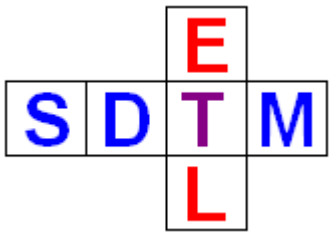
SDTM-ETL™



New features in version 2.2

Author: Jozef Aerts – XML4Pharma
February 2013

Table of Contents

Introduction

SDTM-ETL™ 2.2 is a new, improved version of our popular ETL software for the generation of mappings between operational data and metadata in CDISC ODM format, and the CDISC SDTM or SEND standard.

In version 2.2 the "CDISC SDTM 1.2 Amendment 1" has been replaced by a full implementation of the final version 1.3 of the SDTM standard (SDTM-IG 3.1.3).

Furthermore, version 2.2 now comes with the new, by CDISC published, medical devices domains (DI, DU, DX, DE, DT, DR, and DO), the draft oncology domains (TU, TR, RS), the draft non-subject-data domains (NSDM, NSSU, NSMH, NSLB, NSAE, NSSC), and a number of the draft SDTM 1.4 (SDTM-IG 3.1.4) domains (CV, DD, MI, MO, PR, SS, TD).
The templates for these new domains can be merged with the already existing domains in case the user would like to use these new domains.

The major changes in this release are however behind the curtains: a considerably lower memory footprint, improved drag-and-drop performance, and considerably better performance especially in the case of very large mappings.

Some of the dialogs have been further improved, such as the dialog to "generalize" for StudyEvents, Forms, ItemGroups and/or Items. Instead of only allowing to define "exceptions", it is now also possible to do active selections, i.e. define "only for" selections.

Furthermore, on request of the FDA, the SDTM/SEND variable length can be optimized starting from the operational clinical data, thus avoiding waste of space in the result SAS XPT files.

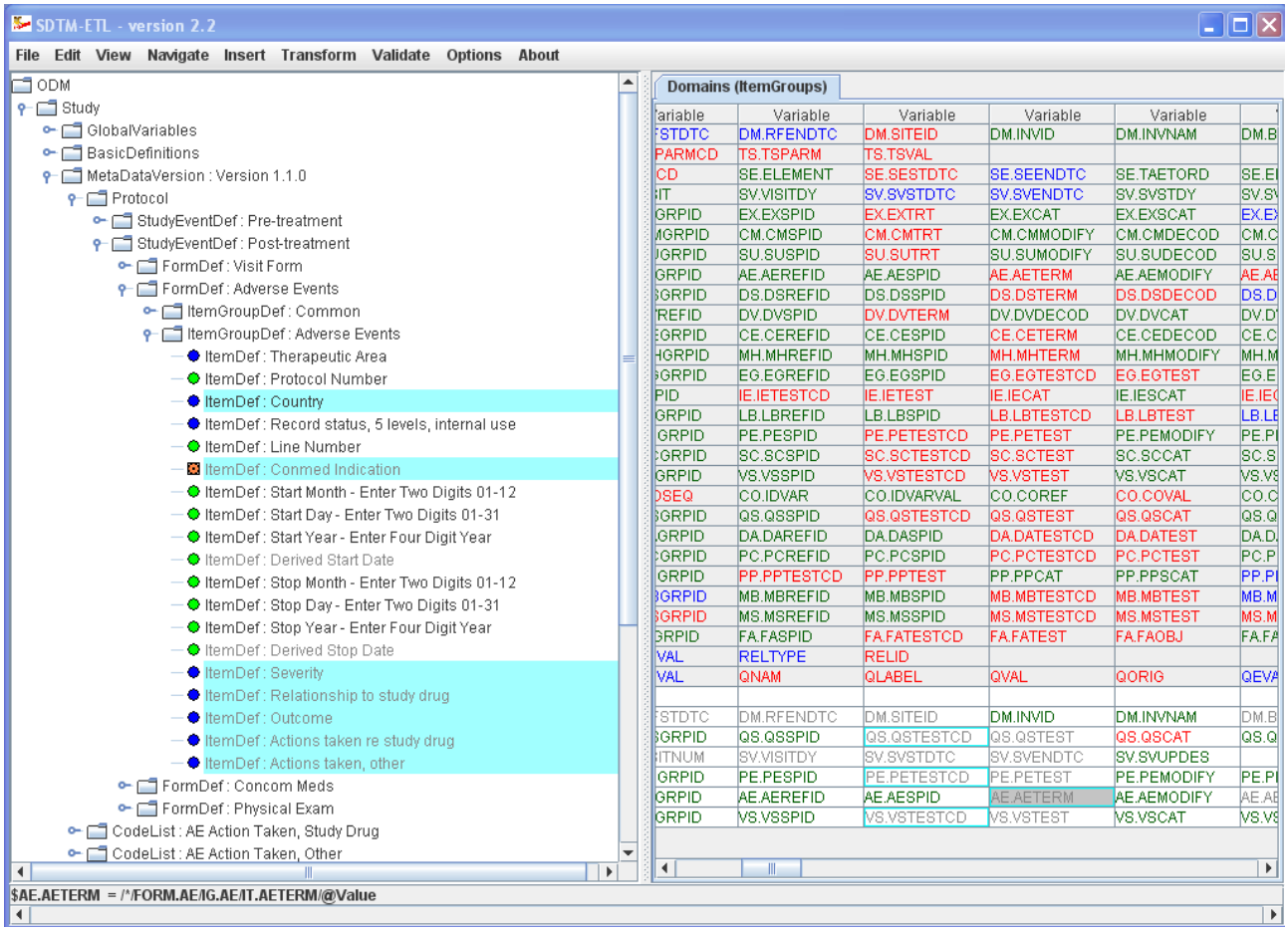
We are very proud to be able to present SDTM-ETL v.2.2.
We are convinced that this is the best tool on the market for mapping operational clinical data to submission data (either SDTM or SEND), with the lowest total cost of ownership (TCO).

