SDTM-ETL 5.0 User Manual and Tutorial
Editing Variable Properties

Author: Jozef Aerts, XML4Pharma

Last update: 2025-02-27

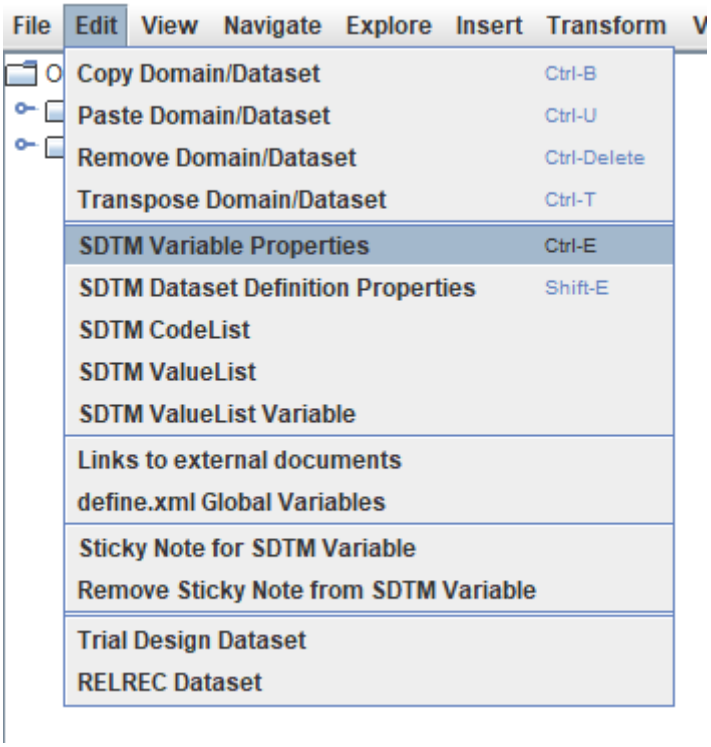
Editing the define.xml ItemDef properties (SDTM Variable properties)

Table of contents:

Setting the maximal length of the variable 2
Assigning or changing the codelist 3
Setting and editing the "Origin" of the data point: define.xml 2.0 7
Setting and editing the "Origin" of the data point: define.xml 2.1 9

In the define.xml, an SDTM Variable is defined by its "ItemRef – ItemDef" pair.
So, if we want to e.g. change the maximal (string) length of an SDTM variable, we need to edit the "ItemDef" properties of that SDTM variable in the underlying define.xml structure. In SDTM-ETL, this is however mostly hidden for the normal user, but implemented in a user-friendly way.

In order to change the properties of an SDTM variable, select a cell representing an SDTM variable (in this example case, select QS.QSTEST) in the SDTM table and select "Edit -> SDTM Variable Properties[1]".

