SDTM-ETL 4.4 User Manual and Tutorial

Author: Jozef Aerts, XML4Pharma

Last update: 2023-11-28

Automated generation of -LOBXFL flags: extended features

Table of Contents

Introduction

Since SDTMIG-3.3, a new variable has been added to all Findings domains on request of the FDA, the "--LOBXFL" "Last Observation Before Exposure Flag". The value of the --LOBXFL flag can only be "Y" or null (i.e. empty). The description by the SDTMIG is "*Operationally derived indicator used to identify the last non-missing value prior to RFXSTDTC*".

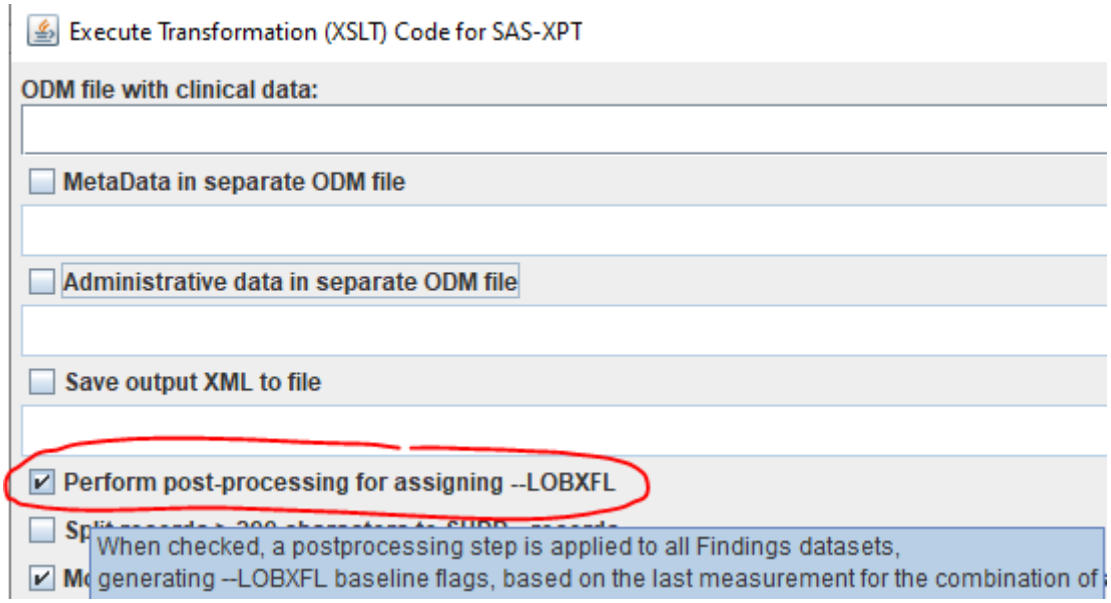
Essentially, FDA wanted this additional variable, as the tools of the reviewers seem to be unable to calculate which is the "last non-missing value" themselves ...
The variable is meant to serve as a "baseline flag", although that should not be in SDTM (as SDTM is about "categorized source data").

What "last non-missing value" exactly means, and on what it should be based on is however not explained by the SDTMIG, nor in the SDTM model itself.
It is however pretty obvious that the assignment should be done per individual test per subject, but what is the definition of "individual test"?

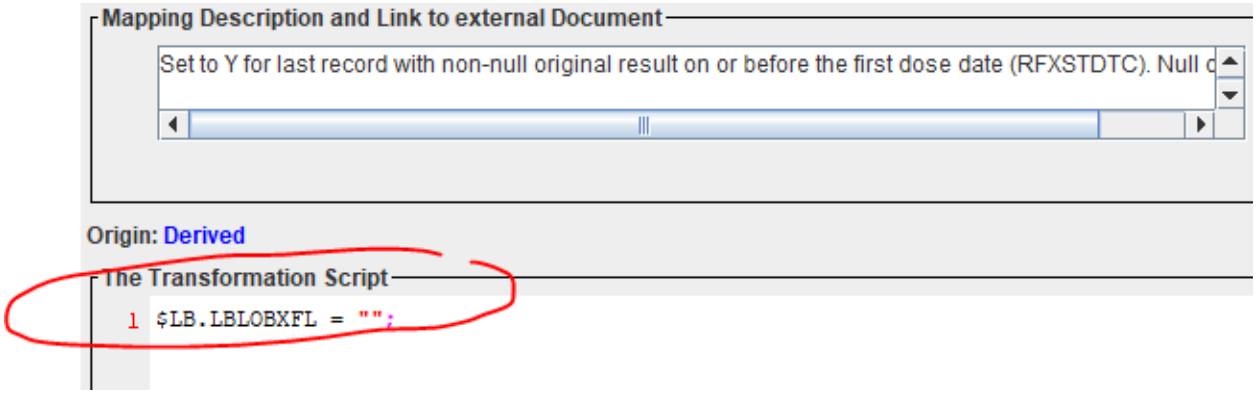
Most people think that "individual test" is defined by --TESTCD, and this is also the way it has been implemented so far in SDTM-ETL, but this is not always true.
Think about the situation that albumin is both measured in blood as in urine. For both the tests, LBTESTCD=ALB (LBTEST=Albumin), so does this mean that there will only be one record per subject for albumin that can be flagged as "last non-missing value"? If so, which one then? The one for in urine? Or the blood one?

