PhUSE 2016

Paper CD07

OCCDS – Creating flags or records

Rob Wartenhorst, GSK Vaccines, Amsterdam, Netherlands

ABSTRACT

June last year the Occurrence Data Structure guide was released. The guide includes occurrence flagging. These occurrence flags make it really transparent, which record is counted in a table. However these flags have the potential to increase exponentially when flagging several subsets. This paper addresses our approach to implement the OCCDS guide, keeping the number of occurrence flags limited to the standard flags listed in the IG and instead creating records, making programming, and using the data, simple and transparent.

INTRODUCTION

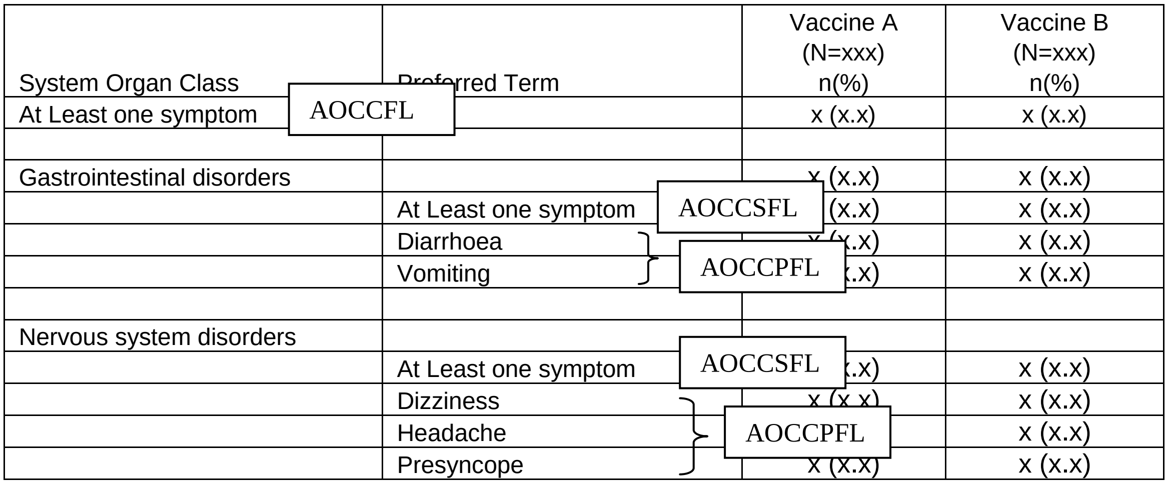
Our adverse events, concomitant and medical history ADaM datasets were set up using the ADAE data structure before the Occurrence Data Structure (OCCDS) guide was released. We enthusiastically started implementing the occurrence flags. But soon the dataset became very wide. The question was how to keep the data manageable and still include the occurrence flags.

To address this question this paper first elaborates more on the problem and then how it can be addressed.

This paper contains the opinions of the author.

OCCDS OCCURRENCE FLAGS

In the original ADAE data structure there was not a defined identifier for the first treatment emergent record. As a result in TFL programming one would first need to select the first or unique record before counting the number of occurrences. The OCCDS guide provides better traceability with the permissible occurrence flags AOCCFL, AOCCPFL, AOCCSFL. Flagging the first occurrence within respectively Subject, Preferred Term and System Organ Class. The use of these flags allows on to get the counts for a typical adverse events summary table by simply filtering the data on the occurrence flag and treatment variable selection. The example below shows how each variable contributes to the table.



