

ABSTRACT

INTRODUCTION

We here present a GUI-based (non-statistical) software named "SDTM-ETL" that is extremely "CDISC-savvy", and mostly uses "drag-and-drop" in combination "wizards" to generate and execute mapping code to SDTM or SEND, together with a synchronized define.xml.

The SDTM-ETL software comes as zero-install package together with all the define.xml templates for all recent SDTM and SEND standards versions, and all CDISC Controlled Terminology of the last 10 years. Templates and files for new standard versions and new controlled terminology can simply be added without the need of a software update. Also the "CDISC Code Tables" (as an implementation of the "CDISC Biomedical Concepts") come with the distribution and new ones can easily be added.

BASICS OF SDTM-ETL

[illegible]

In case the ODM metadata is CDASH- or SDTM-annotated, and an item is selected, then the most suitable candidate on the SDTM side is highlighted.

The process of generating a mapping for a specific domain is started by drag-and-drop the row for that domain to the bottom, creating a "study-specific instance" of that domain.

ODM Item **SDTM CodeList Item**

☒ Show ODM decoded values

I_LB_RBC: Erythrocytes	RBC	Search
I_LB_WBC: Leukocytes	WBC	Search
MISSED VALUE		Search

☒ Generate subset code list from selected SDTM items, and assign to the SDTM variable **LB.LBTESTCD**

☒ Also create a subset code list for the corresponding **LB.LBTEST** (test name) variable, and generate the corresponding mapping script for the corresponding **LB.LBTEST** variable

☐ Adapt variable Length for longest CodeList item

☐ Add comment line to each mapping

☐ Except for items already mapped

☒ Also use CDISC Synonym List

☒ Also use Company Synonym List

Attempt 1:1 mapping **Reset from 1:1 mapping attempt**

The mapping scripts can be further fine-tuned, but this is usually only necessary in 10-20% of the cases, as most is already taken care of by the wizards. For working on the mappings, a rich set of functions is available, including many date and time manipulation functions. Some of these functions use RESTful web services, like a function that automatically generates values for LBTESTCD, LBTEST, LBSPEC (specimen), LBMETHOD, LBRESSCL (result scale), etc. starting from the LOINC code. Another one (from the National Library of Medicine) takes care of unit conversion, e.g. between US conventional and SI units (and the other was around). These conversions can also be cached.

The Transformation Script

```

1 # Mapping using ODM element ItemData with ItemOID I_LB_WBC - value from attribute ItemOID
2 # Generalized for all StudyEvents
3 # Generalized for all Items within the ItemGroup
4 # Mapping automatically generated for ODM Items [I_LB_RBC, I_LB_WBC] to SDTM CodeList LB.LBTESTCD
5 # with CodeList OID 'CL.C65047.LBTESTCD'
6 %CODEVAL = %path(/StudyEventData/FormData[@FormID='F_LAB']/ItemGroupData[@ItemGroupID='IG_LB_HEMATOLOGY
7 if (%CODEVAL = 'I_LB_RBC') {
8   %NEWCODEVAL = 'RBC';
9 }
10 %NEWCODEVAL = 'WBC';
11 %NEWCODEVAL = 'MISSED VALUE';

```

Scripting Language Functions

More date and time functions	matches	not	!=
Previous	log10	exp	exp10
Next	sum	count	is-a-number
earliestdateafter	modulus	number	string
latestdate	day-in-year	day-in-month	day-in-week
earliestdatetime	earliesttimeafter	te	createdatetime
latestdatetime	Value of the earliest datetime in a series of datetimes	dateadd	datediff
earliesttimeafter	Takes a single argument that represents an array of datetimes	more datetimes	RESTful WS
latesttime	more datetimes	RESTful WS	My Functions
dateadd			
datediff			
timeadd			

Execute Transformation (XSLT) Code

ODM file with clinical data:

D:\SDTM-ETL\ETLFiles\ODM1-3-1CES_ClinicalData_multiple_subjects_with_unscheduled_visit_before_first_visit.xml **Browse...**

