Remapping of codes (and decodes) in analysis datasets for electronic submissions

Joerg Guettner

Bayer Pharma AG, Wuppertal, Germany

For electronic submissions, the U.S. Food and Drug Administration requests, according to the Study Data Specifications (version 1.6, June 2011), that variable names and codes in analysis datasets should be consistent across studies and, where feasible, the NCI CDISC Vocabulary should be used. For integrated analysis, it is necessary to harmonize codes across all studies. Unfortunately, during the life cycle of a product, codes may change either due to project needs or due to external circumstances, e.g. CDISC introduces new controlled terminology. For analysis datasets following the Analysis Data Model, there are often pairs of corresponding variables containing a code and a decode which both have to be updated. This paper describes a workflow to remap codes and also decodes for a study or an integrated database with a macro using metadata and standard SAS format datasets with a few additional variables.

Keywords: Code remapping, Analysis dataset, Integrated analysis, FDA, Electronic submission

Introduction

As codes may change during the life cycle of a project, there are two main reasons that could make a remapping of codes necessary:

- the Food and Drug Administration requests for electronic submissions, that variable names and codes in analysis datasets should be consistent across studies and where feasible, the NCI CDISC Vocabulary should be used;[1]
- for integrated analyses, it is necessary to harmonize codes across studies to assure a consistent approach for analyses.

A prominent example is the laboratory tests. In the first release of CDISC controlled terminology, less than 100 laboratory test terms were available. Meanwhile, there are over 700 terms for laboratory tests. In case that there was no CDISC controlled term available at the time of analysis, the sponsor extended the codelist by adding this term. At the time of submission, this term has been added to the controlled terminology, of course with a different CDISC submission value. Newer studies may have already used this new CDISC controlled term. As mentioned above, the Food and Drug Administration requests use of controlled terminology. Therefore, it is necessary to remap the sponsor's term to the controlled term for the electronic submission.

In the Study Data Tabulation Model datasets, controlled terminology and sponsor-defined terminology are usually captured only in character variables, i.e. without a numerical representation. There are just a few exceptions like the variable pairs xxTEST and xxTESTCD (e.g. lab test or examination name) or VISIT and VISITNUM (visit name and number).

In contrast to Study Data Tabulation Model, analysis datasets following the Analysis Data Model often have pairs of corresponding variables containing a decode and a code (character and numerical representation), e.g. AVISIT and AVISITN (analysis visit) or flags like ABLFL and ABLFN (baseline record flag). In the case of a necessary code remapping, both code and decode have to be updated.

If a table with the remapping information is available, the remapping can be easily performed for the codes. The corresponding codelist should be available, either directly applied to the data or found in the metadata. For decodes this is a bit more tricky. To make sure that code and decode match after a remapping, it is necessary to identify the code in the corresponding variable. Due to the limitation of eight characters for variable names, it might be even difficult to identify the corresponding variable automatically, e.g. at Bayer we have pairs of variables like LBMETHOD and LBMTHODN (method of test or examination). As LBMETHOD is already eight characters long, it has to be shortened somehow to add the 'N' for the numeric variable. This was carried out by omitting the 'E' in 'LBMETHOD' in this case.

For the suggested remapping workflow, the following information has to be available:

Correspondence to: J Guetter, Bayer Pharma AG, Aprather Weg 18a, D-42113 Wuppertal, Germany. Email: joerg.guettner@bayer.com

Published by Maney
DOI 10.1179/175709311X13166801334118

- the remapping information;
- the codelist of a variable;
- which variables represent a pair of corresponding variables, containing a decode and a code.

Furthermore, it is assumed that codelists and metadata are available as SAS datasets. Ideally, there should be one or more repositories with all codes and decodes in the form of standard SAS format datasets, so the remapping information can be stored by adding a variable to the codelists. The format of a variable should be part of the metadata, and the information on which variables are a pair of corresponding variables, containing a decode and a code, should go into the metadata.

Based on these codelists and metadata, it is possible to implement a workflow that automatically updates codes and the according decodes.

Codelists — the Place to Store the Remapping Information

At Bayer several repositories for SAS format datasets are used:

- Global Medical Standards;
- Therapeutic Area Standards;
- Project Standards (project means compound level in this case);
- Analysis Datasets.