System Organ Class	Preferred Term	Vaccine A (N=xxx) n(%)	Vaccine B (N=xxx) n(%)
At Least one symptom		x (x.x)	x (x.x)
Gastrointestinal disorders		x (x.x)	x (x.x)
	At Least one symptom	x (x.x)	x (x.x)
	Diarrhoea	x (x.x)	x (x.x)
	Vomiting	x (x.x)	x (x.x)
Nervous system disorders			
	At Least one symptom	x (x.x)	x (x.x)
	Dizziness	x (x.x)	x (x.x)
	Headache	x (x.x)	x (x.x)
	Presyncope	x (x.x)	x (x.x)

AOCCFL
AOCCSFL
AOCCPFL
AOCCSFL
AOCCPFL

HOW MANY FLAGS DO WE NEED

The IG provides 6 common occurrence flags. One set of 3 for "first occurrence of an event". And one set of 3 for the "first maximum severity/intensity of an event". Additional flag naming is in the format of AOCCzzFL. For a very basic study and analysis this set of 6 flags is sufficient. However in a vaccines analysis the focus is often on a specific set of events categorizations. There are 2 distinct categorizations:

The first category is timing of the event. Examples of this are: the first event after a vaccination, the first event within the first 7 days after vaccination, the first event from 8 days after vaccination until the next vaccination, etc.. To take all these categories into account, we would need another 6x3=18 flags. This in addition to the 6 we already had, makes a total of 24 flags.

The second category are adverse events of special interest. For example related ae's, ae's leading to premature withdrawal, ae's leading to hospitalization, serious adverse events, etc.. In our standard we have 11 of these categories. For each of the first category (of timing), an analysis of the second category can be done. E.g. serious ae's overall, serious ae's after each vaccination, serious ae's in the first 7 days etc..

The end result is that we would need 24x11 = **264** occurrence flags. This leads to several issues. First is that the AOCCzzFL only allows 99 variables. Second is that specifying such a huge number of variables is a lot of work and error prone due to the amount of similar specifications. And last but not least, it is not transparent at all as one does not know what AOCC78FL signifies. Plus on top of that, a flag may refer to a different selection in another analysis.

POSSIBLE SOLUTION

One solution would be to not use the occurrence flag variables. They are permissible meaning you do not need to include them. However for improved traceability I prefer to use the occurrence flag variables.

A second solution is to create records for each set of categories for the source data. The I.G. provides a categorization variable ACATy for this. Add variables ACAT1 and ACAT2 to identify the categories. See blow for table 2 created from data in table 1. This way we always have a definite set of 6 occurrence flag variables that always have the same meaning.

Table 1

Example of an adverse event "HEADACHE" that occurred twice.

AEDECOD	ASTDT	AEREL	AESER	AEOUT
HEADACHE	11-OCT-2015	Y	Y	
HEADACHE	9-OCT-2016			WITHDRAWAL

Taking the data from table 1, to ensure full traceability, both records are included in ADAE with for both records ACAT1 set to 'OVERALL'. These are the grey shaded records in table 2 below. Next, each record is assigned to a time period and a new record is created. These are the blue shaded records in table 2. Note a record can be assigned to multiple time periods. For example if there is an analysis of ae's in the period of day 1-7 after vaccination and an analysis of ae's in the period of day 1-28 after vaccination.

As the last step, the second category ACAT2 is populated. For the original records, ACAT2 is set to 'OVERALL' as well. Finally, based on a selection process each shaded record is evaluated for the applicable criteria, in this case relationship to study vaccine, seriousness and outcome. If the criteria are satisfied, a record is created and the ACAT2 value assigned.

In the second step we do not duplicate all records, only the records satisfying the categorization criteria as to not increase the dataset size too much. The 'OVERALL-OVERALL' category already contains all the source records for full traceability

