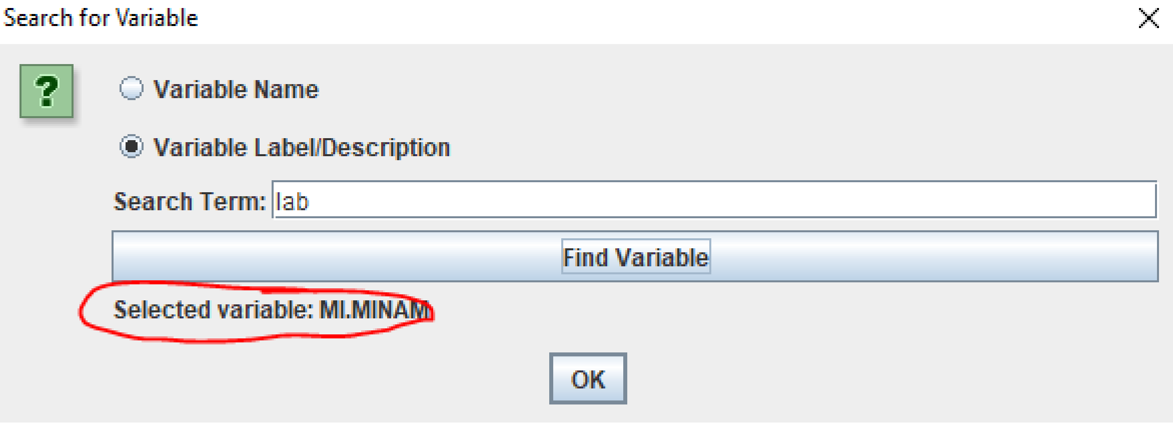


Message

Found 10 Variables.
Please select the one you were looking for

Variable Name	Variable Label
LBTESTCD	Lab Test or Examination Short Name.
LBTEST	Lab Test or Examination Name
LBCAT	Category for Lab Test
LBSCAT	Subcategory for Lab Test
MBNAM	Laboratory/Vendor Name
MSNAM	Laboratory/Vendor Name
MINAM	Laboratory/Vendor Name
TUNAM	Laboratory/Vendor Name
TRNAM	Laboratory/Vendor Name
QLABEL	Qualifier Variable Label

and when we click e.g. on "MINAM" and then "OK", we get an update in the prior dialog:



Search for Variable

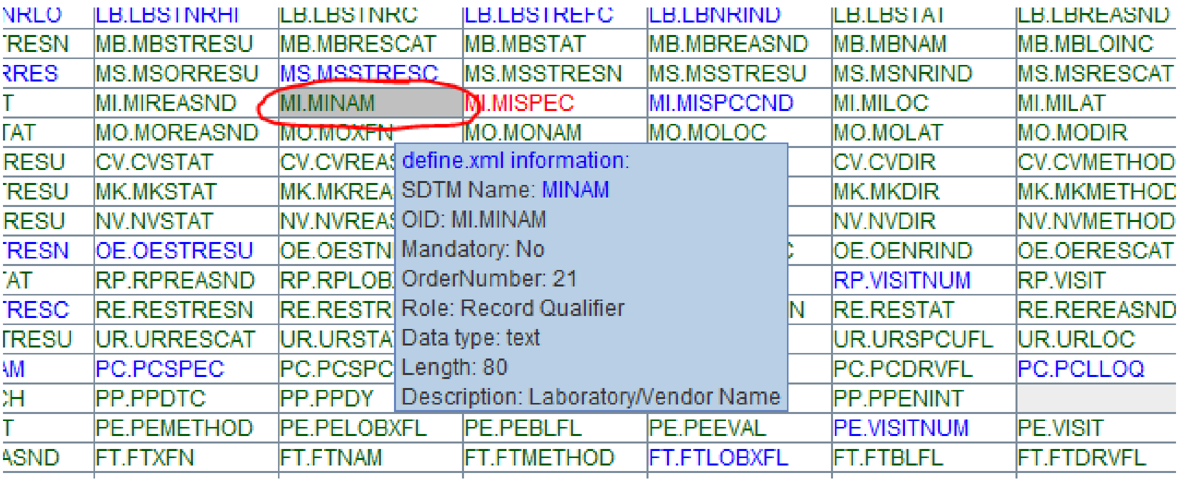
○ Variable Name

● Variable Label/Description

Search Term: lab

Selected variable: MI.MINAM

and when then clicking "OK", the variable is selected and highlighted in the template table:



NRLO	LB.LBSTNRHI	LB.LBSTNRC	LB.LBSTREFC	LB.LBNRIND	LB.LBSTAT	LB.LBREASND
RESN	MB.MBSTRESU	MB.MBRESCAT	MB.MBSTAT	MB.MBREASND	MB.MBNAM	MB.MBLOINC
RRES	MS.MSORRESU	MS.MSSTRESC	MS.MSSTRESN	MS.MSSTRESU	MS.MSNRIND	MS.MSRESCAT
T	MI.MIREASND	MI.MINAM	MI.MISPEC	MI.MISPCCND	MI.MILOC	MI.MILAT
TAT	MO.MOREASND	MO.MOXFN	MO.MONAM	MO.MOLOC	MO.MOLAT	MO.MODIR
RESU	CV.CVSTAT	CV.CVREAS			CV.CVDIR	CV.CVMETHOD
RESU	MK.MKSTAT	MK.MKREA			MK.MKDIR	MK.MKMETHOD
RESU	NV.NVSTAT	NV.NVREAS			NV.NVDIR	NV.NVMETHOD
RESN	OE.OESTRESU	OE.OESTN		C	OE.OENRIND	OE.OERESCAT
AT	RP.RPREASND	RP.RPLOB			RP.VISITNUM	RP.VISIT
RESC	RE.RESTRESN	RE.RESTR		N	RE.RESTAT	RE.REREASND
TRESU	UR.URRESCAT	UR.URSTA			UR.URSPCUFL	UR.URLOC
AM	PC.PCSPEC	PC.PCSPC			PC.PCDRVFL	PC.PCLLOQ
CH	PP.PPDTC	PP.PPDY			PP.PPENINT	
T	PE.PEMETHOD	PE.PELOBXFL	PE.PEBLFL	PE.PEEVAL	PE.VISITNUM	PE.VISIT
ASND	FT.FTXFN	FT.FTNAM	FT.FTMETHOD	FT.FTLOBXFL	FT.FTBLFL	FT.FTDRVFL

define.xml information:
SDTM Name: MINAM
OID: MI.MINAM
Mandatory: No
OrderNumber: 21
Role: Record Qualifier
Data type: text
Length: 80
Description: Laboratory/Vendor Name

Extended CLI (batch) execution

SDTM-ETL can also be used to execute developed mappings in non-GUI, i.e. "batch" mode, using a "Command Line Interface". We have now extended the parameters used for the CLI, especially due to new features that have been introduced in versions 4.4 and 4.5, such as automated generation of VISITNUM for "unscheduled" visits.

Other minor updates

When saving a "working define.xml" to file, the "Length" attributes for data type "date", "datetime" and similar are not removed anymore. Reason is that when later contnueing the work, and then generate SAS-XPT, the "length" is still need for allocating the number of bytes in the XPT in order to "optimize" the XPT file sizes. When however using "Save cleaned define.xml", the "Length" attribute is being removed for those variables for which the Define-XML standard does not allow the "Length" attribute, as the aim of this feature is to generate a "near-submission-ready" define.xml, and not a define.xml for developing the mappings.

Bug fixes

The tooltip on the button "round" (for the "round" function) in the "Mapping Script Editor" was incorrect. It said that this function was for rounding to the nearest integer. This it incomplete: the function also round to the provided number of characters after the decimal point (second argument of the function). For example:

$pi = 3.1415927
$a = round($pi, 0); => results in "3"
$b = round($pi,4);   => results in "3.1416".
The tooltip has been updated to reflect this.

Under certain circumstances, the "Dataset Label" was not written to the "header" in SAS-XPT datasets. This has been corrected.

In the case of "split domain" dataset definitions, containing a "relrec" variable for automated generation of RELREC records, the assignment of RDOMAIN and IDVAR could be wrong under some circumstances. This has now been fixed.

Limitations

As CDISC is currently updating the Dataset-JSON standard (to version 1.1) not all newer features have been implemented for generation of SDTM or SEND datasets in Dataset-JSON format. We recommend users to not use the Dataset-JSON export for production at the moment, also as it is still using Dataset-JSON v.1.0.
Once Dataset-JSON 1.1 is final and published, we will again adapt the software for the new standard, which we expect also to become the submission format of choice of the FDA in near future.

Further development of SDTM-ETL

It is now very clear that the FDA is committed to push the replacement of the SAS Transport 5 (XPT) format by the by CDISC developed Dataset-JSON format: the pilot with the FDA was extremely successful, and a CDISC working group (of which we are part) is currently refining the new standard, which will be name Dataset-JSON 1.1.

It is also very clear that FDA fully supports the CDISC CORE project for validation of submission datasets. For example, FDA has asked CDISC to add all "FDA business rules" to CORE. Essentially, this means that for validation, Pinnacle21 is expected to be replaced by CDISC CORE, also at the agency.

Both these have implications for the new versions of the SDTM-ETL software: SDTM-ETL 5.0 has Dataset-JSON 1.1 as the primary format for the generated datasets and SAS Transport 5 as the secondary. Support for the old Dataset-XML format is terminated.
As one of the co-developers of CDISC-CORE we will further support and extend the use of this open-source validation software, so that the world can finally get rid of this buggy and user-unfriendly P21 software.
We are also thinking about features such as "validate as you map", providing immediate feedback for the CORE validation engine for each case that a mapping was added or changed.
Also, as soon as it is available, we will add and support the feature that users (or companies) can add their own validation rules to the engine in a secure way[5].

[5] also meaning that any user- or company-specific validation rules are not shared with CDISC nor anyone else outside the company.