SDTM-ETL™

The user-friendly ODM – SDTM Mapping software package

Transforming CDISC ODM datasets with clinical data into SDTM datasets is not an easy process.

Therefore, XML4Pharma has developed a visual, easy-to-use mapping software package for defining and executing ODM to SDTM mappings, and to create and fill an SDTM database[1].

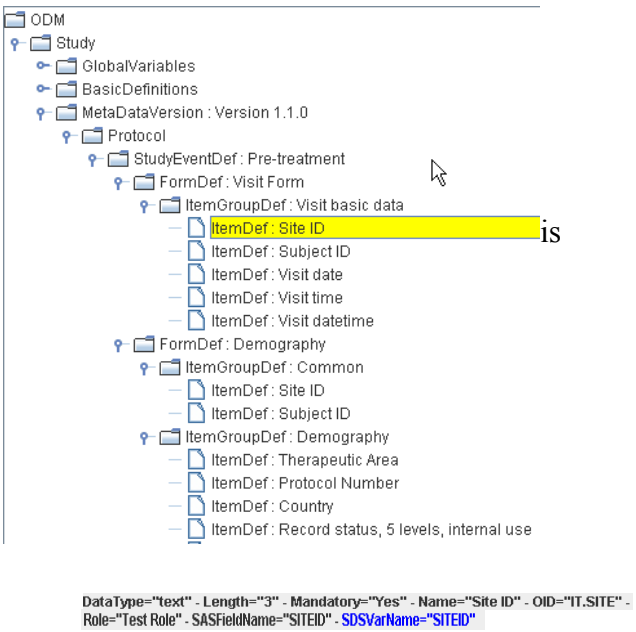
The SDTM-ETL™ software package is a real ETL software tool allowing to:
Extract: ODM metadata and clinical data from ODM files
Transform: describe (even complicated) mappings between ODM and SDTM and execute them.
Load: create an SDTM database and load the SDTM data into the database.

Extract ODM data

The mapping process typically starts with loading ODM metadata (study setup) and reference data. These are validated against the XML-Schema and against the ODM specification. The metadata and reference data are transformed into a deep tree, which visualized in the left part of the graphical interface.

A good number of functions is then available to navigate through the tree, and to find similar items in other groups, forms and visits.

If requested, also data files with clinical data can be validated against the standard and the currently loaded metadata.

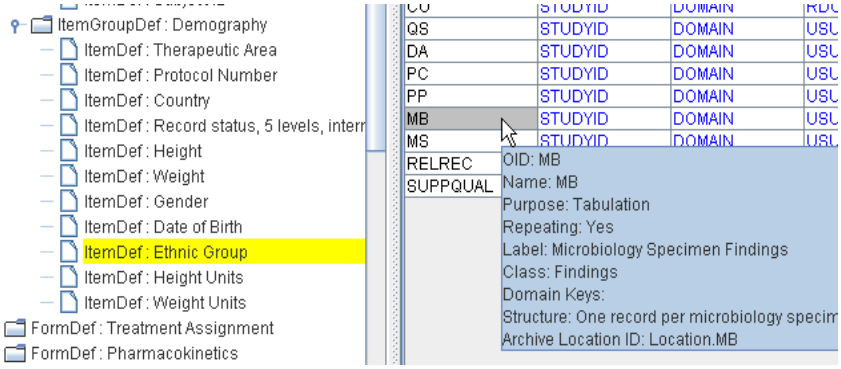


is

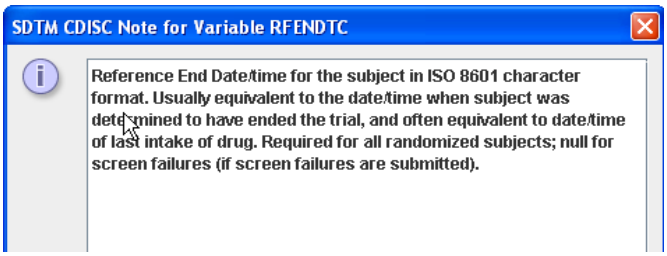
1 The SDTM-ETL software is the further development of the SDTMWandler, a project initiated by TMF e.V., the German Platform for Telematics in Medical Research.

