SDTM-ETL 4.x User Manual and Tutorial

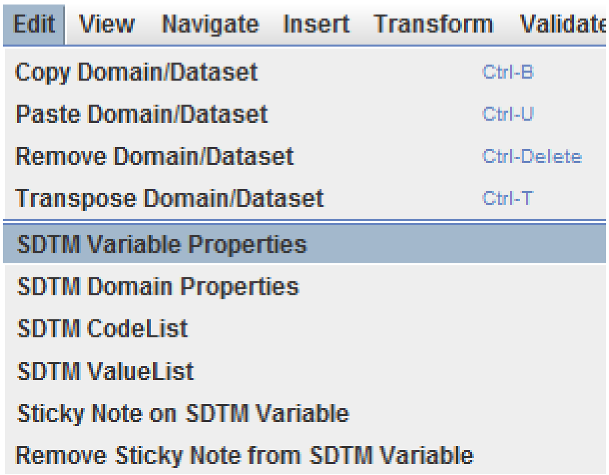
Author: Jozef Aerts, XML4Pharma

Last update: 2022-06-20

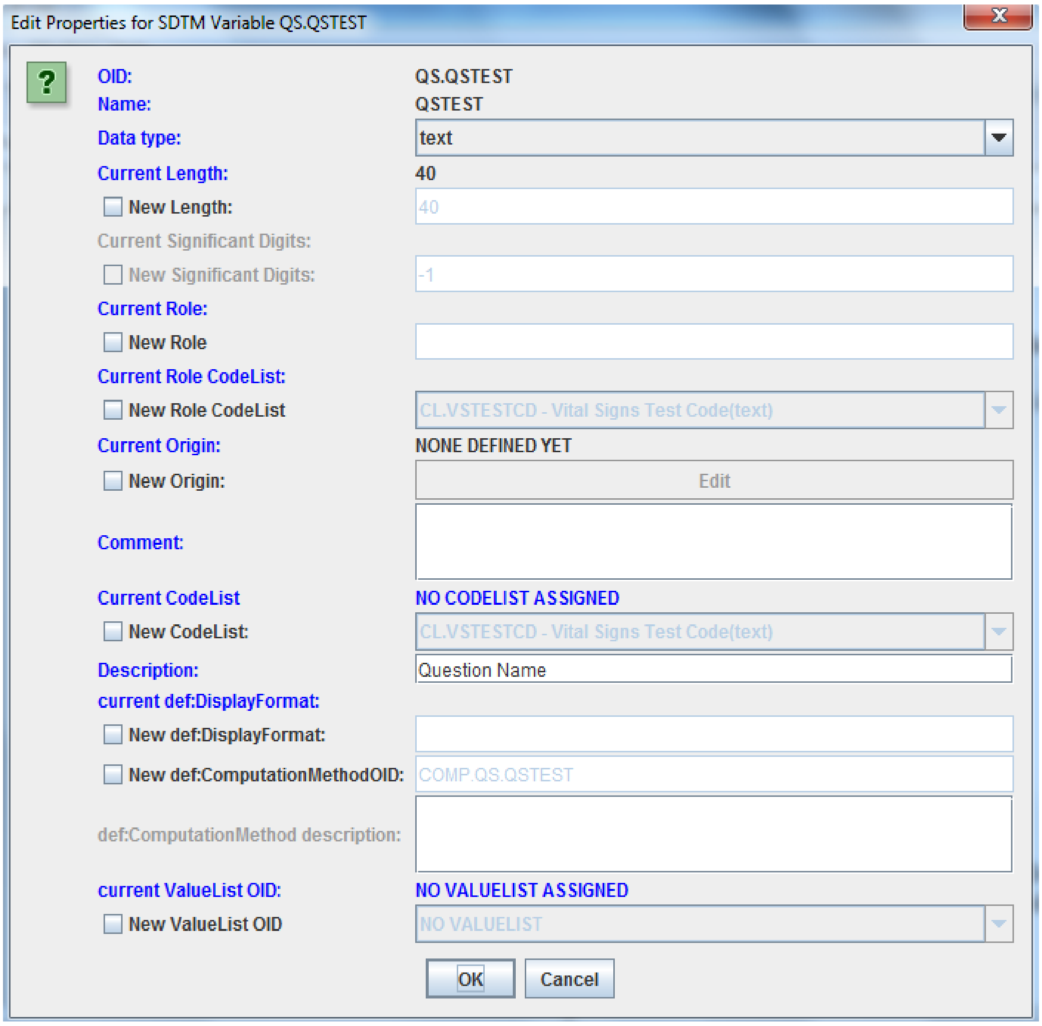
Editing the define.xml ItemDef properties (SDTM Variable properties)