New features in SDTM-ETL 4.4

In SDTM-ETL, one can automate the generation of the assignment of --LOBXFL flags fully automatically, in a "post-processing" step (that is what the SDTMIG means with "operationally derived"), without any additional programming. In order to do so, one should check the checkbox "Perform post-processing for assigning --LOBXFL".



The only requirements for making this possible, is that there is a "placeholder mapping" for the --LOBXFL variables, e.g. for LBLOBXFL:



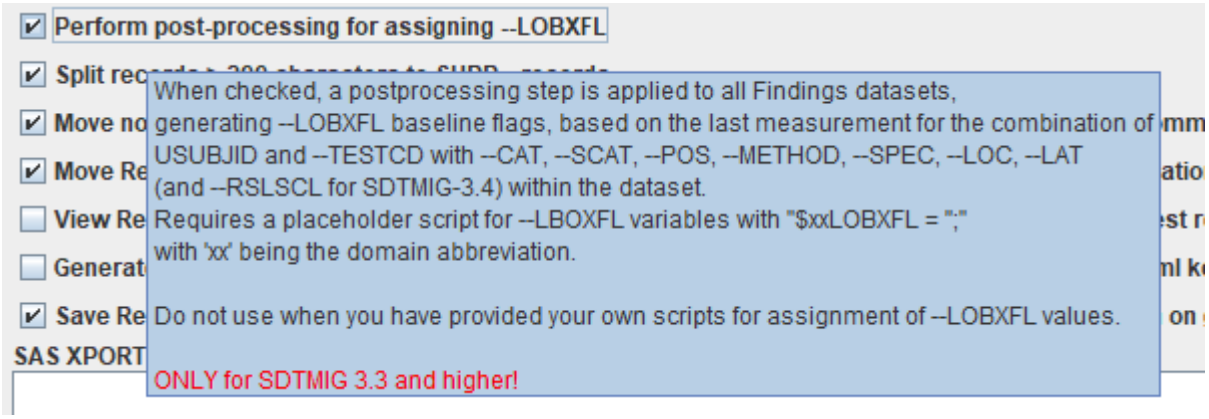
and that also the mapping for the DM (Demographics), with variable RFXSTDTC (Date/Time of First Study Treatment) is loaded.

So, if you find that the checkbox "Perform post-processing for assigning --LOBXFL" is disabled, first check whether the --LOBXFL variable is present, and that it contains the "placeholder mapping", and that also DM with RFXSTDTC is loaded.
Also important: the --LOBXFL variable is only known in SDTM as of SDTMIG v.3.3. So, if you are still working with the outdated version 3.2, you will not see this variable, and you will not be able to use this mechanism.

