Merging Supplemental Data onto Parent Domains

Dirk Van Krunckelsven, CS01, Phuse 2015
Supplemental Qualifiers

<table>
<thead>
<tr>
<th>SUPPQUAL</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>STUDYID</td>
<td>RDOMAIN</td>
<td>USUBJID</td>
<td>IDVAR</td>
<td>IDVARVAL</td>
<td>QNAM</td>
<td>QLABEL</td>
<td>QVAL</td>
<td>QORIG</td>
<td>QEVAL</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Store non-standard variables in SDTM
along with their association to parent records
in general-observation class and demographics
Supplemental Qualifiers

- **value (QVAL)**
- **variable name (QNAM) and label (QLABEL)**
- **basic association: subject, domain, study**
- **specific record association**
Early struggle: 2 datasets!?

We now have to create two datasets for almost every domain: a parent dataset and a supplemental dataset. This involves a lot more work to create and validate. Additional steps are now involved. And why? When analyses are being performed then the data is being put back together again. So we are just doing this for the sake of doing it rather than focusing on our data being solid.

I have heard the promise of standard data bringing standard programs and easier data handling in the future, but now, at this moment in time, it only brings headache and overtime…
2010: pick up existing code
Make available as standard program
Robust, user-friendly

Stick to promise: Standard data brings standard programs
Instrumental in SDTM acceptance
%supp2par_v1

Code and documentation available to all – no repeat of user doc.

_Supp2par_v1.sas: Merge SDTM supplemental with parent_

1 Table of Contents
1 Table of Contents ................................................................. 1
2 Description ................................................................. 1
3 Location ........................................................................... 1
3.1 Signature ....................................................................... 2
3.2 Input Data Parameters ................................................. 2
3.3 Output Data Parameters ............................................... 2
3.4 Control and Log Parameters .......................................... 2
4 Data Requirements ............................................................ 3
5 Output Produced ............................................................... 3
5.1 Merged dataset ............................................................. 3
5.2 DROP dataset ............................................................... 5
5.3 DUPP dataset ............................................................... 5
5.4 IGNOR dataset ............................................................. 5
6 Return codes ...................................................................... 6
6.1 WARNING Return Codes ............................................... 6
6.2 Termination Return Codes .............................................. 6
7 Changes and enhancements ............................................. 7
7.1 Release 2 ....................................................................... 7
7.2 Release 3 ....................................................................... 7
Macro Call

%SUPP2PAR_v1(

* Input parameters, required *
* Output parameters, optional *
* Control and Log parameters, optional *

inlib=, /* Name of input library for Parent and Supplemental Datasets */
parent=, /* Name of Parent dataset */
supp=, /* Name of Supplemental dataset */
outlib=, /* Name of destination library for Merged dataset */
outname=, /* Name of Merged dataset */
clean=, /* Whether or not (Y/N) to clean work environment at end */
dev=, /* Whether or not (Y/N) to show all messages in the log */
Info=, /* Level of information to provide in the log:
1 (all), 2 (start and summary), 3 (only summary) */
RC= /* Name of Return Code Macro variable */);
Essence of %Supp2par_v1

- Transpose the supplemental dataset
- Merge this transposition onto parent dataset
- Using the specified keys

EASY  TRIVIAL
Step 1: Transpose

```
proc transpose data = S2P_SUPP[NAME]
  (rename = (RDOMAIN = DOMAIN))
out = S2P_SUPP[NAME]T
  (drop = _NAME_ _LABEL_);
by STUDYID DOMAIN USUBJID [IDVAR(VAL)];
var qval;
  id qnam;
  idlabel qlabel;
run;
```
Step 1: Complication IDVAR[VAL]

Complete the association for a variable (IDVAR) having a value (IDVARVAL)

Mostly sequence number if multiple records per subject
If empty STUDYID RDOMAIN and USUBJID are sufficient keys

Does not have to be - - SEQ !!!
Does not have to be the same variable within a supp. dataset !!!
### PARENT DATASET

<table>
<thead>
<tr>
<th>USUBJID</th>
<th>AESEQ</th>
<th>AECAT</th>
<th>[..p]</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>1</td>
<td></td>
<td>[..p]</td>
</tr>
<tr>
<td>0001</td>
<td>2</td>
<td>NEURO</td>
<td>[..p]</td>
</tr>
<tr>
<td>0002</td>
<td>1</td>
<td></td>
<td>[..p]</td>
</tr>
<tr>
<td>0002</td>
<td>2</td>
<td></td>
<td>[..p]</td>
</tr>
<tr>
<td>0003</td>
<td>1</td>
<td></td>
<td>[..p]</td>
</tr>
<tr>
<td>0003</td>
<td>2</td>
<td>NEURO</td>
<td>[..p]</td>
</tr>
<tr>
<td>0003</td>
<td>3</td>
<td>NEURO</td>
<td>[..p]</td>
</tr>
<tr>
<td>0003</td>
<td>4</td>
<td>NEURO</td>
<td>[..p]</td>
</tr>
</tbody>
</table>