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.

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".


Edit
View
Navigate
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
SDTM Domain Properties
SDTM CodeList
SDTM ValueList
Sticky Note on SDTM Variable
Remove Sticky Note from SDTM Variable

A 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:



Edit Properties for SDTM Variable QS.QSTEST

OID: QS.QSTEST

Name: QSTEST

Data type: text

Current Length: 40

□ New Length: 40

Current Significant Digits:

□ New Significant Digits: -1

Current Role:

□ New Role

Current Role CodeList:

□ New Role CodeList: CL.VSTESTCD - Vital Signs Test Code(text)

Current Origin: NONE DEFINED YET

□ New Origin: Edit

Comment:

Current CodeList: NO CODELIST ASSIGNED

□ New CodeList: CL.VSTESTCD - Vital Signs Test Code(text)

Description: Question Name

current def:DisplayFormat:

□ New def:DisplayFormat:

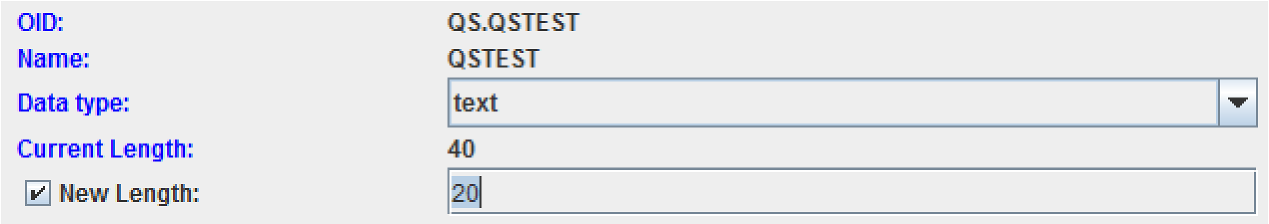
□ New def:ComputationMethodOID: COMP.QS.QSTEST

def:ComputationMethod description:

current ValueList OID: NO VALUELIST ASSIGNED

□ New ValueList OID: NO VALUELIST

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



OID: QS.QSTEST

Name: QSTEST

Data type: text

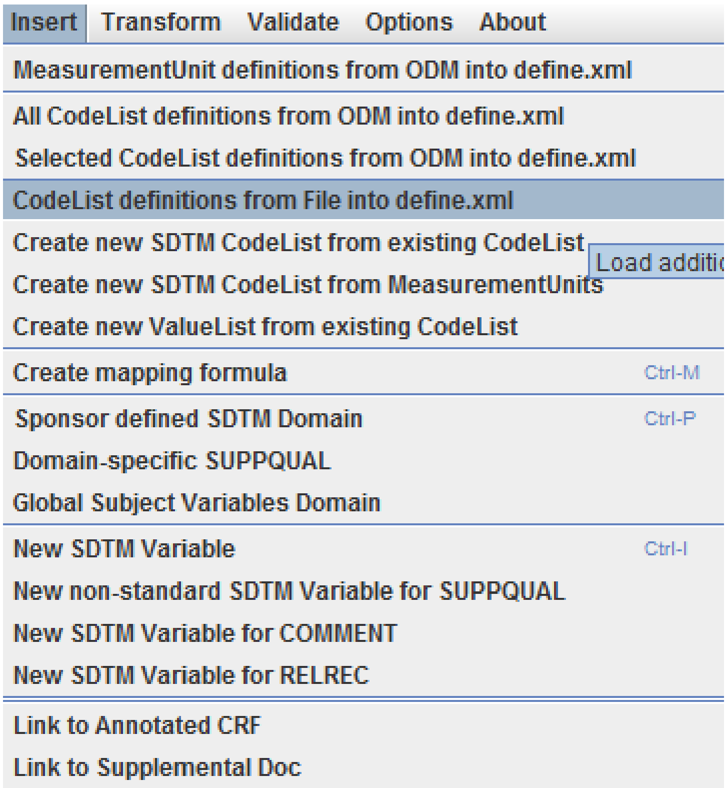
Current Length: 40

☑ New Length: 20

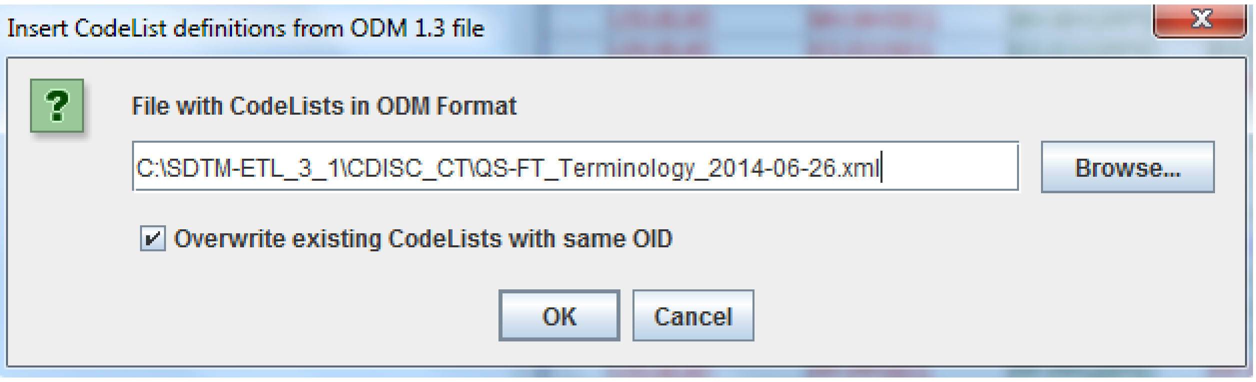
As one sees, one can also change the associated codelist (if any), and some other "ItemDef" properties in the define.xml structure.

Assigning or changing the codelist

For example, if we know that QSTEST follows the latest CDISC controlled terminology, we first need to insert the QS-codelists. In order to do so, first click "OK" and then use the menu "Insert – CodeList definitions from File into define.xml":


Insert
Transform
Validate
Options
About
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
Create new SDTM CodeList from existing CodeList
Load additi
Create new SDTM CodeList from MeasurementUnits
Create new ValueList from existing CodeList
Create mapping formula Ctrl-M
Sponsor defined SDTM Domain Ctrl-P
Domain-specific SUPPQUAL
Global Subject Variables Domain
New SDTM Variable Ctrl-I
New non-standard SDTM Variable for SUPPQUAL
New SDTM Variable for COMMENT
New SDTM Variable for RELREC
Link to Annotated CRF
Link to Supplemental Doc

and e.g. choose the file "QS-FT_Terminology_2014-06-26.xml" (which you can find in the directory „CDISC_CT"):



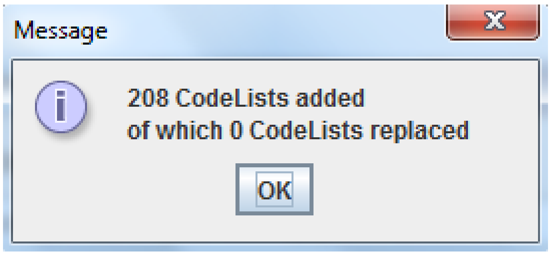
Insert CodeList definitions from ODM 1.3 file