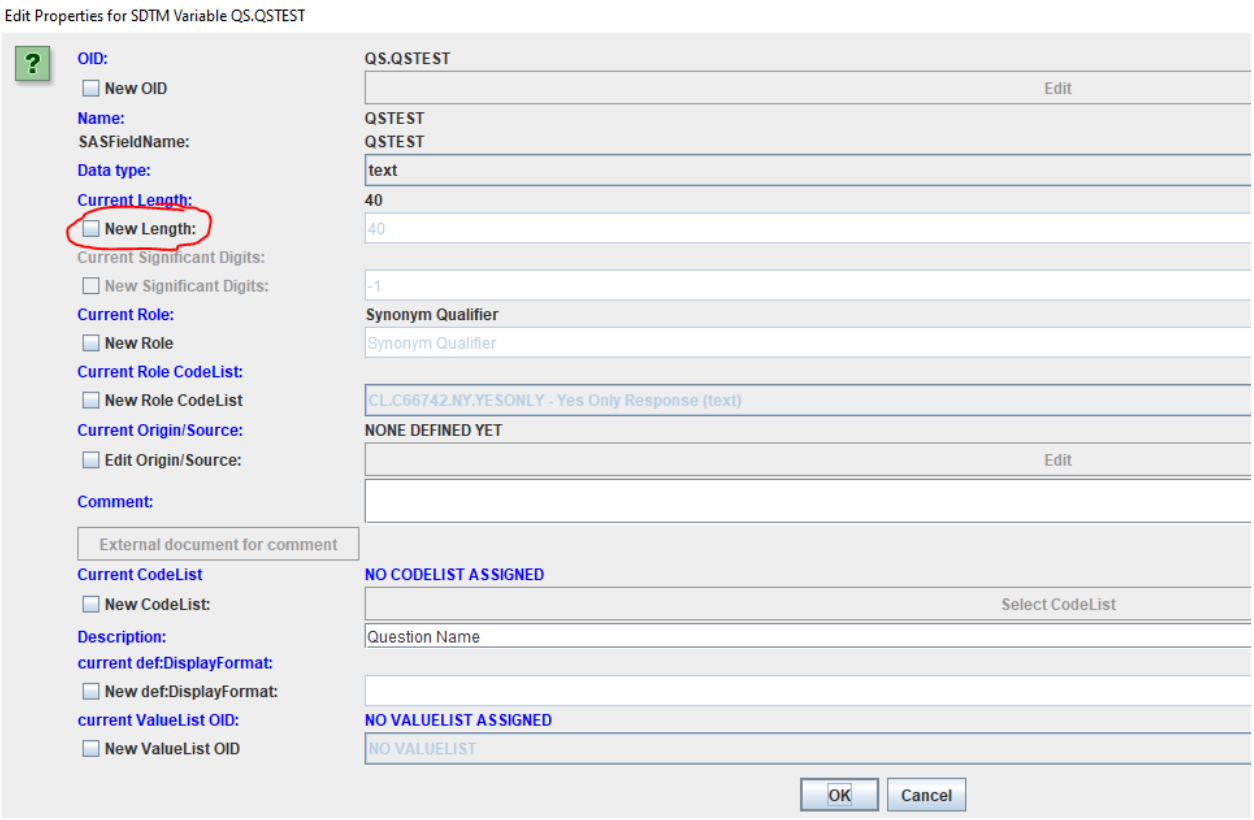


Setting the maximal length of the variable

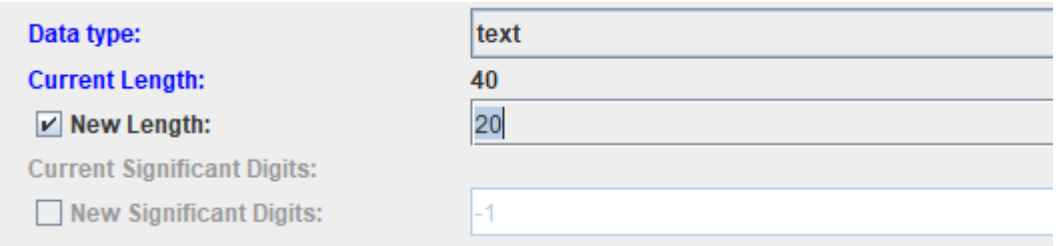
[1] In the case of SEND, the menu will be "Edit - SEND Variable Properties"

Especially in the case of SAS-XPT output, it is important to have a correct maximal length assigned[2]. So we might set the maximal length in advance when we know what it will be.
When then generating the datasets, there will still be an opportunity for that the system calculates the maximal length itself, which may be helpful.

When using "Edit - SDTM Variable Properties" dialog is shown, allowing to edit some of the properties of the corresponding define.xml ItemDef:
The default length for QS.QSTEST was 40, but we set it to 20, because we know that its value will never be larger than 20:



We tick the "New Length" checkbox. An input field becomes available, where we type "20":



In case of a numeric variable, like an --STRESN (Numeric Result/Finding in Standard Units) variable, and having assigned the "Data type" as "float", also the field checkbox "New Significant Digits" would become available.

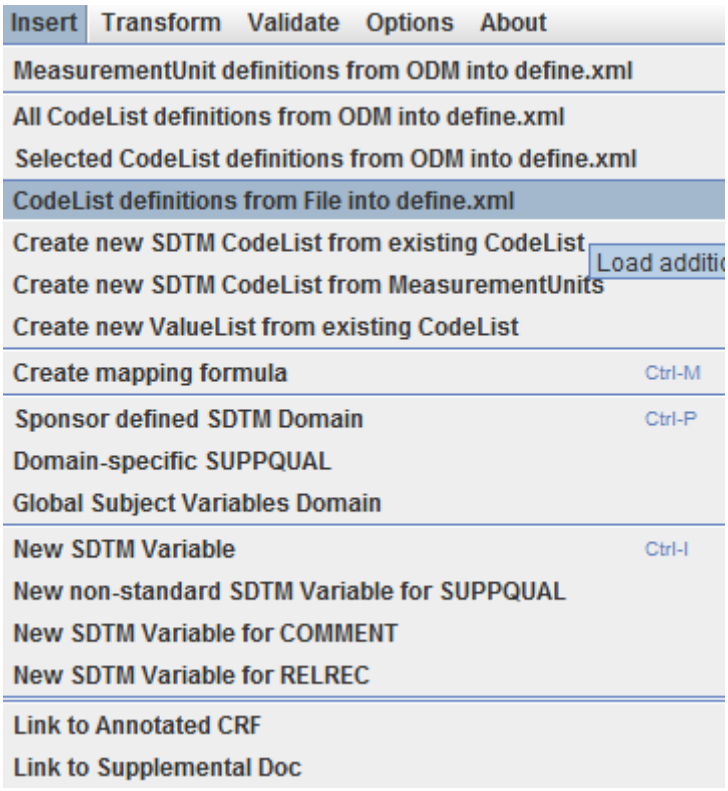
As one sees, one can also change the associated codelist (if any), and some other "ItemDef" properties in the define.xml structure.

---

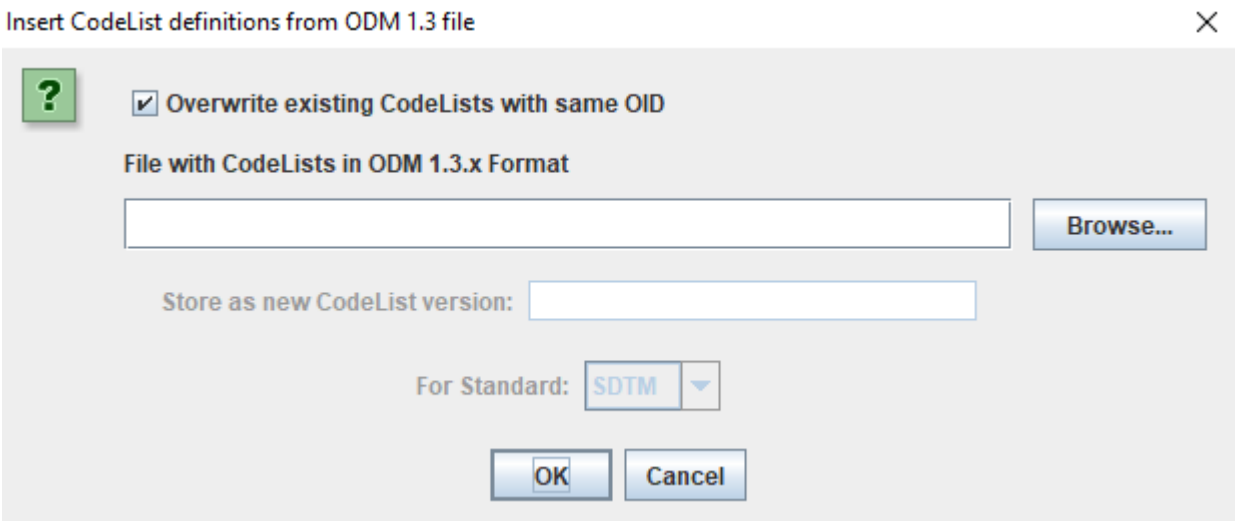
[2] See e.g. the article "How SAS-XPT works" to understand why this is so.