The advantage of these repositories is that all studies share the same codelists and can be decoded with exactly the same codelist. An important restriction on these repositories is that codes are not allowed to be deleted once they have been assigned. Otherwise, it would not be possible to decode studies using this code. It is possible to change the decode, e.g. to correct typing errors, but modifying the decode to give a different meaning to the code at any time after it has been initialized is not allowed, i.e. if a code 'COLD' was assigned to a codelist with decode 'Commn Cold', it would be possible to correct the typing error. But changing the decode to 'Chronic Obstructive Lung Disease' would not be permitted, as this would change the content. On the other hand, this restriction makes it possible to use these codelists for storing the remapping information, as obsolete codes are still available in the codelists.

To distinguish between codes that should be used and codes that have been retired and for traceability, it was necessary to add a few administrative variables to the SAS format datasets. The most important ones are:

- STATUS: A — active, R — retired;
- REASON: short description for changes on the record (not only necessarily status changes);
- SYSDATE: date and time of last change of the record.

To capture the remapping information, another administrative variable named UPMAP has been added to the SAS format datasets. As an example, an extract of the codelist LBTEST is displayed in Table 1. Each time it is decided that a code has to be remapped to another code, either due to project needs or new controlled terminology becoming available, the new code has to be added to the corresponding record in the remapping of the SAS format dataset, and the old code has to be retired.

For example, in Table 1, a new code 'ETHANOL' with decode 'Ethanol' was added to the codelist LBTEST, but there is already an old code 'ETHYLALC' with decode 'Ethyl Alcohol', which has an equivalent meaning. As 'ETHANOL' is controlled terminology, the code 'ETHYLALC' needs to be remapped to the new code 'ETHANOL'. Therefore, a record with status A (active) for code 'ETHANOL' was added to the codelist LBTEST. At the same time, the status of code 'ETHYLALC' was changed to R (retired), and the variable UPMAP was set to 'ETHANOL'.

There is another limitation to using this approach to store the remapping information — it is only possible to store a mapping on a one-to-one basis. A retired code can be remapped to exactly one other code, not to more than one. A mapping to more than one code would require some side conditions. A remapping to a different codelist is not possible.

Metadata

The metadata are used for two important things in the workflow:

- to identify the codelists used by a variable;
- to identify the pairs of corresponding variables containing a decode and a code.

Bayer uses a SAS environment whose production area for data is strongly metadata-based, and these metadata are stored as SAS datasets. Strongly metadata-based means that it is not possible to move data to the production area that do not comply with

Table 1 Extract of codelist LBTEST

FMTNAME	START	LABEL	TYPE	UPMAP	STATUS	Reason	Sysdate
LBTEST	ETHANOL	Ethanol	C		A	Creation	28FEB2011:17:21:38
LBTEST	ETHYLALC	Ethyl Alcohol	C	ETHANOL	R	Updated 28 February 2011	28FEB2011:17:21:38
LBTEST	FAC7	Factor VII	C	FACTVII	R	Update request 30 March 2011	30MAR2011:16:11:24
LBTEST	FACTVII	Factor VII	C		A	Creation	20DEC2010:07:32:30
LBTEST	HPOCROM	Hypochromia	C		A	Source: g_codelist_20110325.zip	25MAR2011:12:50:49
LBTEST	HYPO	Hypochromia	C	HPOCROM	R	Update request 30 March 2011	30MAR2011:16:11:24
LBTEST	PROLAC	Prolactin	C	PROLCTN	R	Update request 30 March 2011	30MAR2011:16:11:24
LBTEST	PROLCTN	Prolactin	C		A	Creation	20DEC2010:07:32:30

the metadata, e.g. checks run during the transfer to the production area verify that all codelists used in a dataset exist, and all codes used can be decoded with these codelists. A partial example of the metadata SAS dataset for ADLB is presented in Table 2. In this table, the variable name can be found in the column SASNAME, and its associated codelist in the column CODLST.

This metadata already contained a variable for the codelists. Unfortunately, it was not possible to just add a new metadata variable identifying pairs of corresponding variables due to the fact that the Bayer SAS environment does not allow the extension of metadata datasets without any changes to the underlying system. Due to this restriction, it was decided to use an already existing variable to store the information about variable pairs. A good choice seemed to be the variable COMMENTS as it was not populated very often. To distinguish between normal comments and a corresponding variable name, the variable name of the variable containing the associated code is added in uppercase at the end of the comment surrounded by square brackets, e.g. in Table 2, the values of variable 'LBTEST' are the decodes of the values of variable 'LBTESTCD', as '[LBTESTCD]' is the corresponding value of the column COMMENT.