File with CodeLists in ODM Format

C:\SDTM-ETL_3_1\CDISC_CT\QS-FT_Terminology_2014-06-26.xml

☑ Overwrite existing CodeLists with same OID

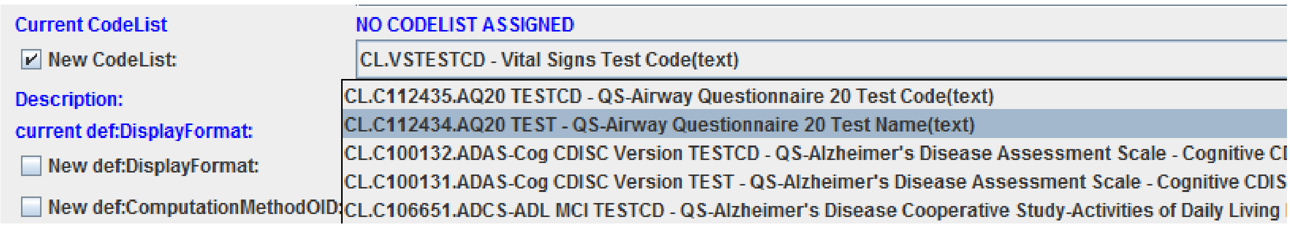
A confirmation message is displayed:



Message

208 CodeLists added
of which 0 CodeLists replaced

If you know edit the properties for "QSTEST" again, you can associate the corresponding codelist, e.g.:



Current CodeList NO CODELIST ASSIGNED

☑ New CodeList: CL.VSTESTCD - Vital Signs Test Code(text)

Description:

current def:DisplayFormat:

☐ New def:DisplayFormat:

☐ New def:ComputationMethodOID:

CL.C112435.AQ20 TESTCD - QS-Airway Questionnaire 20 Test Code(text)
CL.C112434.AQ20 TEST - QS-Airway Questionnaire 20 Test Name(text)
CL.C100132.ADAS-Cog CDISC Version TESTCD - QS-Alzheimer's Disease Assessment Scale - Cognitive Cl
CL.C100131.ADAS-Cog CDISC Version TEST - QS-Alzheimer's Disease Assessment Scale - Cognitive CDIS
CL.C106651.ADCS-ADL MCI TESTCD - QS-Alzheimer's Disease Cooperative Study-Activities of Daily Living

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 inspect your define.xml in the browser.
The tutorial "Subsetting and Extending Codelists" explains this very well. SDTM-ETL 4.1 also has new features for automated subset generation and assigning these subset codelists to the corresponding variables. See the "SDTM-ETL 4.1 New Features" document for more detail.

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!