Assigning or changing the codelist

For example, if we know that QSTEST follows CDISC controlled terminology, we may want to insert a QS-codelists, if it was not included yet in the current CDISC-CT that we started at the beginning. For example, if we want to use the codelist "Category of Functional Test" from 2014, we will need to load it separately. In order to do so, first click "OK" and then use the menu "Insert – CodeList definitions from File into define.xml":

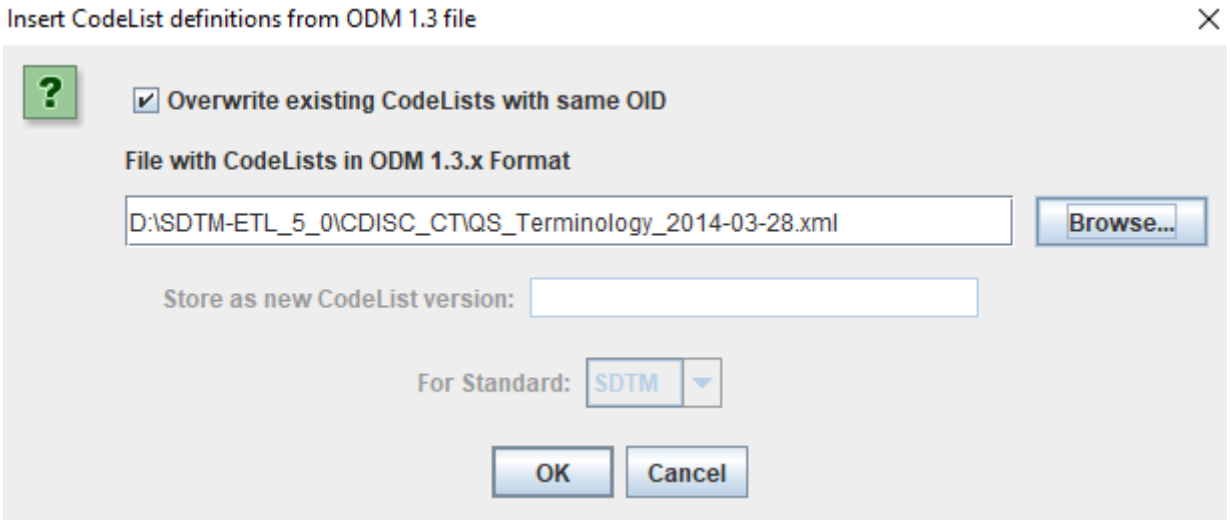


The following dialog is then displayed:

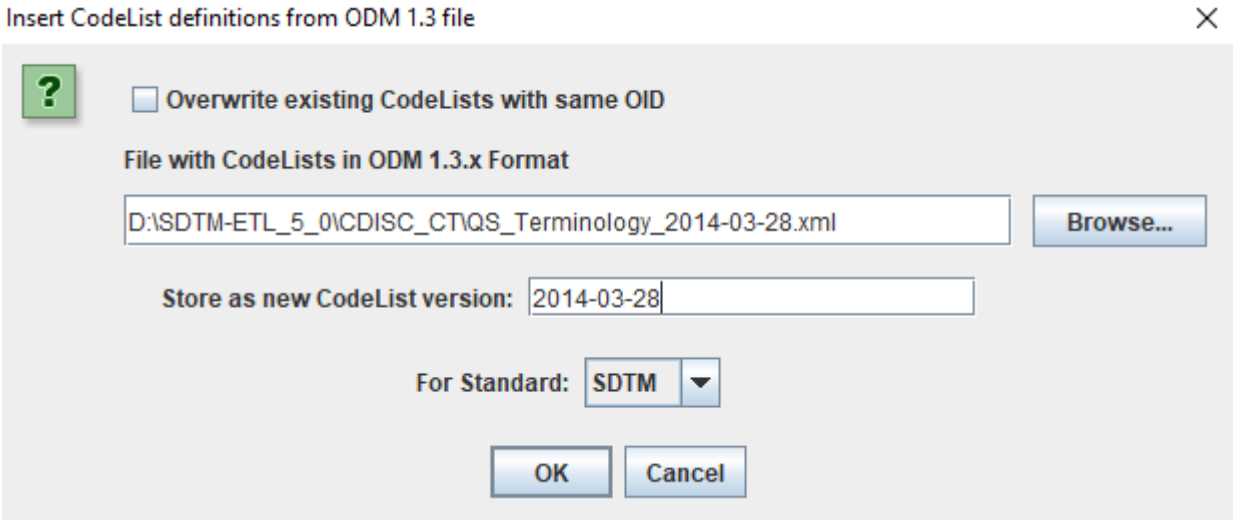


The checkbox "Overwrite …" will take care that if a codelist with the same identifier (OID) is already present, it will be overwritten.