New domain templates: medical devices, oncology, draft SDTM 1.4 domains

CDISC recently published version 1.0 (final) of the medical devices domains. These are:

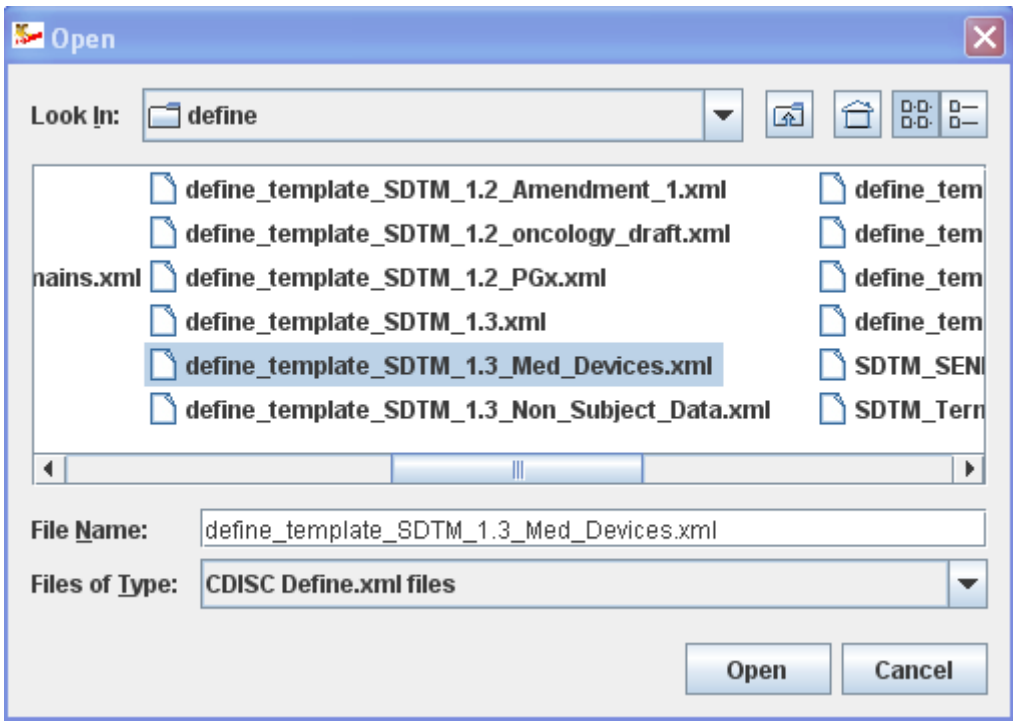
- Device Identifiers (DI)
- Device In-Use (DU)
- Device Exposure (DX)
- Device Events (DE)
- Device Tracking and Disposition (DT)
- Device-Subject Relationships (DR)
- Device Properties (DO)

In order to use these domains, you will need to merge them with the already existing template domains[1] or existing study mapping. For example, when starting from an existing mapping:



Now use the menu “File -> Load Template define.xml” (this is a new menu item). You will be directed to the “define” directory that is part of your installation and where the template define.xml files reside. You can then choose an additional template define.xml, e.g. the template define.xml containing the new Medical Devices domains:

1 It is unimportant which version of SDTM you are using



When clicking the "Open" button, the template will be merged with your existing define.xml. The result on the screen is:

FA	STUDYID	DOMAIN	USUBJID	FA.FASEQ	FA.FAGRPID	FA.FASPID	FA.FATESTCD	FA.I
RELREC	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	RELTYPE	RELID	
SUPPQUAL	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	QNAM	QLABEL	QV
TE	STUDYID	DOMAIN	TE.ETCD	TE.ELEMENT	TE.TESTRL	TE.TEENRL	TE.TEDUR	
TA	STUDYID	DOMAIN	TA.ARMCD	TA.ARM	TA.TAETORD	TA.ETCD	TA.ELEMENT	TA.
TI	STUDYID	DOMAIN	TI.IETESTCD	TI.IETEST	TI.IECAT	TI.IESCAT	TI.TIRL	TI.T
TV	STUDYID	DOMAIN	TV.VISITNUM	TV.VISIT	TV.VISITDY	TV.ARMCD	TV.ARM	TV.
DI	STUDYID	DOMAIN	SPDEVID	DI.DISEQ	DI.DIPARMCD	DI.DIPARM	DI.DIVAL	
DU	STUDYID	DOMAIN	USUBJID	SPDEVID	DU.DUSEQ	DU.DUGRPID	DU.DUREFID	DU
DX	STUDYID	DOMAIN	USUBJID	SPDEVID	DX.DXSEQ	DX.DXGRPID	DX.DXSPID	DX.
DE	STUDYID	DOMAIN	USUBJID	SPDEVID	DE.DESEQ	DE.DESPID	DE.DETERM	DE.
DT	STUDYID	DOMAIN	SPDEVID	DT.DTSEQ	DT.DTTERM	DT.DTMODIFY	DT.DTDECOD	DT.
DR	STUDYID	DOMAIN	USUBJID	SPDEVID				
DO	STUDYID	DOMAIN	SPDEVID	DO.DOSEQ	DO.DOGRPID	DO.DOREFID	DO.DOSPID	DO
MyStudy:GLOBAL	RFSTDTC	RFENDTC						
MyStudy:DM	STUDYID	DOMAIN	USUBJID	SUBJID	DM.RFSTDTC	DM.RFENDTC	DM.SITEID	DM
MyStudy:QS	STUDYID	DOMAIN	USUBJID	QS.QSSEQ	QS.QSGRPID	QS.QSSPID	QS.QSTESTCD	QS.
MyStudy:SV	STUDYID	DOMAIN	USUBJID	SV.VISIT	SV.VISITNUM	SV.VISITDY	SV.SVSTDTC	SV.
MyStudy:PE	STUDYID	DOMAIN	USUBJID	PE.PESEQ	PE.PEGRPID	PE.PESPID	PE.PETESTCD	PE.
MyStudy:AE	STUDYID	DOMAIN	USUBJID	AE.AESEQ	AE.AEGRPID	AE.AEREFID	AE.AESPID	AE.
MyStudy:VS	STUDYID	DOMAIN	USUBJID	VS.VSSEQ	VS.VSGRPID	VS.VSSPID	VS.VSTESTCD	VS.

You can now instantiate (using drag-and-drop or using the menu "Edit -> Copy Domain/Dataset" followed by "Edit -> Paste Domain/Dataset") one or more of the additionally loaded template domains for your study.