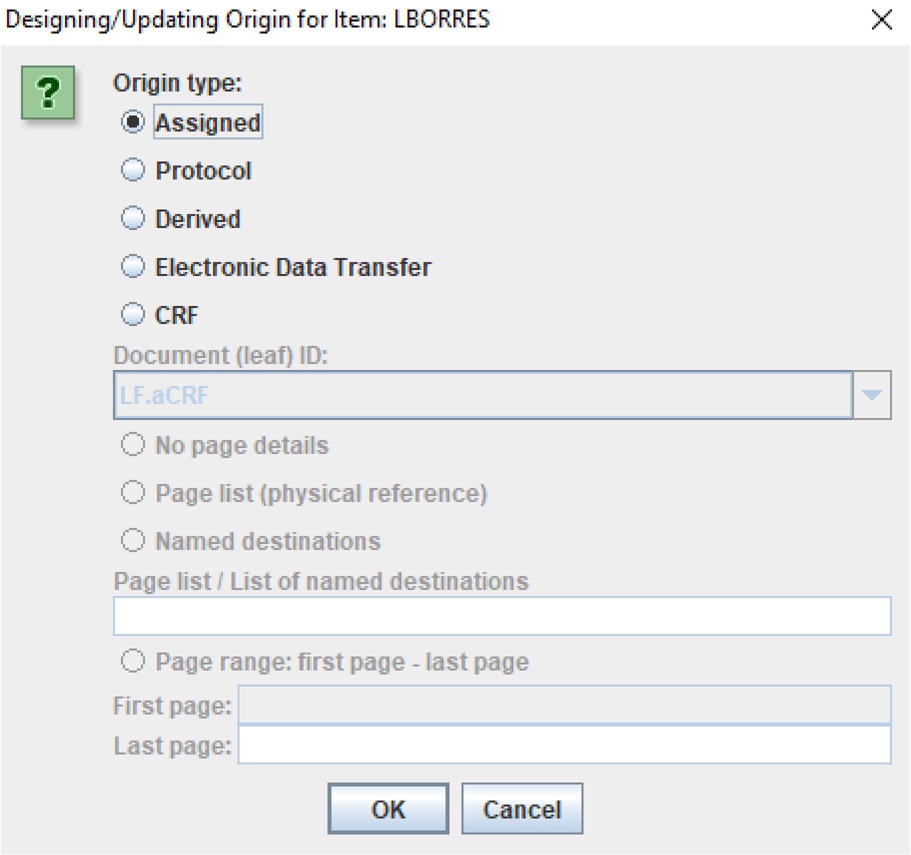
One can also associate a "ValueList", which usually only makes sense when the variable is an --ORRES, --ORRESU, --STRES, --STRESU or a --QVAL variable (see the define.xml 2.0 specification section 4.4 and/or the define.xml 2.1 specification section 4.5).

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:



Designing/Updating Origin for Item: LBORRES

Origin type:

● Assigned

○ Protocol

○ Derived

○ Electronic Data Transfer

○ CRF

Document (leaf) ID: LF.aCRF

○ No page details

○ Page list (physical reference)

○ Named destinations

Page list / List of named destinations

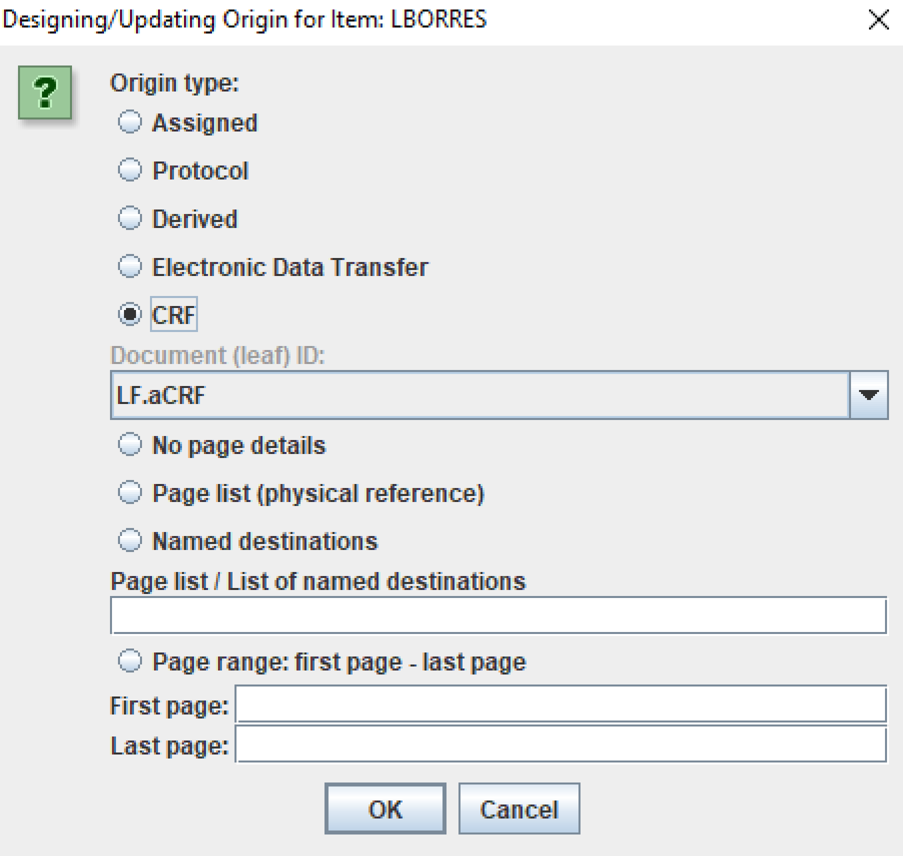
○ Page range: first page - last page

First page:

Last page:

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 get 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:



Designing/Updating Origin for Item: LBORRES

Origin type:

○ Assigned

○ Protocol

○ Derived

○ Electronic Data Transfer

● CRF

Document (leaf) ID: LF.aCRF

○ No page details

○ Page list (physical reference)

○ Named destinations

Page list / List of named destinations:

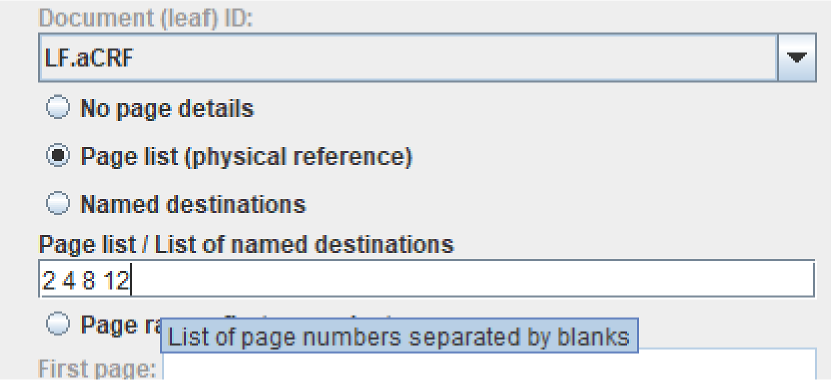
○ Page range: first page - last page

First page:

Last page:

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.:



Document (leaf) ID: LF.aCRF

○ No page details

● Page list (physical reference)

○ Named destinations

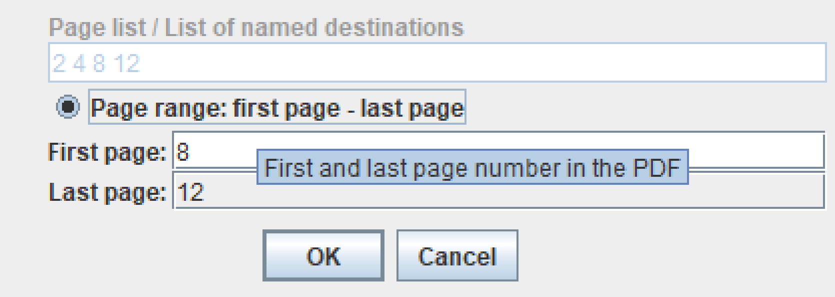
Page list / List of named destinations: 2 4 8 12

○ Page ra

List of page numbers separated by blanks

First page:

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



Page list / List of named destinations: 2 4 8 12

● Page range: first page - last page

First page: 8

First and last page number in the PDF

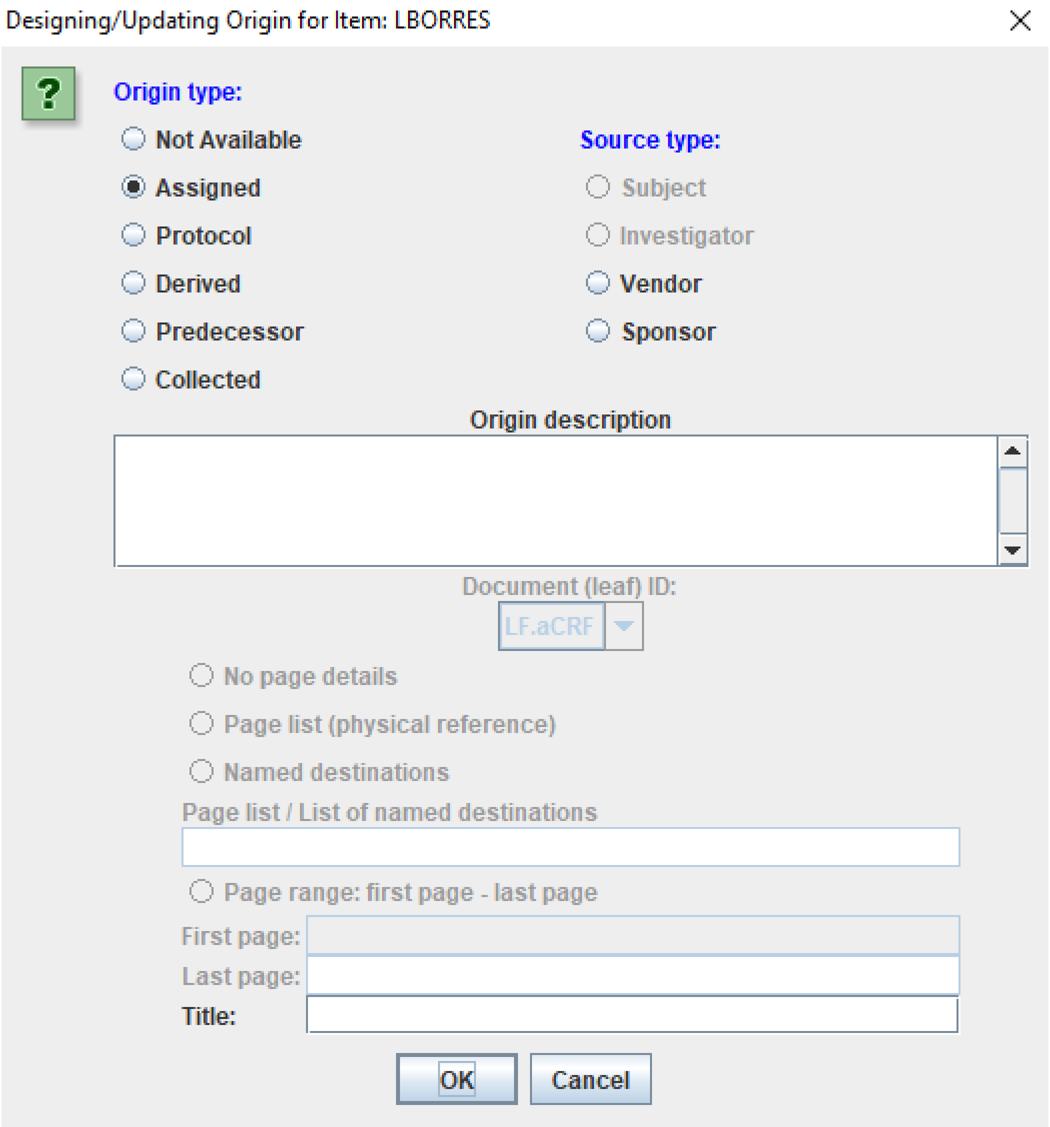
Last page: 12

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:



Designing/Updating Origin for Item: LBORRES

Origin type:

○ Not Available
● Assigned
○ Protocol
○ Derived
○ Predecessor
○ Collected

Source type:

○ Subject
○ Investigator
○ Vendor
○ Sponsor

Origin description

Document (leaf) ID: LF.aCRF

○ No page details
○ Page list (physical reference)
○ Named destinations

Page list / List of named destinations

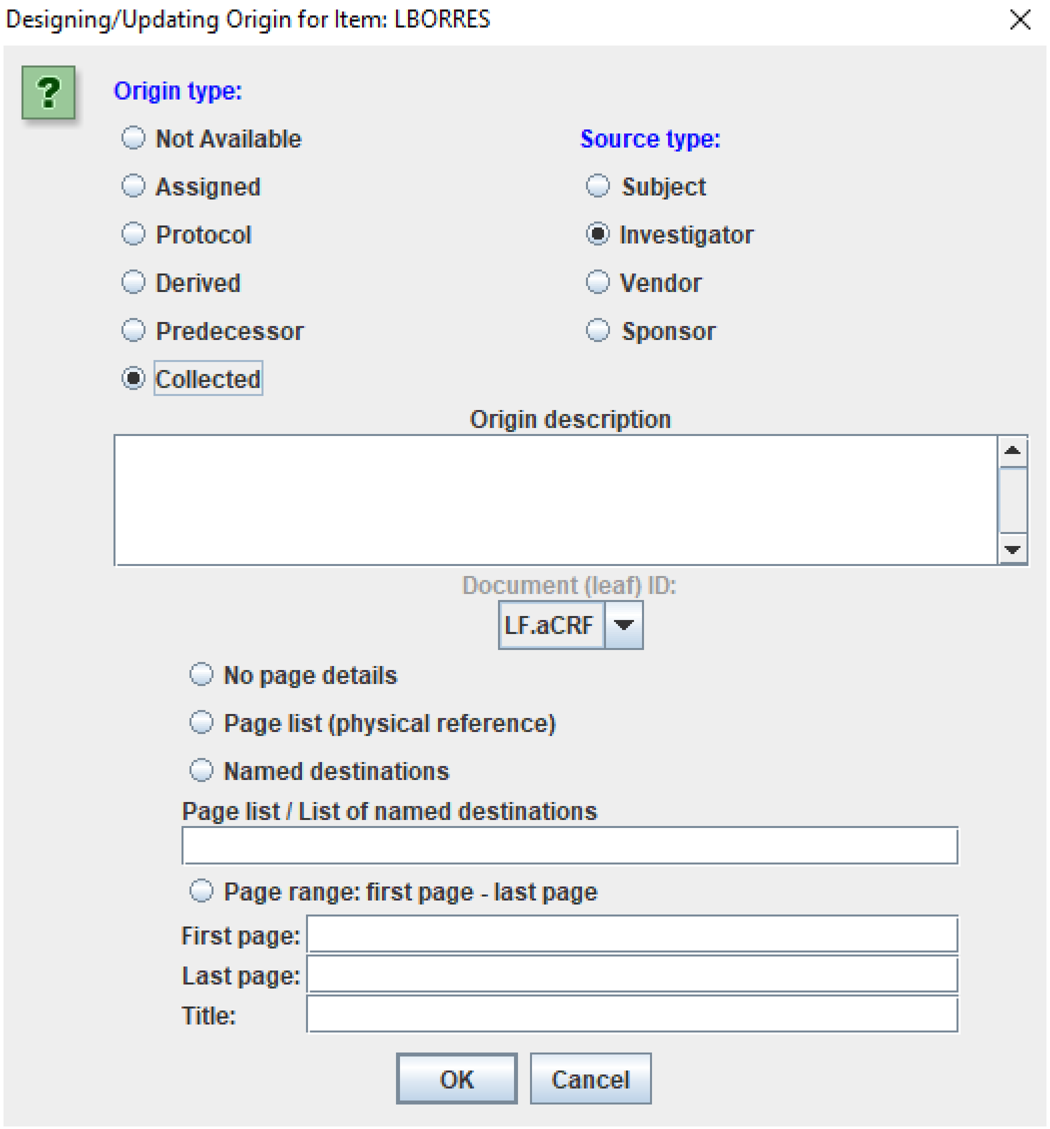
○ Page range: first page - last page

First page:
Last page:
Title:

"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:



Origin type:

○ Not Available
○ Assigned
○ Protocol
○ Derived
○ Predecessor
● Collected

Source type:

○ Subject
● Investigator
○ Vendor
○ Sponsor

Origin description

Document (leaf) ID: LF.aCRF

○ No page details
○ Page list (physical reference)
○ Named destinations

Page list / List of named destinations

○ Page range: first page - last page

First page:
Last page:
Title:

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 information is missing, a descriptive message will appear, and the dialog will be shown again.