☐ MetaData in separate ODM file

D:\SDTM-ETL\ETLFiles\ODM1-3-1CES_MetaData_with_unscheduled_visit.xml **Browse...**

☐ Administrative data in separate ODM file

D:\SDTM-ETL\ETLFiles\ODM1-3-1CES_MetaData_with_unscheduled_visit.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 Relrec Variables to Related Records (RELREC) domain

☐ Try to generate 1:N RELREC Relationships

☒ Adapt Variable Length for longest result value

☐ Re-sort records using define.xml keys

☒ View Result SDTM tables

☐ Generate "NOT DONE" records for QS datasets

☒ Save Result SDTM tables as:

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

☒ Perform CDISC CORE validation on generated SDTM files

SDTM export files directory:

D:\temp **Browse...**

☐ Add location of generated SDTM files to define.xml

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

☐ Store link as relative path

Messages and error messages:

Execute Transformation on Clinical Data

Close

Automated codelist mapping can be done based on word similarity and/or using CDISC- or company-defined synonyms. This allows to gain a lot of time as this process is otherwise very tedious.

The same applies to mapping to CDISC-CT based on the distinct values found in the collected data (when not enumerated).

The result of this process then is a mapping script, which can still be edited when desired.

Execution for all mapped variables is then performed on a file with ODM clinical data.

For the execution of the mappings, the user has the choice between four different output formats: SAS-XPT, the new CDISC Dataset-JSON format, Comma-separated-values, and SQL "Insert" statements: When classic XPT format is selected, generation of SUPPQUAL datasets for variable values beyond the XPT 200-character limit is automatically selected. Also, moving "non-standard variables" to SUPPxx is then selected.

Optionally, one can also allow for post-processing steps to automatically assign --LOBXFL (Last Observation Before First Exposure) and to automatically assign fractional VISITNUM for unscheduled visits. The later of course requires that the data is chronologically organized, which can be achieved when necessary, by using the define.xml keys.

All this also applies to the new CDISC Dataset-JSON format, except that there is no need to "ban" values beyond the XPT 200-character limit to a SUPPQUAL dataset.

One can also choose to generate SQL "INSERT" statements to populate an SDTM/SEND database, where one has the choice between "Standard SQL" and "Oracle SQL".

Furthermore, there is the option to start CORE (CDISC Open Rules Engine) validation immediately after the datasets have been generated. This is also important already during the development of the mappings.

DEFINE-XML GENERATION

In SDTM-ETL, the "engine under the hood" is define.xml. Starting from the template for the selected standard and version, the underlying define.xml is updated each time the user does something (like setting a maximum length for a variable, using a CDISC-BC for the mapping, leading to a "ValueList", etc.). Also the mapping scripts themselves are stored within the define.xml itself. This also allows for reuse of the mappings for different, similar studies.

The user can then generate a "cleaned" define.xml file, which is "almost-submission-ready", and can be further enhanced by any modern Define-XML software, such as the "Define-XML Designer".

OTHER FEATURES

The software comes with a build-in graphical "Trial Design dataset editor", allowing to set up and generate trial design datasets in one of the four formats mentioned above. For TS (trial summary), it has a wizard that has all CDISC-CT as well as "FDA-desired" TS parameters together with the allowed values for each of these.

See e.g. <https://www.youtube.com/watch?v=skIKQxRap-Q> for a demo movie.

CONCLUSION (HEADER 1)

The SDTM-ETL software allows the mapping to and generation of CDISC SDTM and SEND submission datasets in an extremely user-friendly way, due to the use of a large number of "wizards". No knowledge of SAS or R is required, so that the user can fully concentrate on SDTM or SEND. The software allows automation for many types of tasks. As it uses Define-XML "under the hood", a define.xml fully aligned with the datasets can be generated at all times during, and after the development of the datasets

DOCUMENTATION AND FURTHER INFORMATION

Further information is available at the website www.xml4pharma.com/SDTM-ETL, including a large number of tutorials.

CONTACT INFORMATION

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

Jozef Aerts

XML4Pharma

Email: Jozef.Aerts@XML4Pharma.com

Website: www.xml4pharma.com

Brand and product names are trademarks of their respective companies.