Other new domains for which a template is provided are the new oncology (draft) domains (file "define_template_SDTM_1.2_oncology_draft.xml") and a number of the new SDTM 1.4 draft domains (file "define_template_SDTM_1.4_new_domains.xml");

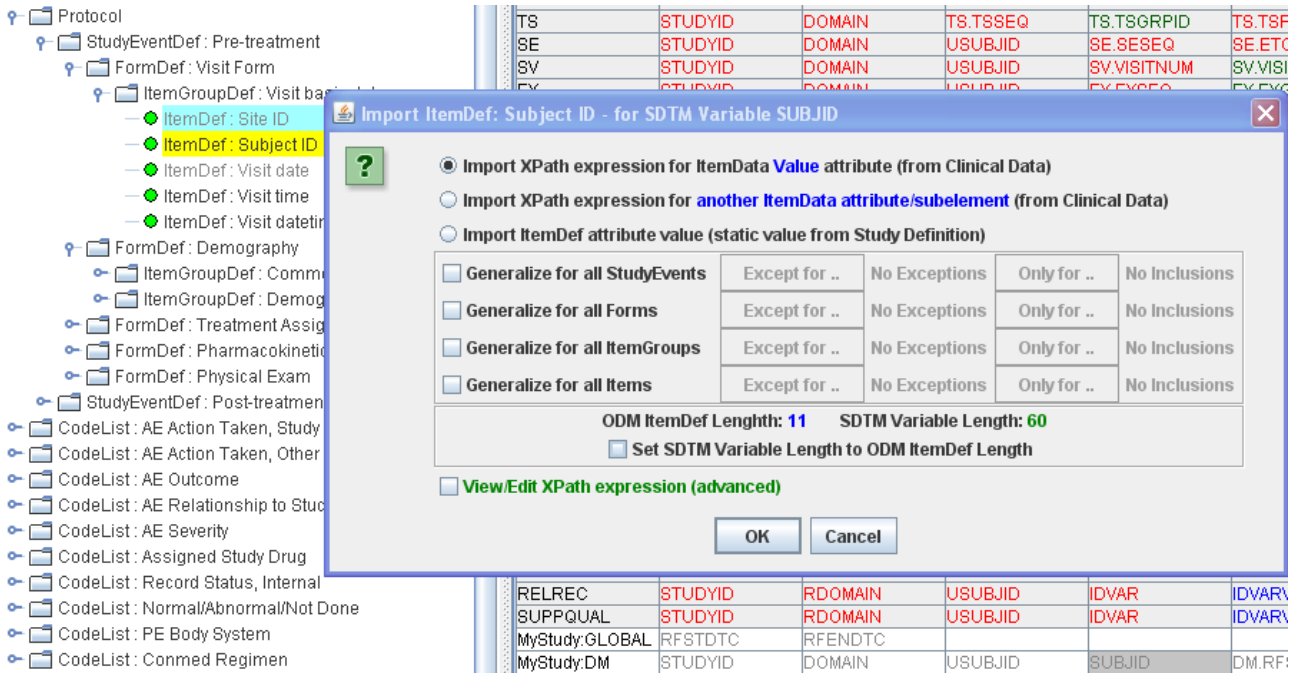
New functionality: suggestion for the SDTM/SEND Variable Length

In its publication "CDER/CBER's Top 7 CDISC Standards Issues", the FDA complainted that in many submissions, the SDTM/SEND variable lengths have been chosen inadequately, leading to

large file sizes for the SAS XPT files[2]. For example, many sponsors always set the variable length for most variables to the maximum of 200, leading to very large file sizes[3].

In SDTM-ETL, the default variable lengths (from the templates) are 8 (for numeric values), 40 (for test labels) and 60 or 80 (for other variables). In case an SDTM variable is mapped to an ODM Item, the length of the SDTM variable is compared with that of the ODM Item, and if the latter is larger, the user is asked whether the SDTM variable length must be adapted to the one from the ODM Item.
The user can also always manually define or change the length for the SDTM variable using the menu "Edit -> SDTM Variable Properties".

In the new version, additional functionality has been added to allow to automatically copy the variable length from the ODM Item to the SDTM variable when doing drag-and-drop.
For example, the default length for the SDTM variable "SUBJID" in the DM domain is 60.
When now doing drag-and-drop from the ODM Item "Subject ID" in the "Visit basic data" form of the first visit, the user is now presented the SDTM variable length as well as the ODM Item length:



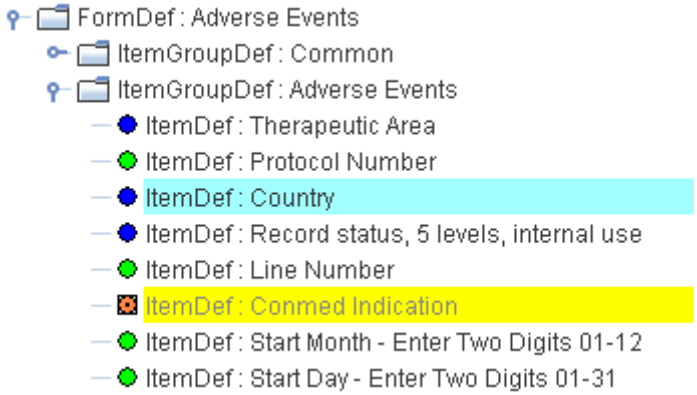
The ODM Item length being 11 and the SDTM variable length having the default value of 60. At this point, a checkbox is presented "Set SDTM Variable Length to ODM ItemDef Length".
When checked, the software automatically adapts the variable length for SDTM variable "SUBJID" to 11, and the mapping wizard continues its work.
If the mapping is interrupted or cancelled, the user is asked whether he/she wants to revert to the original value of 60 for the SDTM variable length:

2 As well known, SAS XPT is very inefficient with disk space. For example, if the longest AETERM value has 100 characters, the variable length needs to be set to at least 100. All other values for AETERM will then also automatically take 100 characters, even if the value has only a few characters, the remaining being filled with blanks. Even worse, many sponsors set the variable length for many variables to the maximum value of 200 (just to be sure), leading to extremely large file sizes.

3 In fact, file size itself should not be a problem - it is the memory usage that can become a problem if the software is not "smart" enough to deal with variable values that have a lot of tailing blanks. Apparently, the FDA's standard software tools are not able to deal with such issues.

It might however be that in the study design the maximal length for an ODM Item has been set to another (usually higher) value than found for the actual captured values. For example, for an "AE reported term", the maximal length may have been set to e.g. 100, but after closing the database, the actual maximal length for this item was found to be e.g. 20.
Also in such a case it may be worth setting the length of the corresponding SDTM variable (in this case AETERM) to 20, the length of the actually largest (in character length) found answer in the database.