Table 2

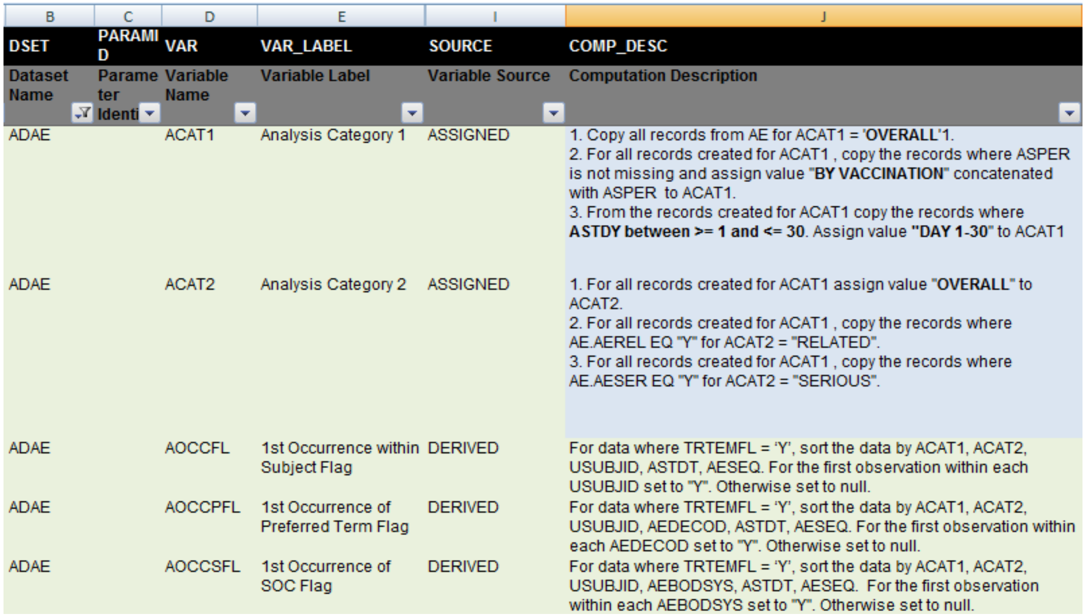
Records in the ADAE dataset

AEDECOD	ASTDT	ACAT1	ACAT2	AOCCFL	AOCCPFL	AOCCSFL	AOCCIFL	AOCCPIFL	AOCCSIFL
HEADACHE	11-OCT-2015	OVERALL	OVERALL	Y	Y	Y	Y	Y	Y
HEADACHE	11-OCT-2015	OVERALL	RELATED AE	Y	Y	Y	Y	Y	Y
HEADACHE	11-OCT-2015	OVERALL	SERIOUS AE	Y	Y	Y	Y	Y	Y
HEADACHE	9-OCT-2016	OVERALL	OVERALL						
HEADACHE	9-OCT-2016	OVERALL	LEAD TO WD	Y	Y	Y	Y	Y	Y
HEADACHE	11-OCT-2015	AFTER VAC 1	OVERALL	Y	Y	Y	Y	Y	Y
HEADACHE	11-OCT-2015	AFTER VAC 1	RELATED AE	Y	Y	Y	Y	Y	Y
HEADACHE	11-OCT-2015	AFTER VAC 1	SERIOUS AE	Y	Y	Y	Y	Y	Y
HEADACHE	9-OCT-2016	AFTER VAC 2	OVERALL	Y	Y	Y	Y	Y	Y
HEADACHE	9-OCT-2016	AFTER VAC 2	LEAD TO WD	Y	Y	Y	Y	Y	Y

SOLUTION FROM PROGRAMMING PERSPECTIVE

This solution keeps the specifications and programming logic quite simple and easy to check. It can be simple as.

Specifications:



B	C	D	E	I	J
DSET	PARAMID	VAR	VAR_LABEL	SOURCE	COMP_DESC
Dataset Name	Parameter Identi	Variable Name	Variable Label	Variable Source	Computation Description
ADAE		ACAT1	Analysis Category 1	ASSIGNED	1. Copy all records from AE for ACAT1 = '**OVERALL**'1. 2. For all records created for ACAT1 , copy the records where ASPER is not missing and assign value "**BY VACCINATION**" concatenated with ASPER to ACAT1. 3. From the records created for ACAT1 copy the records where **ASTDY between >= 1 and <= 30**. Assign value **"DAY 1-30"** to ACAT1
ADAE		ACAT2	Analysis Category 2	ASSIGNED	1. For all records created for ACAT1 assign value "**OVERALL**" to ACAT2. 2. For all records created for ACAT1 , copy the records where AE.AEREL EQ "Y" for ACAT2 = "RELATED". 3. For all records created for ACAT1 , copy the records where AE.AESER EQ "Y" for ACAT2 = "SERIOUS".
ADAE		AOCCFL	1st Occurrence within Subject Flag	DERIVED	For data where TRTEMFL = 'Y', sort the data by ACAT1, ACAT2, USUBJID, ASTDT, AESEQ. For the first observation within each USUBJID set to "Y". Otherwise set to null.
ADAE		AOCCPFL	1st Occurrence of Preferred Term Flag	DERIVED	For data where TRTEMFL = 'Y', sort the data by ACAT1, ACAT2, USUBJID, AEDECOD, ASTDT, AESEQ. For the first observation within each AEDECOD set to "Y". Otherwise set to null.
ADAE		AOCCSFL	1st Occurrence of SOC Flag	DERIVED	For data where TRTEMFL = 'Y', sort the data by ACAT1, ACAT2, USUBJID, AEBODSYS, ASTDT, AESEQ. For the first observation within each AEBODSYS set to "Y". Otherwise set to null.

Programming:

```
DATA adae1;
  SET ae;
  acat1 = 'OVERALL';
  OUTPUT;
  acat1 = 'BY VACCINATION '||vaccinationvariable;
  OUTPUT;
RUN;

DATA adae2;
  SET adae1;
  acat2 = 'OVERALL';
  OUTPUT;
  IF aerel = 'Y' THEN DO;
   acat2 = 'RELATED';
   OUTPUT;
  END;
  IF aeser = 'Y' THEN DO;
   acat2 = 'SERIOUS';
   OUTPUT;
  END;
  IF AEOUT = 'WITHDRAWAL' THEN DO;
   acat2 = LEAD TO WD';
   OUTPUT;
  END;
RUN;

PROC sort DATA = adae2 OUT = adae;
  BY acat1 acat2 usubjid aedecod;
RUN;

DATA adae;
  SET adae;
  BY acat1 acat2 usubjid aedecod;
  IF FIRST.aedecod THEN aoccpfl = 'Y';
RUN;
```

The analysis is then very straight forward. With a few lines of code all the counts for all categories can be produced. For example:

```
PROC sort DATA = adam.adae OUT = adae;
  BY acat1 acat2 actarmcd;
RUN;

PROC freq DATA = adae;
  BY acat1 acat2 actarmcd;
  TABLES aedecod aoccpfl;
RUN;
```

CONCLUSION

The occurrence flags are only useful if they have same meaning in each study. This means, as much as possible the flags should be restricted to the flags suggested in the implementation guide. From the name you can deduce its purpose. A couple of AOCCzzFL flags can also be useful if they serve their own unique purpose, but they should not have the same purpose as the suggested occurrence flags, for a different subset. E.g. do not create AOCC01FL for the first occurrence within subject for adverse events occurring between day 1 to day 7 after vaccination. The specification and logic for creating the records and performing the analysis is straight forward and compact. Creating additional records takes less computing time then creating an uncountable number of flags. The reason is that for each flag a sort step needs to take place. By limiting the number of flags, far fewer sort steps are needed. The dataset does get bigger. But the purpose of an analysis dataset is to make analysis transparent and easy. The tradeoff is a bigger dataset. A comment from one of our programmers to this approach, after initial skepticism was that it is now too easy. All he needed to do was write a simple restriction clause for the analysis results metadata and apply that to the TFL program.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Rob Wartenhorst
GSK Vaccines
Hullenbergweg 81-89
1101 CL Amsterdam
The Netherlands
Work Phone: +31 205640564
Email: Rob.x.wartenhorst@gsk.com