Load SDTM tables

The whole SDTM standard table (SDTM v.1.1 or v.1.2) is loaded from a template define.xml file, which is shown in the right part of the graphical interface. The use of a define.xml file has the advantage that the template can easily be extended when CDISC publishes new SDTM domains, or when sponsors have own standard domains.



This SDTM template is the source for the development of study-specific domains. These are created by a simple drag-and-drop procedure, during which mappings for a number of basic SDTM variables (USUBJID, DOMAIN, --SEQ) can already be generated automatically.

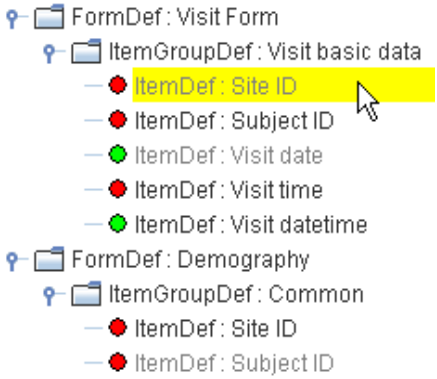


"CDISC Notes" have been implemented for each SDTM variable, so that it is not necessary to search through the CDISC Implementation Guide in order to find out what the meaning of an SDTM variable is and what its use is.

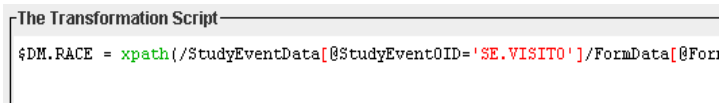
Several options for viewing the ODM data as well as the SDTM tables are provided. For example, the SDTM table can be viewed as a set of HTML tables, with easy navigation (hyperlinks provided).

Create mappings

When the user selects an SDTM variable, the suitability for mapping of each of the items in the ODM tree is shown using "traffic lights", and "hot candidates" are highlighted[2]. When a tree ODM item is selected (this can also be a StudyEvent, Form or ItemGroup), it can simply be dragged into the SDTM cell for starting the mapping.
A set of wizards then guides the user through the "import" process, allowing to make the right decisions about what exactly should be used in the mapping.



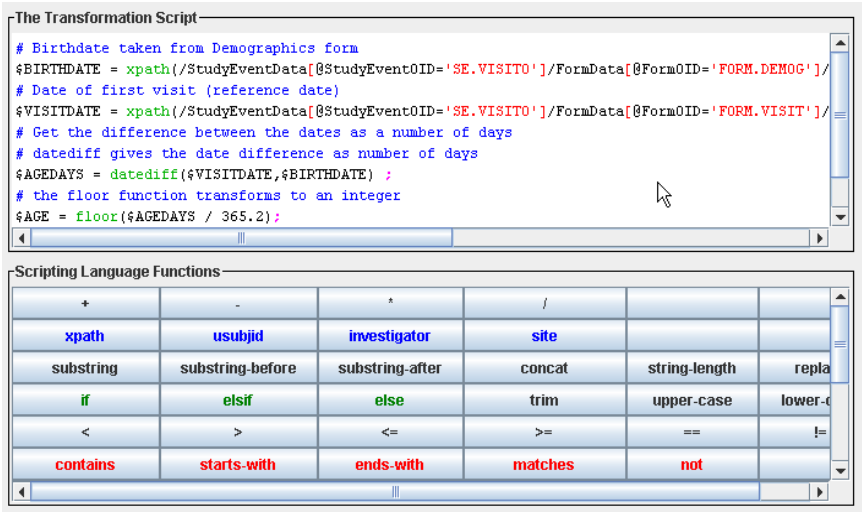
The wizard then automatically creates the mapping script.



If a 1:1 mapping is needed, this is usually the only action that needs to be taken.