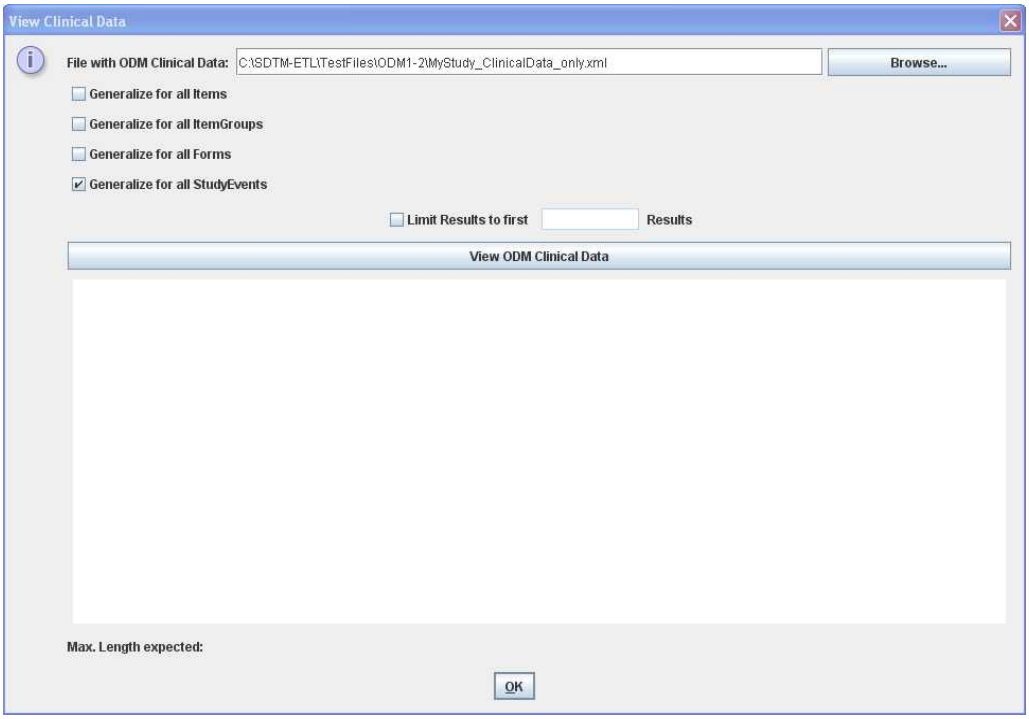
In order to do so, select the ODM Item for which one would like to find the maximal length encountered. In our example study, this is the ODM Item "Conmed Indication" (OID "IT.AETERM") which was annotated as being a "hot candidate" for the mapping to the SDTM "AETERM" variable (using the "SDSVarName") attribute[4].



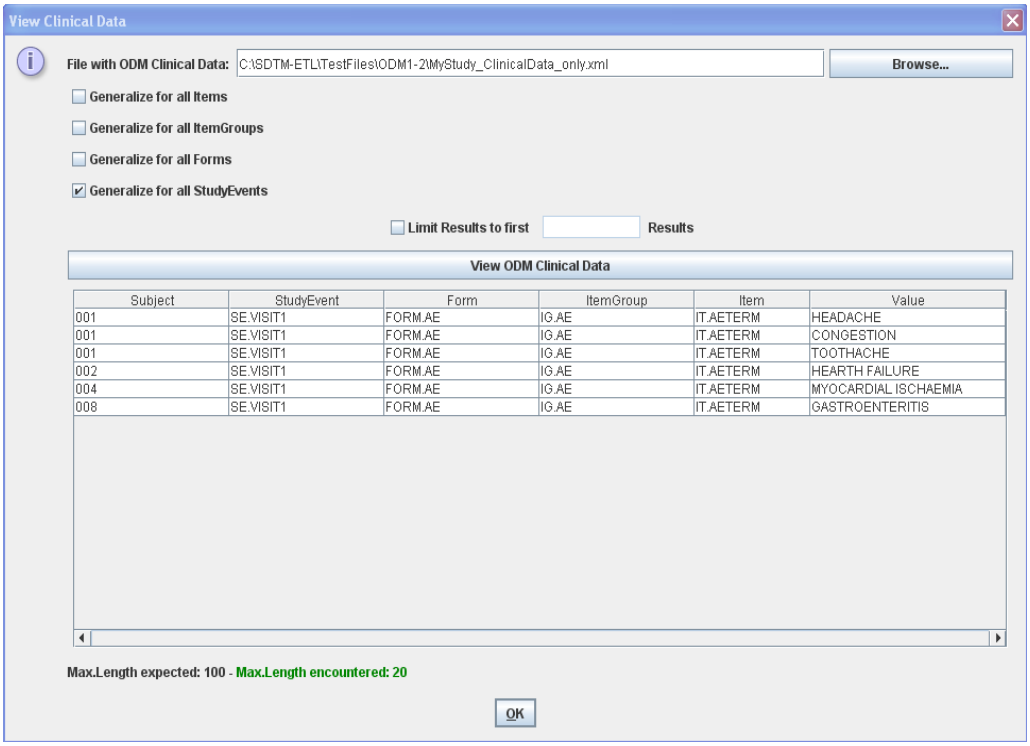
One can see that "Conmed Indication" is a "hot candidate" as it is marked with a square in front.

Then use the menu "View -> ODM Clinical Data". The following dialog shows up:

4 For information about annotating ODM for SDTM, see our blog "Annotating ODM with SDTM and CDASH information".



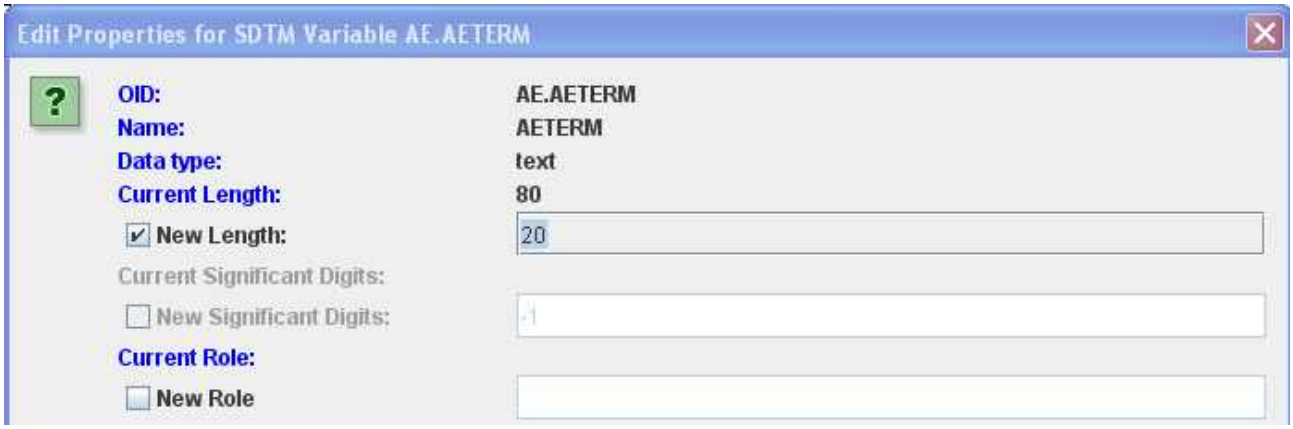
One can choose a file with ODM clinical data.
In our case, we also select "Generalize for all StudyEvents" as we would like to obtain the reported terms for all the visits, and not only those for the visit that was selected in the ODM tree.
Upon clicking "View ODM Clinical Data", the following table is displayed:



Remark the label near the bottom: it states that in the database, the maximal length was set to 100, but the longest actual captured value had a length of only 20.
So if you are sure that these are the final captured data, you may decide on setting the length for the corresponding SDTM variable AETERM to 20.