And then using "Browse", e.g. choose the file "QS-FT_Terminology_2014-06-26.xml" (which you can find in the directory "CDISC_CT"):

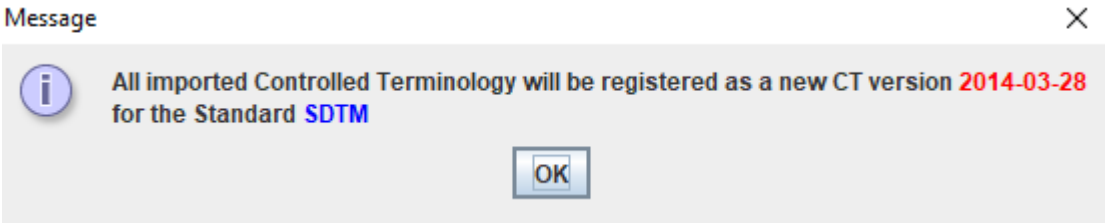


Alternatively, one may want to keep the original version, and must then also fill the field "Store as new CodeList version", e.g.:

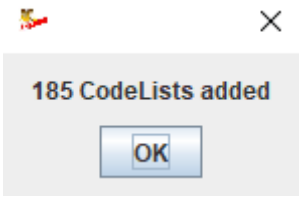


Remark that this will only work for Define-XML v.2.1, as v.2.0 does not have a good mechanism to store different versions of the same codelist.

When we do so, after clicking "OK", a message is displayed:

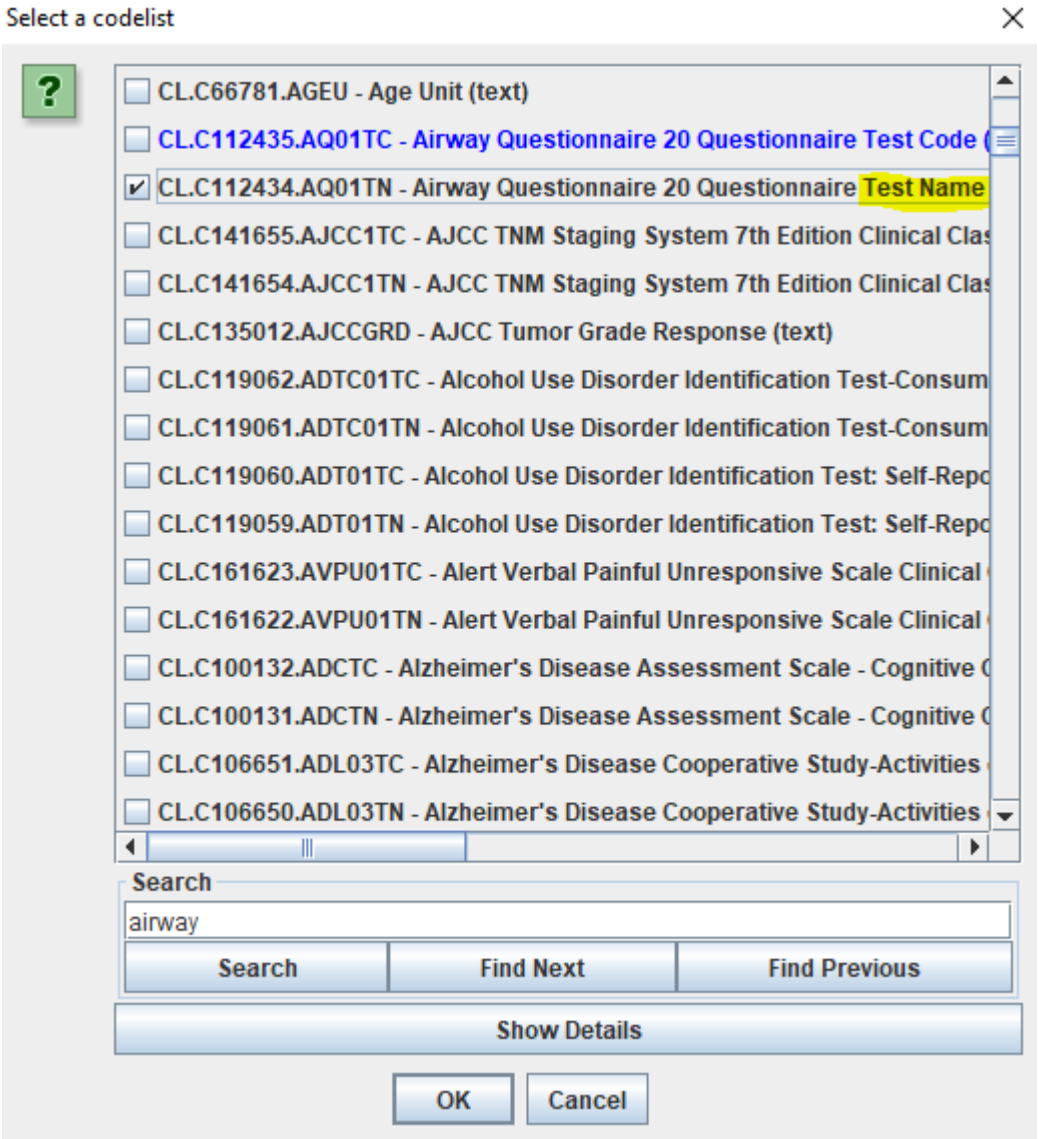


and a confirmation message is displayed:



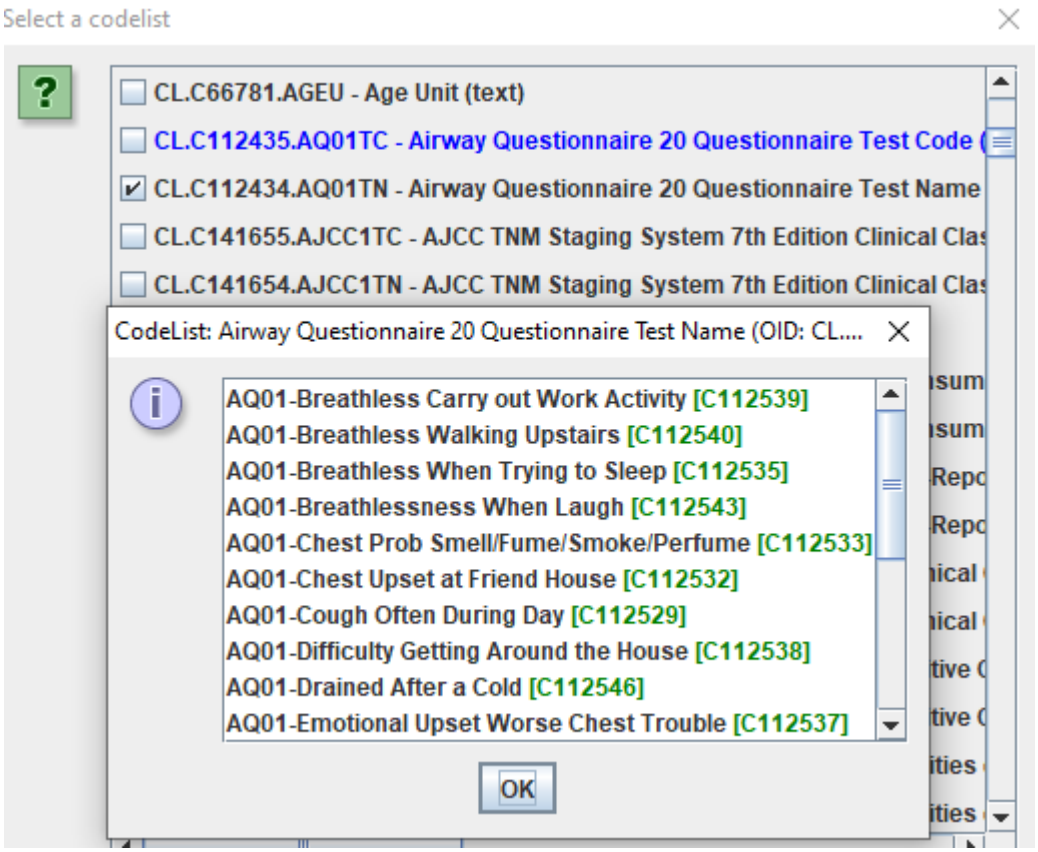
saying that 185 new codelists have been loaded

If you now edit the properties for "QSTEST" again, you can associate the corresponding codelist, using the "New CodeList" checkbox and the dropbox becoming available, e.g.:



As we are assigning to QSTEST, we of course need to select the "AQ01TN" codelist, not the "AQ01TC", as the latter is for QSTESTCD.

"Show details" is always helpful, leading to:



showing all the items from the codelist.

Remark that in most cases, it will not be useful to keep two versions of the same codelist.

Remind that this is just an example, you might need a completely different codelist in your case.

Click OK to confirm your edits.

If you already had a codelist for QSTEST test names in your ODM, you can of course also import that one in the underlying define.xml, e.g. using the menu „Insert – Selected CodeList definitions from ODM into define.xml. This will especially be important for so-called "sponsor-defined codelists".

Other properties that you can add or change are the data type, length and significant digits (when appropriate, depending on the data type), role and role codelist.

What one will often want to do, is to subset a codelist, especially when the codelist is very large. This is very often e.g. the case for LBTESTCD and LBTEST, as these have over 2000 entries, and one does not want all of these to appear this in the browser (HTML) view when the regulatory reviewer inspects your define.xml in the browser.
The tutorial "Subsetting and Extending Codelists" explains this very well. SDTM-ETL also has features for automated subset generation and assigning these subset codelists to the corresponding variables. See the separate tutorial "Subsetting and Extending CodeLists" for more details.

One can also add a comment, a display format and/or a reference to a computation method (in the define.xml this is the "MethodDef"). It is however strongly discouraged to change the latter, as it contains the mapping script!

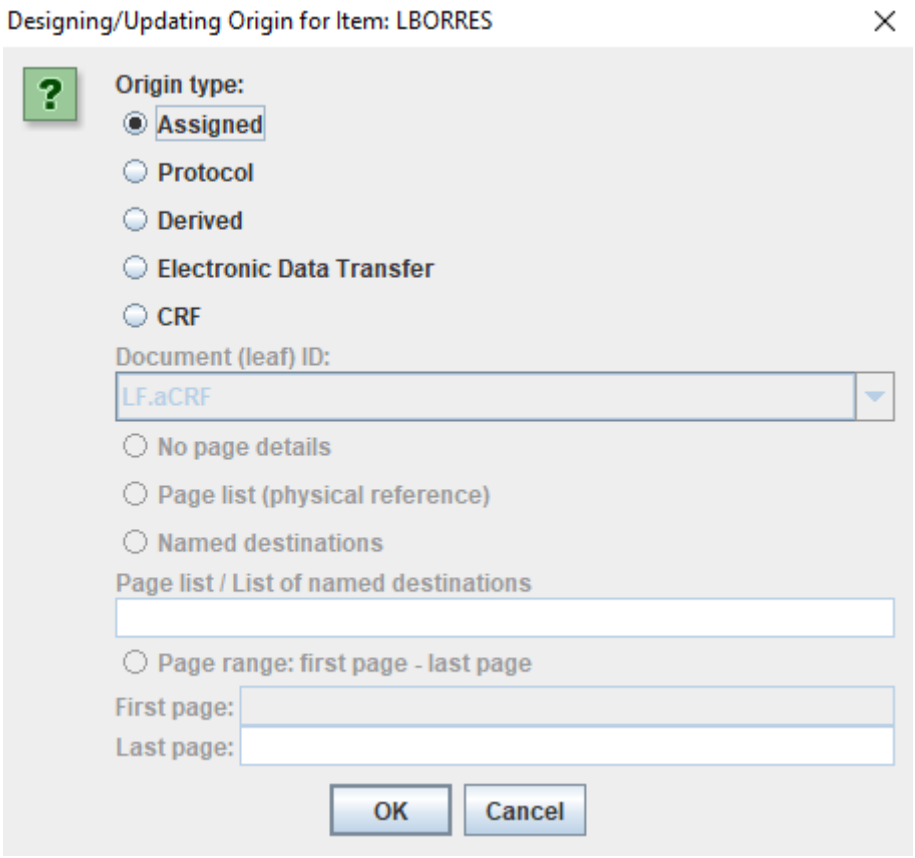
One can also associate a "ValueList", which usually only makes sense when the variable is an --ORRES, --ORRESU, --STRES, --STRESU or a --QVAL variable (see the define.xml 2.0 specification section 4.4 and/or the define.xml 2.1 specification section 4.5).