2 Based on the value of the ODM SDSVarName attribute

In many cases however, several ODM data needs to be combined in order to create a mapping to an SDTM variable. In those cases, the building blocks can simply be dragged into the SDTM cell and

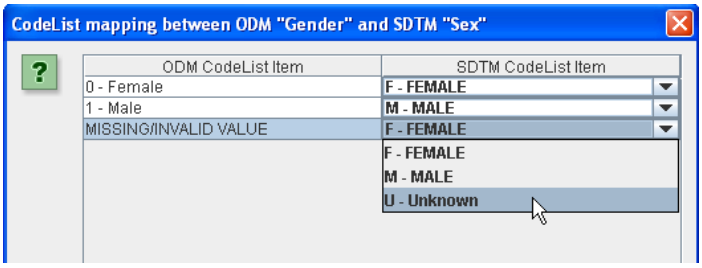


then combined using a simple, extremely easy-to-learn scripting language.
For creating the combinations, a large number of functions is available, for which the script snippets are generated by a simple mouse click.
Mathematical functions are available, as well as string manipulation functions, as well as date, time, and datetime manipulation, extraction and creation functions.
Advanced users can also create their own scripting functions.

Once a script generated, it can be tested on real clinical data.

During drag-and-drop, the system checks whether codelist are involved either at the ODM side or at the SDTM site, and generates wizards for setting up the necessary codelist mappings.

The wizard then automatically creates the mapping script for transformation between the codelists involved, or to categorize the data according to the SDTM codelist[3].



Also import of the ODM codelists and measurement unit definitions into the SDTM structure is possible using a menu function.

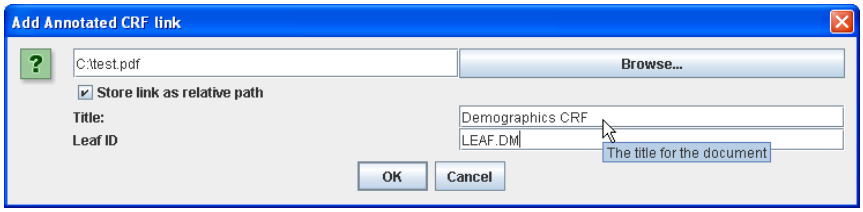
Many of the SDTM domains have a structure, e.g. “One record per subject per visit per time point per measurement”. These structures are already implementated in the define.xml template, but can easily be adapted for each study-specific domain. The mappings can be constructed so, that “generalizations” can be defined, e.g. for the QS domain, where a record needs to be generated for each question (ItemDef) that was ever asked to a patient using a validated standard form. When later executing the mapping, the software automatically creates the necessary iterations implementing the required SDTM domain structure.

All mapping and other information is always stored in a define.xml structure that can easily be exported to a define.xml file. As such, at the end of the mapping process, the resulting define.xml is ready for use for submission to the FDA[4].

3 The latest CDISC codelists from the CDISC Controlled Terminology team are included in the template define.xml file

4 The define.xml structure can also be validated against the standard, and for internal consistency

Remark that also the locations for annotated CRFs (def:AnnotatedCRF) and supplemental documents (def:SupplementalDoc) can be added in a visual way.