In order to do so, close the dialog by clicking OK, be sure that you selected “AETERM” in the SDTM table, and use the menu “Edit -> SDTM Variable Properties”. The following dialog appears (partial view):

We see that the length for the SDTM variable AETERM is still 80 (which was the default value from the template). You can now set it to 20 by checking the “New Length” checkbox, and fill in “20” in the textfield on the right:

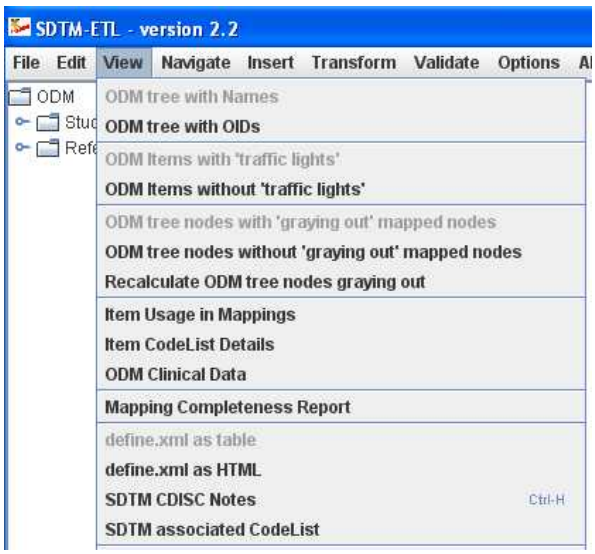


After clicking the “OK” button, the value for the length of the AETERM variable in the underlying define.xml is updated to “20”.

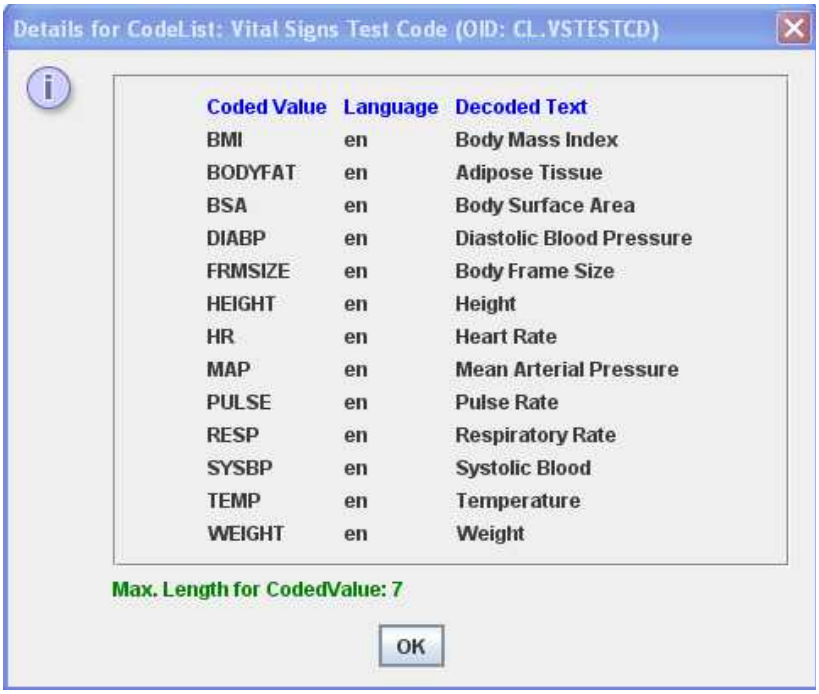
SDTM codelists and variable lengths

When an SDTM variable is coded, i.e. there is an associated codelist, then it is not always simple to find out to what value the SDTM variable length should be set.
Two new features make it now easier to find this out.

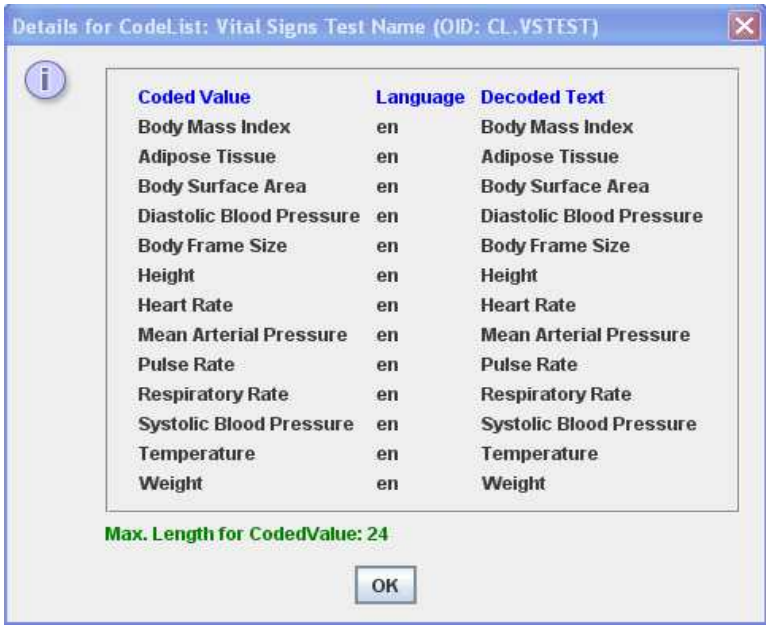
In the previous versions, it was already possible to see the details of an associated codelist for an SDTM variable by selecting the variable in the table, and then using the combination CTRL-right-click. The same can now however also be accomplished by the menu “View -> SDTM associated CodeList”:



This brings up the following dialog:



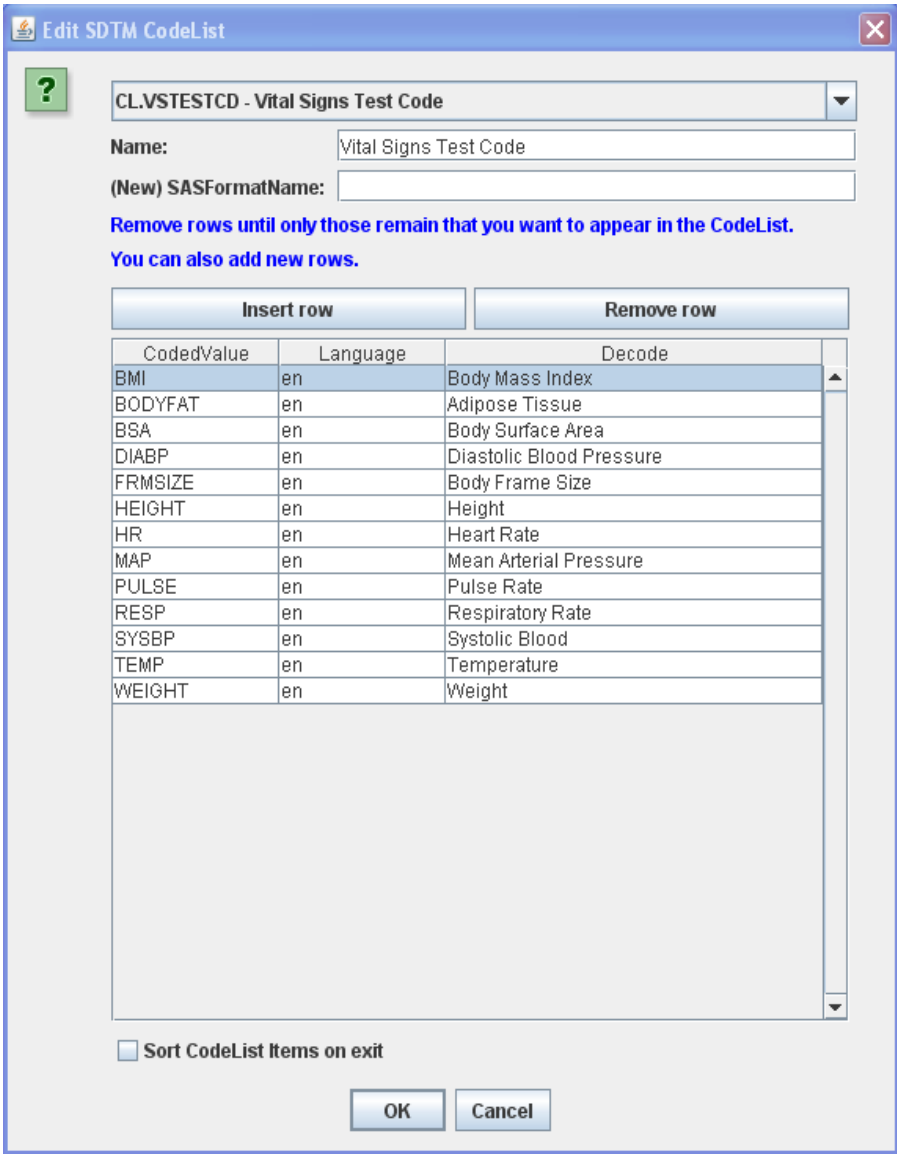
New in this dialog is the last line: it shows what the maximal length of the coded values in the CodeList is. This value can then be used to set the length value of the corresponding SDTM variable. For example, in this case, the length for the variable VSTESTCD can be set to 7, as no coded value in the associated codelist is longer than 7 characters.
Similarly, for the variable VSTEST (test name), the length can be set to 24, as no coded values in the associated codelist CL.VSTEST are longer than 24 characters:



Edit SDMT CodeList

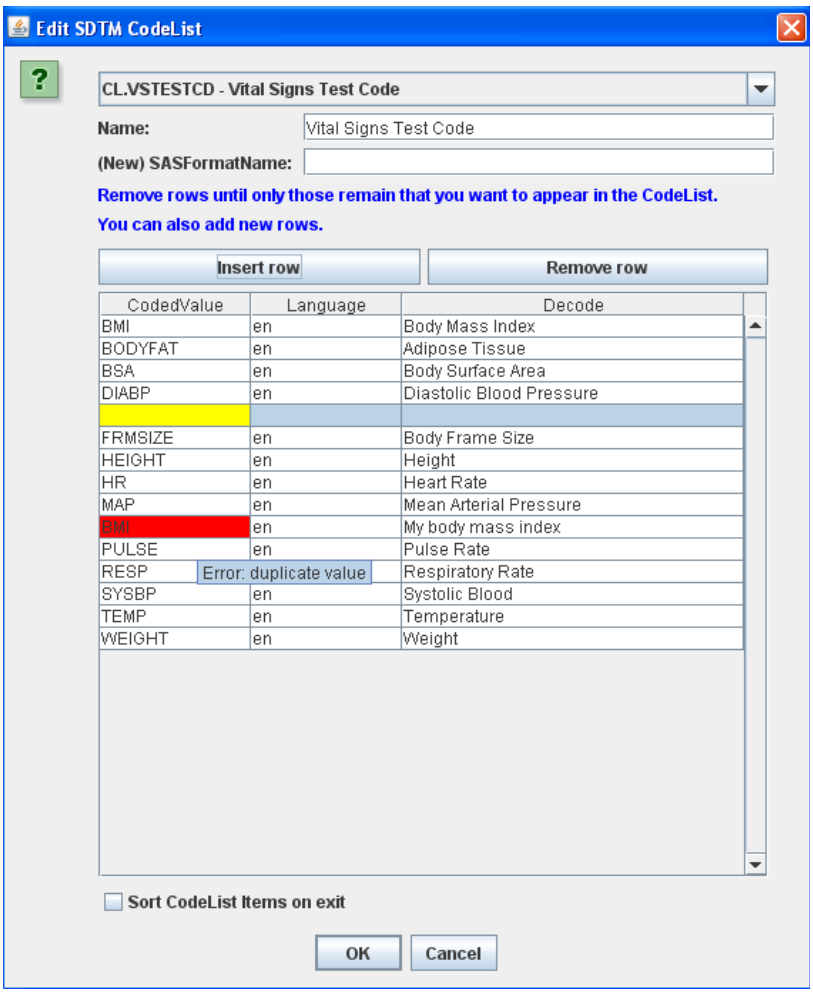
In the previous versions it was possible to make a clone of an SDTM codelist and then change the properties (OID, Name, coded values) using the menu “Insert -> Create new SDTM CodeList from existing CodeList”. This is still the xxx way to generate subsets of by CDISC published codelists. For example for the VSTESTCD codelist, one can so clone that list, and then only keep those codes tha were really used in the study. Similarly, one can so generate a codelist with units of measurement starting from the CDISC list (containing hundreds of codes) and then only keep those units that were really used for a specific test.