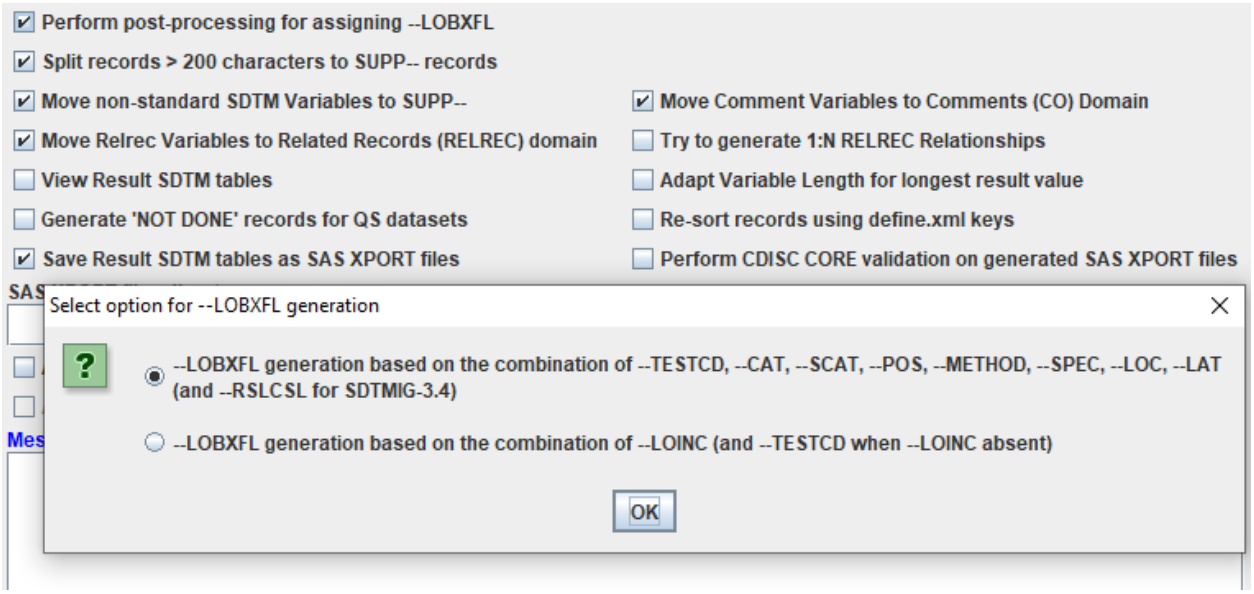
Before SDTM-ETL v.4.4, the assignment of values for --LOBXFL was always based on the values of the combination of USUBJID and --TESTCD, and this is also a valid assumption for about 90% of the cases. However, if the value of --TESTCD does not uniquely define the test, one should do more.
Essentially, the best identifier for unique tests is the LOINC code, which goes into the --LOINC variable. This is not only true for lab values, but also for microbiology tests, vital signs, ECG tests and even questionnaires - all of these are covered by LOINC.
However, LOINC is still too much "not invented here" at CDISC, and mostly ignored if it were not

that the FDA requires it for the LB dataset[1]. CDISC has been able to partially "circumvent" this requirement by moving more and more tests outside the LB domain, such as MB (Microbiology), which are essentially also lab tests, and recently the IS (Immunogenicity Specimen Assessments) domain.

As of SDTM-ETL 4.4, the user has the choice between two methods to define "test uniqueness, which can already be seen by hovering the mouse over the checkbox:



and when one checks the checkbox, the following dialog appears:



One can choose between either using a combination of the values of
a) --TESTCD, --CAT (category), --SCAT (subcategory), --METHOD, --SPEC (specimen type), --LOC (location, i.e. body part), and and --LAT (latitude, e.g. "left", "right", ...). In case of SDTMIG-3.4, --RSLSCL (result scale, e.g. quantitative, ordinal, ...) is added. All these, except for --TESTCD are optional, meaning that if e.g. "METHOD" is not present or not populated, it is not taken into account for the uniqueness.
b) --LOINC (the LOINC code), and when absent, --TESTCD only is taken.

One may ask: "why not let the user decide to select the variables for defining the unique tests"? We have long considered this, but we fear that this would overstrain most of the users. This especially as the choice is different from the "uniqueness keys" for the records themselves, in which also very

[1] See the FDA "Study Data Technical Conformance Guide", Section 6.7.1:
https://www.fda.gov/media/153632/download

often a timing component (e.g. timepoint or timepoint number is involved).

For our example of albumin measured in both urine and blood, where the test uniqueness keys reduce to --TESTCD and --SPEC (when --CAT and --SCAT are not provided), this would mean that each subject will have up to 2 records with LBLOBXFL=Y, one for albumin measured in blood, and one for albumin measured in urine.

Sometimes there may even be more records with LBLOBXFL=Y per subject. Think about glucose measurements, not only in different specimens, but also measured in different ways, like "quantitative", "ordinal", "presence", ...

Validation

IMPORTANT: when one has more test with the same value of LBTESTCD (as in our Albumin example), one may expect that Pinnacle21 will cause some false-positive errors "Multiple xxLOBXFL records for the same test", as it bases the "test uniqueness" solely on LBTESTCD, LBCAT and LBSCAT. As a user, you cannot say to the Pinnacle21 software on which variable(s) you based your "test uniqueness"!
Of course, one can "avoid" these false positives by assigning "intelligent" values to --CAT and --SCAT, but essentially, one should not manipulate it's submission data just to avoid getting false positives from Pinnacle21 validation.