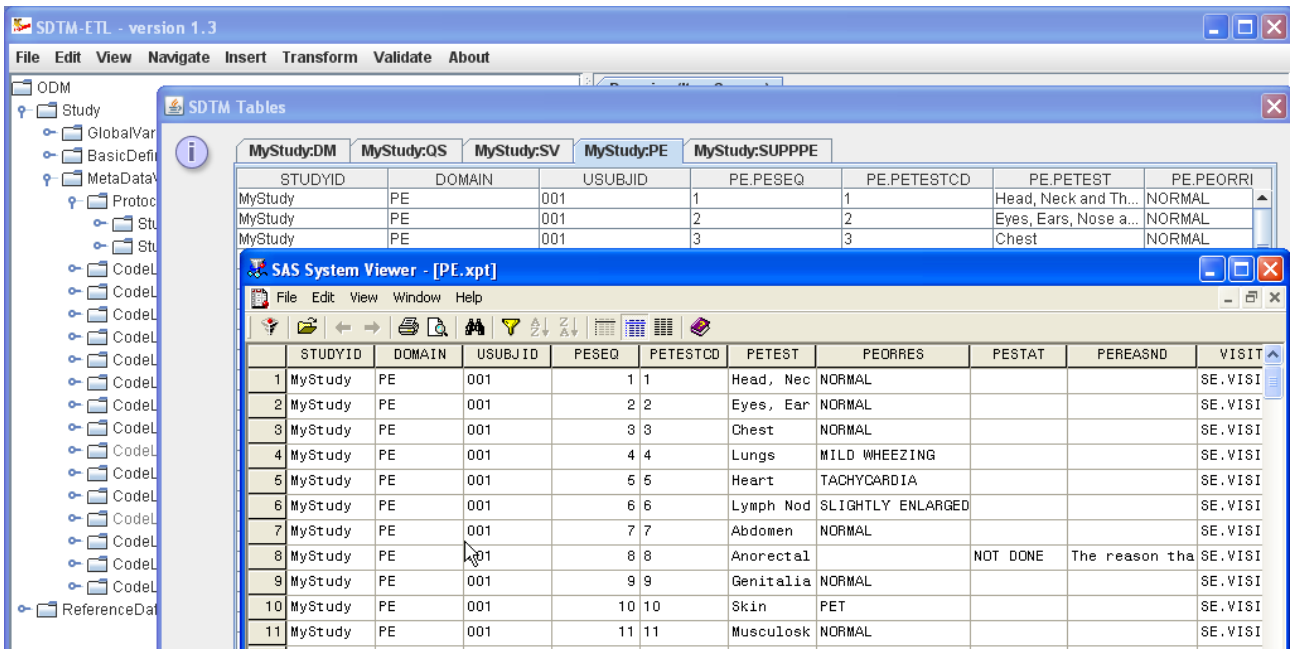


Create SDTM datasets

Once a (partial) mapping has been developed, it can be executed on ODM clinical data files (i.e. transformation of ODM data into SDTM records). The developed mapping scripts are transformed to a set of XSLT scripts (which can be saved to file) and executed on the selected ODM file with clinical data. The output are SDTM records in SAS Transport (XPT) and in XML format, the latter being a precursor of the upcoming SDTM-XML standard[5]. Also a tabular view of the SDTM records is generated. The generated SDTM records can then again be validated against the generated define.xml.
SUPP-- datasets and records are automatically created whenever necessary.
All this is done in such a way that the user does not need to know anything about XSLT, so essentially, XML knowledge is not required at all.

The generated SDTM records in XML format

5 As soon as the SDTM-XML or HL7 standard is available, it will be implementated in the software.



The generated SDTM Records in SAS Transport format

Create an SDTM database and load SDTM records into the SDTM database

An SDTM database can be easily created as the software allows to generate all necessary SQL[6] "Create Table" statements from the define.xml structures. A number of options for renaming table names and variable names is provided through another wizard.

Once the SDTM records have been generated, they can be loaded into the SDTM database, as the software allows to create all necessary SQL "Insert" statements.