In the new version, it is also possible to edit an existing SDTM codelist without cloning it.
This can be done using the menu “Edit -> SDTM CodeList”. The following dialog then shows up:



The selection box at the top allows to select a specific SDTM codelist. One can then alter the name ("Name" attribute in the define.xml) and the "SASFormatName" (when desired). The OID of the codelist can not be changed. Use the cloning feature if you also want to change the OID, but be aware that you may loose the correspondence with the SDTM variable.

In the dialog "Edit SDTM variable", one can then remove coded values and add new ones. During doing so, it is checked whether the normal rules are obeyed, i.e. a coded value cannot be empty and it may also not be duplicate. This is demonstrated below:



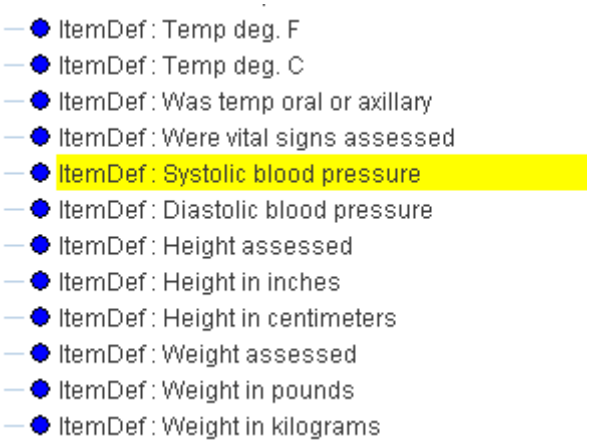
The empty "Coded Value" cell is colored yellow, and the cell with the duplicate value is colored red. It will also not be possible to store this until these errors are corrected.

Near the bottom one observes a checkbox "Sort CodeList Items on exit". When it is checked, the order of the coded values will be sorted alphabetically in case the datatype is "text" and numerically in case the datatype is "integer". This is as CDISC has the custom of publishing coded values in controlled terminology in alphabetical order[5].

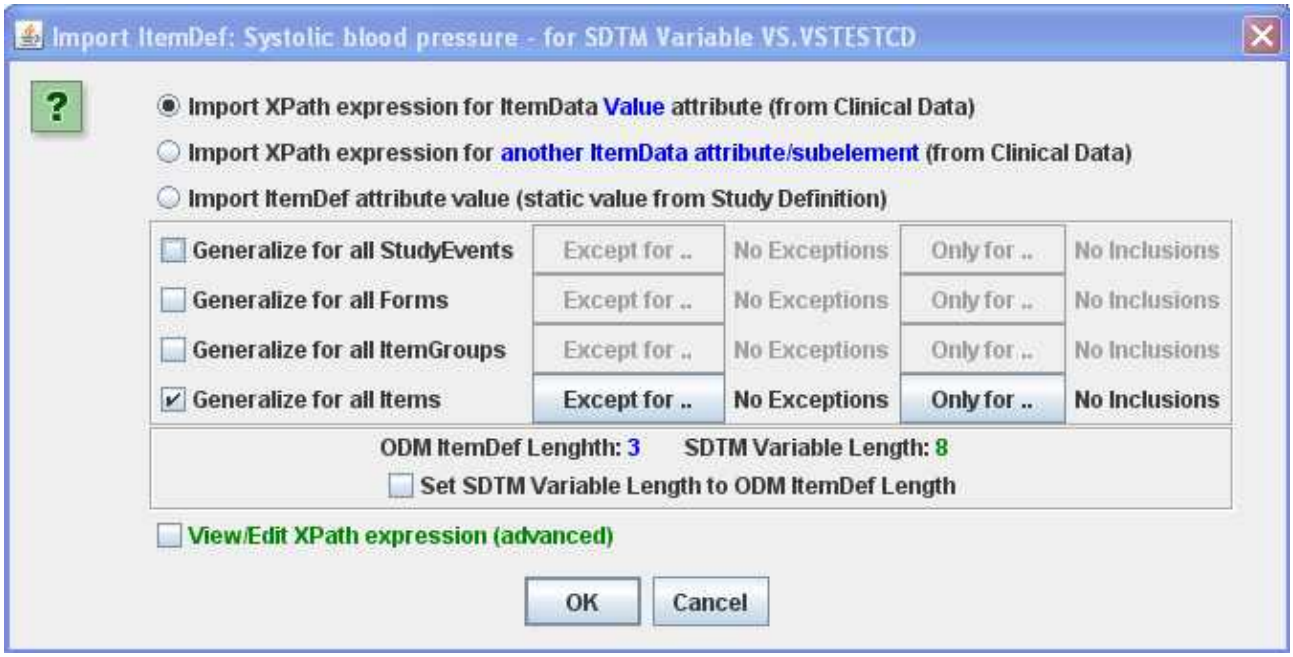
New feature: "positive" selection during "Generalize for ..."

When doing a "Generalization", it was until now only possible to exclude certain Items, ItemGroups, Forms, or Visits. As of this version, it is also possible to "positively" select some items. This is extremely useful when in a long list of e.g. Items with an ItemGroup, only a few need to be selected. For example if we have an ODM group with the following Items:

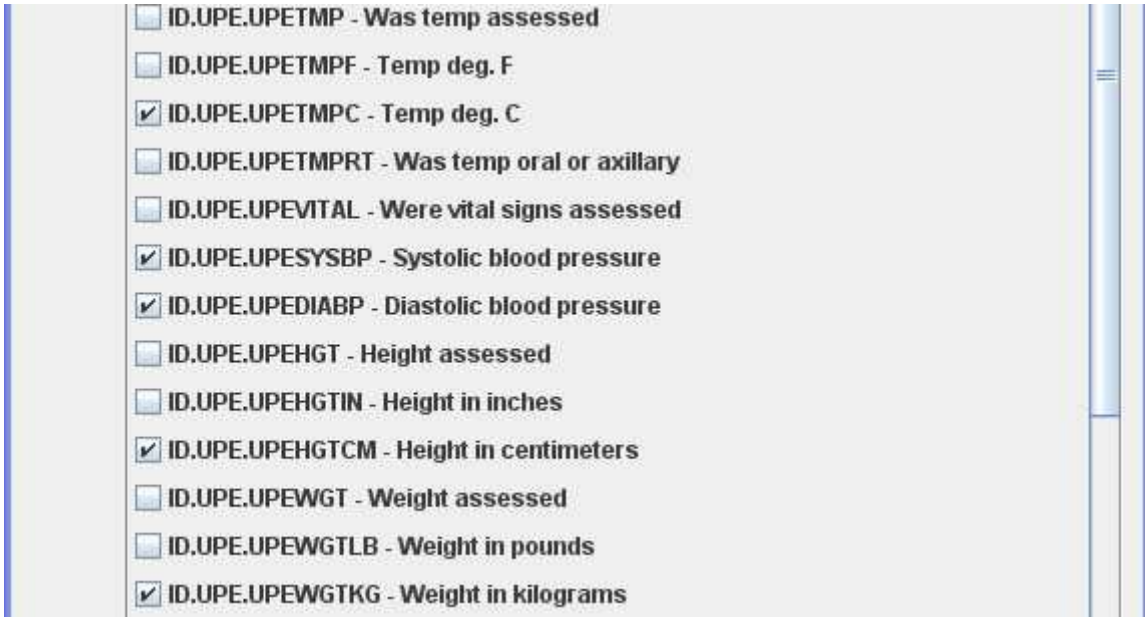
5 In most cases however, the order does not matter at all.