But why is it necessary to make this extra effort and use the code to remap the decode? Would it not be easier just to look for the new decode based on the old one? In an ideal world, this would be indeed much easier. But unfortunately the world is not that ideal, and from time to time there are cases (usually just small typing errors, like the unit 'DA' misspelled as 'Da') where code and decode do not match 100%. This is due to the fact that the current Bayer environment checks the variable containing the code to make sure that it has no codes that cannot be found in the corresponding codelist. However, for the variable containing the decode, only the length is checked — there is no check on the content. Using the associated code for the remapping decode avoids the possibility that not all decodes are remapped as they should be.

Workflow to Update Codes and Decodes

With the additional information stored in the format and metadata SAS datasets, it is possible to apply the following workflow to remap codes for a study or integrated database.

The workflow can be separated into three steps:

1. search the codelists for codes to be remapped;
2. identify the datasets and variables that use codes to be remapped in the metadata;
3. update the identified variables and datasets.

In the first step, all codelists are checked for codes to be remapped, i.e. that the value of column UPMAP is populated in the corresponding SAS format dataset. In the case of multiple remappings (e.g. A is remapped to B and B is remapped to C), only the latest remapping information should be kept (A is remapped to C in this example). Otherwise, multiple passes would be necessary. The result of this step consists of three pieces of information: the codelists with codes to be remapped, the code to be remapped, and the code to be mapped to.

In the next step, these results are used to identify the datasets with variables using codelists which contain codes to be remapped. The corresponding decodes of these variables are collected.

Table 2 Extract of metadata for analysis dataset ADLB

VARSEQ	SASNAME	LABEL	TYPE	OUTFORM	CODLST	DESCRIPT	COMMENT
3	USUBJID	Unique Subject Identifier	C	20.0			
9	LBSEQ	Sequence Number	N	8		LB.LBSEQ	
11	LBTESTCD	Lab Test of Examination Short Name	C	8	LBTEST	LB.LBTESTCD	
12	LBTEST	Lab Test of Examination Name	C	40		LB.LBTEST	[LBTESTCD]
23	PARAM	Parameter Description	C	200		New code based on combination of LBTEST/LBTESTCD, LBSTRESU, LBSPEC, LBMETHOD and …	[PARAMCD]
24	PARAMCD	Parameter Code	C	8	X_PARAMC	New code based on combination of LBTEST/LBTESTCD, LBSTRESU, LBSPEC, LBMETHOD and …	
26	AVAL	Analysis Value	N	17.8		Derived from LBSTRESN based on rules in SAP	
49	AVISIT	Analysis Visit Description	C	40		Windowed value of VISIT according to rules in SAP	[AVISITN]
50	AVISITN	Analysis Visit Number	N	9.4	Z_AVISIT	Windowed value of VISITNUM according to rules in SAP	
89	LBTPT	Planned Time Point Name	C	50		LB.LBTPT	[LBTPTNUM]
90	LBTPTNUM	Planned Time Point Number	N	8	ZTPT	LB.LBTPTNUM	

In the last step, only those datasets using codelists with codes to be remapped need to be checked for those codes. In these datasets, the values of the identified variables, codes as well as decodes, can be easily updated.

Conclusion

With this workflow, it is possible to remap codes and decodes of a study or integrated database in an automated way. There is one pair of variables in the Analysis Data Model Basic Data Structure that cannot be updated with this workflow in all datasets. This is due to the fact that the Analysis Value (AVAL) and Analysis Value (C) (AVALC) might contain a mixture of character and numeric results in some datasets, and may be even a mixture of values from different codelists, e.g. in the questionnaire analysis dataset ADQS, the results, stored in AVAL and AVALC, of visual analogue scale are numeric, and the answers to questions are character.

Acknowledgements

I would like to thank Nancy Brucken (i3), Tanja Petrowitsch, and Peter Bonata (both Bayer Pharma) for their valuable input and contributions towards this paper.

References

1 Food and Drug Administration. Guidance for industry: study data specifications [document on the Internet]. Silver Spring, MD: Food and Drug Administration; 2011 [cited 2011 Jul 18]. Available from: http://www.fda.gov/Drugs/DevelopmentApprovalProcess/FormsSubmissionRequirements/ElectronicSubmissions/ucm248635.htm.