Generation of ValueLists is explained in the tutorial "Working with ValueLists and the WhereClause in the define.xml".

Setting and editing the "Origin" of the data point: define.xml 2.0

A major difference between define.xml 2.0 and 2.1 is how "Origin" is treated.
Essentially, one should assign an "Origin" to each variable definition, either directly, or through the "ValueList" variable definitions.

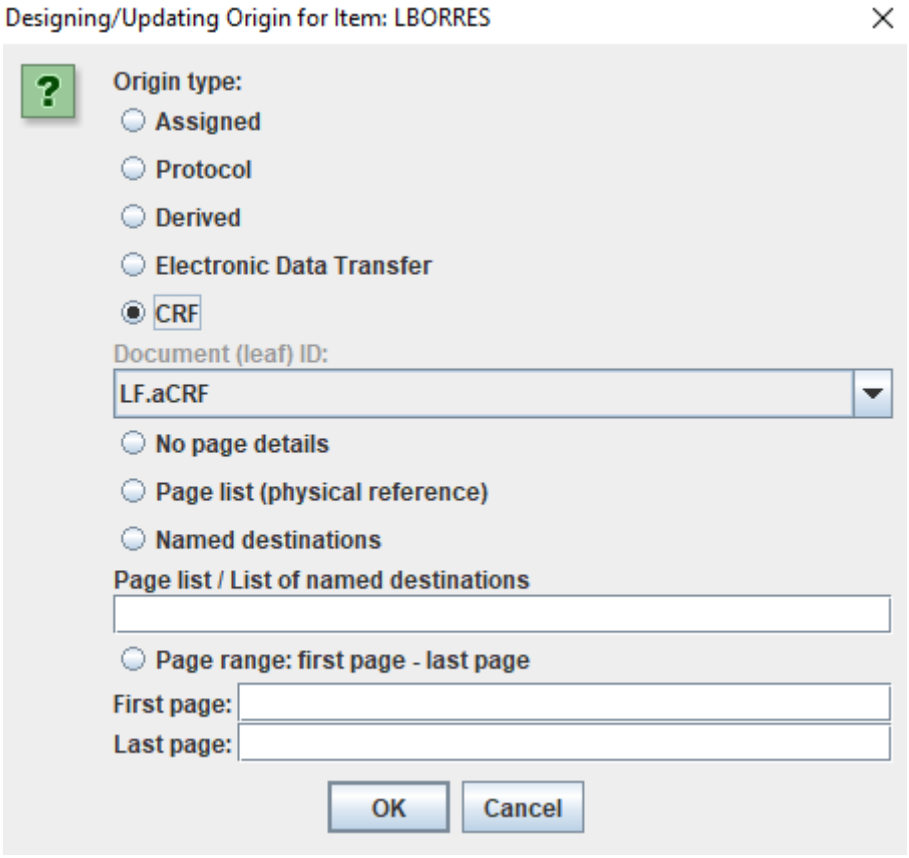
In define.xml 2.0, when checking the checkbox "Edit Origin", following by using the "Edit" button, the following dialog is displayed:



The "origin" can either be "Assigned", "Protocol", "Derived", "Electronic Data Transfer", or "CRF" (Case Report Form). The exact use cases are very well explained in the Define-XML 2.0 specification section 5.3.11.3.
A typical example of "Assigned" is for --TESTCD and --TEST variables and of course for DOMAIN. For "Protocol", a typical example is the STUDYID. "Derived" is for any variable where a calculation of derivation was needed to obtain it, for example USUBJID when it is a concatenation of the STUDYID with the subject ID from the CRF. "Electronic Data Transfer" is often used for any data that came in electronically that is not CRF, and "CRF" is ... well ... data