(not all shown in this picture)
and we only want to select “Temp def.C”, “Systolic blood pressure”, “Diastolic blood pressure”, “Height in centimeters” and “Weight in kilograms”, we can drag-and-drop one of these Items to VSTESCD in the SDTM table, for which the following dialog will be presented:



Remark the button “Only for ..” which is new.
In the old version, we needed to use the button “Except for ..” and then select all the Items within the ItemGroup which we did not want to have included. In the new version however, we can use the button “Only for ..” and then do the selection, e.g.:



When clicking OK, the following script is generated:

```
# Generalized for all Items within the ItemGroup
# Using categorization as a CodeList is associated with the SDTM CodeList
# but no CodeList is associated with the ODM data
$CODEDVALUE =
xpath(/StudyEventData[@StudyEventOID='SE.SV']/FormData[@FormOID='FORM.UPE']/ItemGroupData[@ItemGroupOID='IG.UPE']/ItemData[@ItemOID='ID.UPE.UPETMPC' or @ItemOID='ID.UPE.UPESYSBP' or @ItemOID='ID.UPE.UPEDIABP' or @ItemOID='ID.UPE.UPEHGTCM' or @ItemOID='ID.UPE.UPEWGTKG']/@Value);
```

Remark the “or” in the XPath predicate (condition).

It is not possible to “mix” “exclusions” (“except for ..”) and “inclusions” (“only for”) when using the dialogs. The software takes care of it that when one switches between “except for ..” and “only for ..” the selections are reverted.

Performance improvements

A lot of work has been done to obtain considerabe performance improvements.
In the past, the software was found to become relatively slow in case of very large and complex study designs, containing thousands of Items in the extended ODM tree.

Each time a cell was clicked in the SDTM or SEND table, the color of "traffic lights" (indicating suitability for mapping based on data type) of the ODM Items need to be recalculated. Therefore the possibility of viewing the ODM Items without traffic lights was introduced in version 2.0.

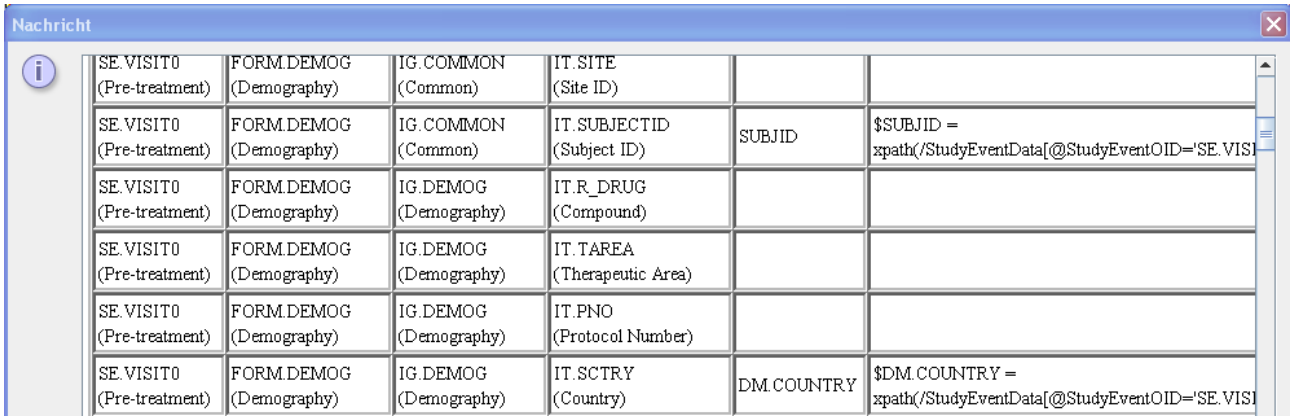
Furthermore, ODM Items that are used in one or more mappings need to be "grayed out" in the ODM tree.

In the case of very many ODM Items (thousands) these calculations can take more than a second, leading to a slower performance. Therefore, new algorithms were introduced for doing these calculations which are very much faster. For example, the "graying" out of ODM Items is only recalculated after a mapping has been added or changed. Also when loading a new define.xml file, the "graying out" of in mappings used ODM Items is recalculated.

For the user, this is visible through a "progress bar" that is displayed when such a recalculation is performed:

The user can however also enforce a recalculation of in mappings used ODM Items using a new menu item: "View -> Recalculate ODM tree nodes graying out".
The recalculation is also automatically done when the user requests a "mapping completeness report" using the menu "View -> Mapping Completeness Report".

For example, the following part of the "Mapping Completeness Report" shows that the ODM Items "Site" (OID: IT.SITE), "Compound" (OID:IT.R_DRUG), "Therapeutic Area" (OID: IT.TAREA) and "Protocol Number" (IT.PNO) have not been used (yet) in any of the mappings.


Nachricht

SE.VISIT0 (Pre-treatment)	FORM.DEMOG (Demography)	IG.COMMON (Common)	IT.SITE (Site ID)		
SE.VISIT0 (Pre-treatment)	FORM.DEMOG (Demography)	IG.COMMON (Common)	IT.SUBJECTID (Subject ID)	SUBJID	$SUBJID = xpath(/StudyEventData[@StudyEventOID='SE.VISI
SE.VISIT0 (Pre-treatment)	FORM.DEMOG (Demography)	IG.DEMOG (Demography)	IT.R_DRUG (Compound)		
SE.VISIT0 (Pre-treatment)	FORM.DEMOG (Demography)	IG.DEMOG (Demography)	IT.TAREA (Therapeutic Area)		
SE.VISIT0 (Pre-treatment)	FORM.DEMOG (Demography)	IG.DEMOG (Demography)	IT.PNO (Protocol Number)		
SE.VISIT0 (Pre-treatment)	FORM.DEMOG (Demography)	IG.DEMOG (Demography)	IT.SCTRY (Country)	DM.COUNTRY	$DM.COUNTRY = xpath(/StudyEventData[@StudyEventOID='SE.VISI