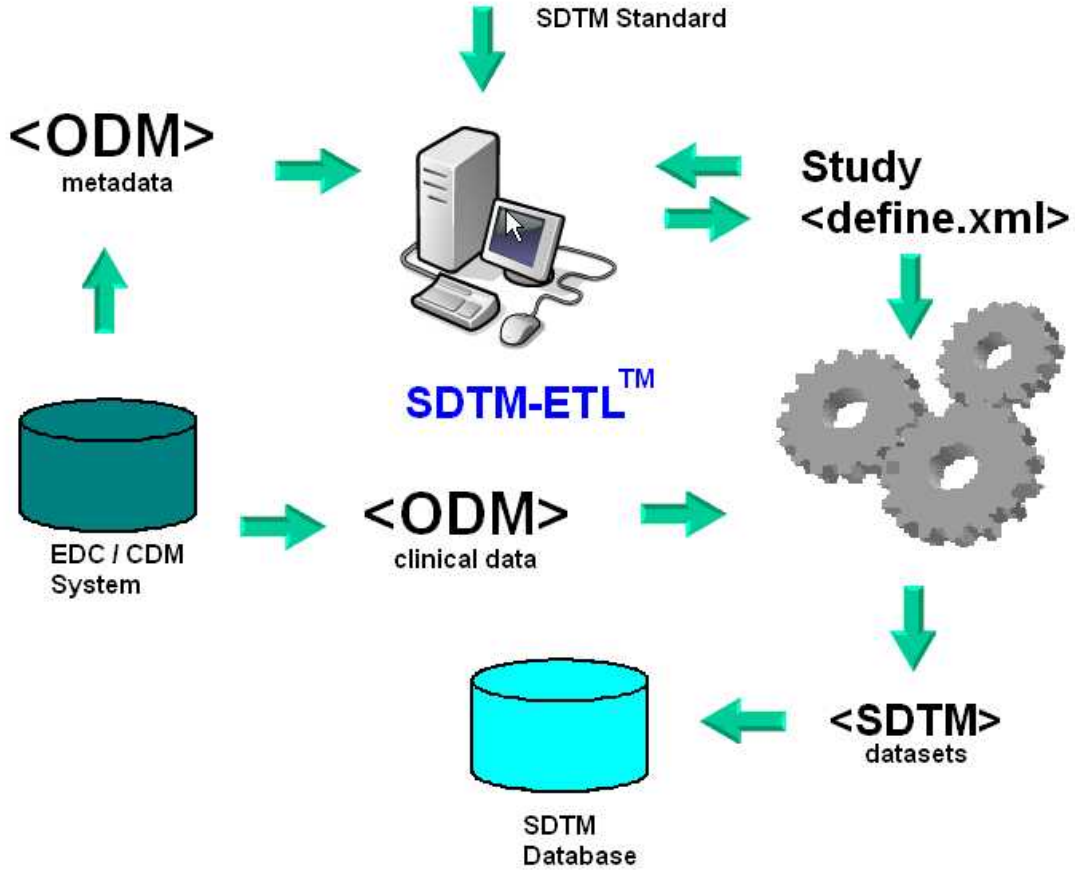
New features in versions 1.3 and 1.2

Version 1.2 allows to create ***sponsor-defined domains***, based on the rules as described in the SDTM Implementation Guide (section 2.6). Also new in version 1.2 is that the user can add ***additional SDTM variables*** for an existing domain according to the domain class rules (SDTM-IG section 2.4). For example, one can add extra timing variables to existing domains. All this is done according to the strict rules described in the SDTM Implementation Guide.
Also the latest published CDISC Controlled Terminology (package 2A and 2B) from the CDISC Terminology team has been implemented as define.xml codelists.

Version 1.3 allows to add ***non-standard variables*** to the domains. These then go into SUPP--datasets and records at execution time.
Version 1.3 also fully implements the new ***SDTM 1.2*** (SDTM-IG 3.1.2)

6 ANSI SQL is generated

USUBJID	CM.CMSEQ	CM.CMGRPID	CM.CMSPID
USUBJID	SU.SUSEQ	SU.SUGRPID	SU.SUSPID
USUBJID	AE.AESEQ	AE.AEGRPID	AE.AEREFID
USUBJID	DS.DSSEQ	DS.DSGRPID	DS.DSREFID
USUBJID	DV.DVSEQ	DV.DVCAT	DV.DVTERM
USUBJID	MH.MHSEQ	MH.MHGRPID	MH.MHREFID
USUBJID	EG.EGSEQ	EG.EGGRPID	EG.EGREFID
USUBJID	IE.IESEQ	IE.IESPID	IE.IETESTCD



The SDTM-ETL process