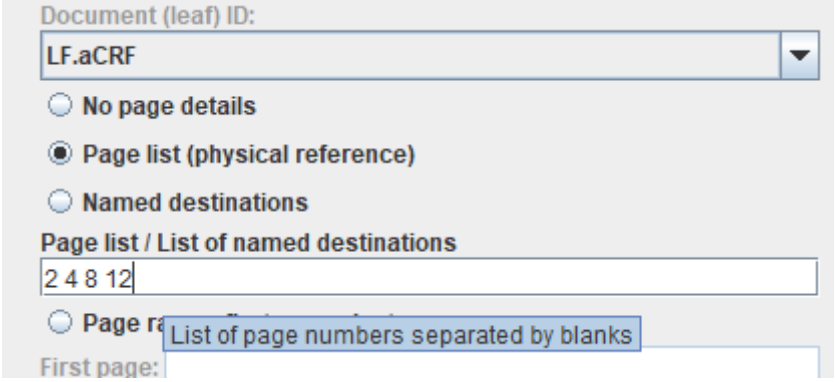
coming from the CRF...
In the latter case, there is a requirement that one provides the page numbers, or a "named destination" (essentially a bookmark), of the PDF file that contains the "annotated CRF". The latter is a document that looks very much like paper CRFs, but is annotated with SDTM information. Such a "paper-like" "annotated CRF" must also be provided when all CRFs were purely electronically! Essentially, also the ODM with the study definition could provide that, but the regulatory authorities are not that far yet, they still live in the age of "electronic paper".

When "CRF" is clicked, some new widgets and fields become available:

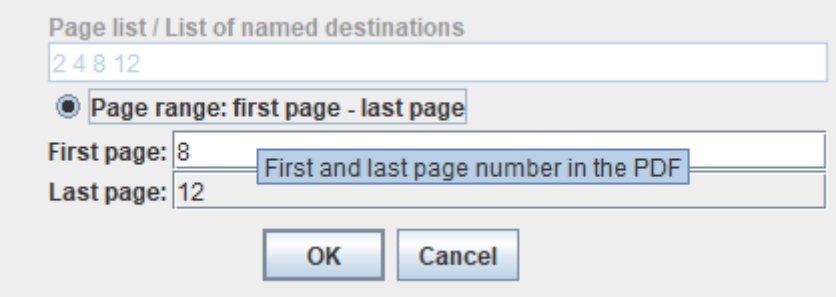


If you have not provided a link to the location of the CRF, there will be no entry like "LF.aCRF" and the system will prompt you to provide that first.

You can now choose between providing no page details (though the reviewers won't like that), a list of pages, a named destination, or a first and a last page (page range).
If one chooses for "page list", this must be a list of page numbers, separated by blanks, e.g.:



When choosing for "page" range, one can e.g. have:

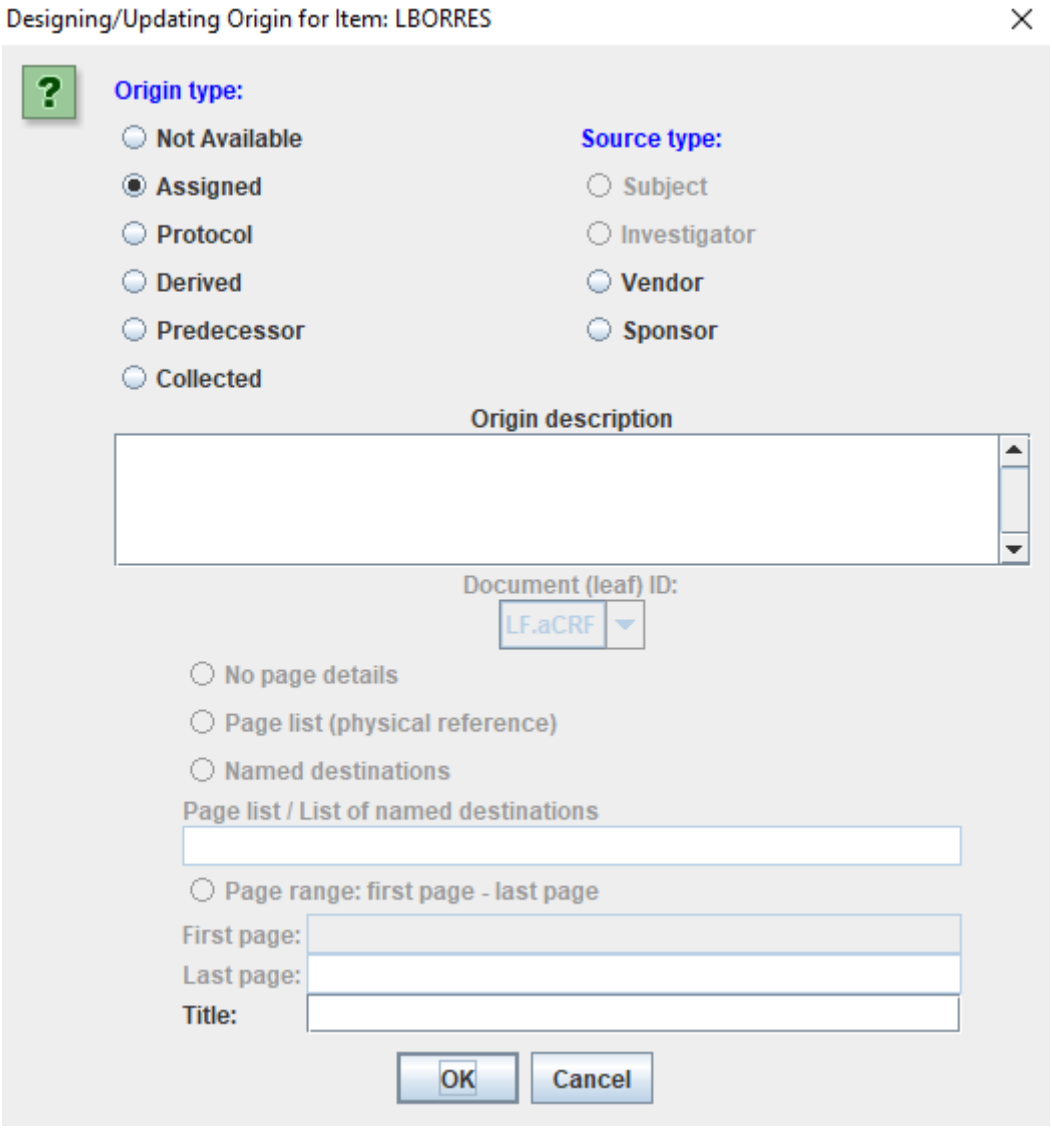


Remark that when you have SEND, there will be other values for "Origin", as e.g. in SEND one does not have CRFs.

Setting and editing the "Origin" of the data point: define.xml 2.1

In Define-XML v.2.1, there is a bit more, and it is a bit more complicated. Not only were the choices for SDTM and SEND aligned, but also a new attribute "Source" has been added.

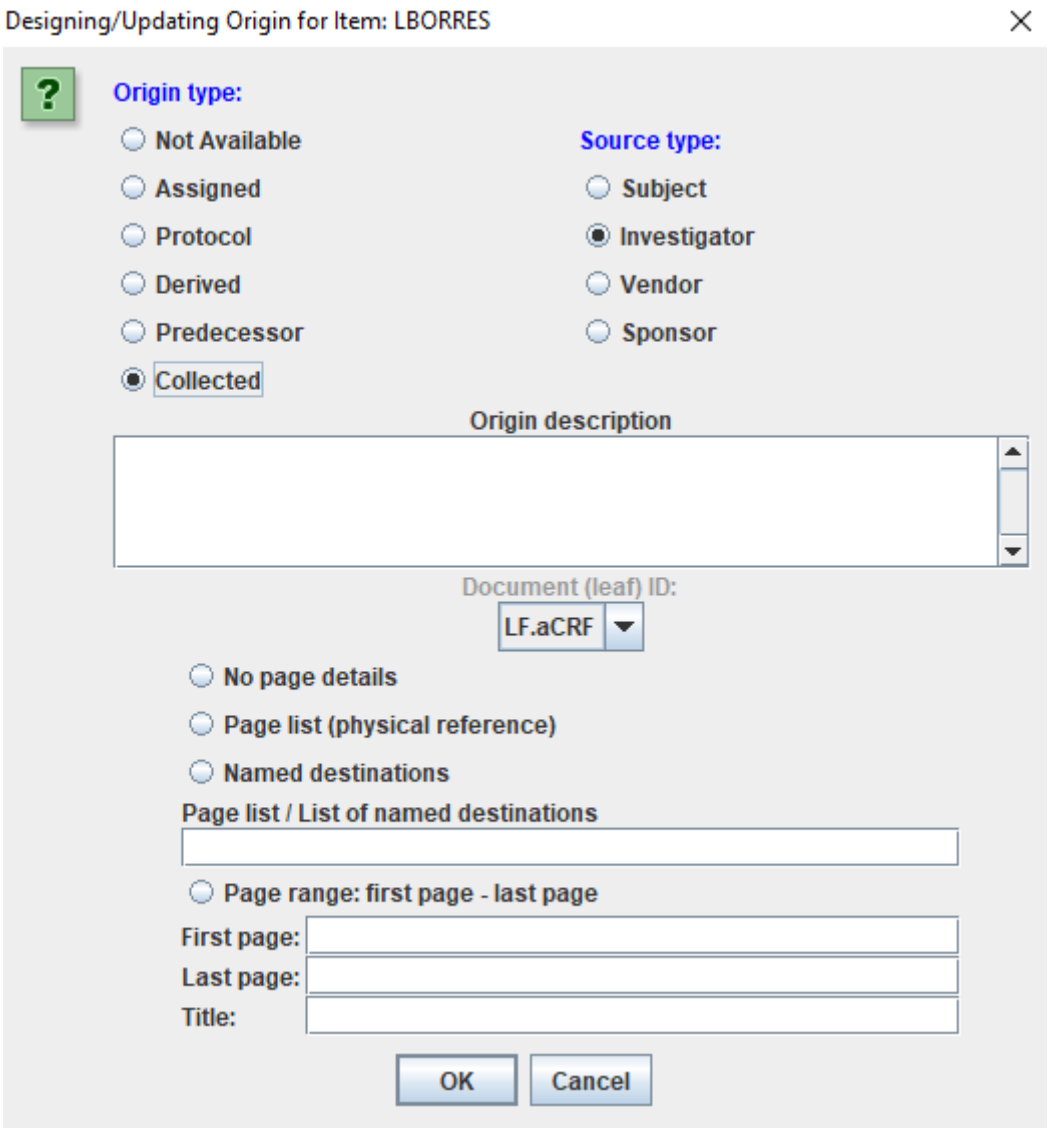
So, when assigning an "Origin" for Define-XML 2.1, the following dialog will be displayed:



"Origin Type" is similar to "Origin" in Define-XML 2.0, but has now been aligned for SDTM and SEND. "Predecessor" is for the case that a variable value has been copied from another variable, from the same domain or from another. This could e.g. be the case for -STRESU variables when the "original" units and the "standard units" are identical.
"Collected" includes data from a CRF, so if we check it, other fields will become available, but need not always be populated, such as in the case of SEND.

New is the "Source Type". The allowed values depend on the value of "Origin Type". The logic for this has been implemented in the software, so no fear that you can do it wrong.

When "Collected" is clicked, the following fields become available:



Essentially enabling to add the same information as for "CRF" in Define-XML 2.0, but now also allowing to add an "Origin description", and a title for the page numbers, page range, or "named destination" of the "annotated" CRF.

When "OK" is clicked, and some required information is missing, a descriptive message will appear, and the dialog will be shown again.