### SUPPLEMENTAL QUALIFIERS DATASET

<table>
<thead>
<tr>
<th>USUBJID</th>
<th>RDOMAIN</th>
<th>IDVAR</th>
<th>IDVARVAL</th>
<th>QNAM</th>
<th>[..s]</th>
<th>QVAL</th>
<th>QORIG</th>
</tr>
</thead>
<tbody>
<tr>
<td>0001</td>
<td>AE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AESEQ</td>
<td>1</td>
<td>RELCMP</td>
<td>[..s]</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>0001</td>
<td>AE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AESEQ</td>
<td>3</td>
<td>RELCMP</td>
<td>[..s]</td>
<td>Y</td>
<td></td>
</tr>
<tr>
<td>0001</td>
<td>AE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AECAT</td>
<td>NEURO</td>
<td>DSM</td>
<td>[..s]</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>0003</td>
<td>AE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AESEQ</td>
<td>4</td>
<td>LABVAL</td>
<td>[..s]</td>
<td>12,45</td>
<td>CRF</td>
</tr>
<tr>
<td>0003</td>
<td>AE</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>AECAT</td>
<td>NEURO</td>
<td>DSM</td>
<td>[..s]</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>USUBJID</td>
<td>AESEQ</td>
<td>AECAT</td>
<td>[..p]</td>
<td>SUPPLEMENTAL VARIABLES</td>
<td>LABVAL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>---------</td>
<td>-------</td>
<td>-------</td>
<td>-------</td>
<td>-------------------------</td>
<td>--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00001</td>
<td>1</td>
<td></td>
<td>[..p]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00001</td>
<td>2</td>
<td>NEURO</td>
<td>[..p]</td>
<td>RECOMP</td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00002</td>
<td>1</td>
<td></td>
<td>[..p]</td>
<td>DSM</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00002</td>
<td>2</td>
<td></td>
<td>[..p]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>00003</td>
<td>1</td>
<td></td>
<td>[..p]</td>
<td></td>
<td>Y</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00003</td>
<td>2</td>
<td>NEURO</td>
<td>[..p]</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00003</td>
<td>3</td>
<td>NEURO</td>
<td>[..p]</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>00003</td>
<td>4</td>
<td>NEURO</td>
<td>[..p]</td>
<td></td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>12,45</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
IDVAR[VAL]: Not so trivial after all…

One transpose (and merge) per IDVAR

Discover the IDVARs

Loop over IDVARs

Datasets Merge lateron

Rename IDVARVAL to be actual variable name (IDVAR)

or if empty merge on USUBJID

IDVARVAL is always character vs. IDVAR-key may not be (SEQ)

Discover attributes on parent + Change IDVARVAL
Step 2: Merge

data S2P_[NAME] S2P_[NAME]_DROP;
   merge S2P_PAR[NAME] (in = PARENT)
       S2P_SUPP[NAME]T (in = SUPP);
   by STUDYID DOMAIN USUBJID [IDVAR(VAL)];
   if PARENT then output S2P_[NAME];
else if SUPP and not(PARENT) then output S2P_[NAME]_DROP;
run;

ROBUST

INFORMATIVE
Robust and Informative

Essentially: transposition and remerge

Over 1000 lines of code for just that!?

Checks on input data

Checks on macro parameters

Keeping track of “special” situations: incremental return codes

Robust and Informative

10 non-terminating messages
Something is odd, process can continue, user is notified

14 terminating messages
Something is wrong, process will fail, user is notified
Robust and Informative: 3 examples

Supplemental not unique on USUBJID, RDOMAIN, [IDVAR(VAL)], QNAM

- Cannot be remedied – process stops
- [NAME]_DUPP dataset created for user to understand and action

Not every supplemental record fits to a parent record

- Can continue but need to know
- [NAME]_DROP dataset created for user to understand and action

Records with empty STUDYID, USUBJID, RDOMAIN or QNAM

- Can continue but need to know
- [NAME]_IGNOR dataset created for user to understand and action
Informative

******************************
Started Processing for
  PARENT = WORK.SV
  SUPP   = WORK.SUPPSV
******************************

******************************
Finished Processing Successfully without Remarks
  CREATED: OD1.SVZ82
******************************

Nevertheless, do inspect the log...
ERROR:  ******************************************************
ERROR:  * DATASET WORK.SUPPEX AS SPECIFIED IN PARAMETER SUPP
ERROR:  *  HAS 1 DUPLICATE RECORD(S)
ERROR:  *  ON THE KEYS USUBJID RDOMAIN IDVAR IDVARVAL QNAM
ERROR:  *  IT IS IMPOSSIBLE TO DETERMINE WHICH RECORD TO MERGE
ERROR:  *  THE MACRO WILL TERMINATE
ERROR:  ******************************************************
ERROR:  * Duplicate Records listed here (limited to 15)
ERROR:  *  Simple list of key values mentioned above
ERROR:  *  All duplicate records stored in WORK.SUPPEX._DUPP
ERROR:  * 275750120002 EX EXSEQ 1 VOLSYN
ERROR:  ******************************************************

* EARLY TERMINATION DUE TO: *
*  Supp not unique on USUBJID RDOMAIN IDVAR IDVARVAL QNAM
ERROR:
******************************************************************************
ERROR: * DATASET INPUT. AS SPECIFIED IN PARAMETER PARENT
ERROR: * DOES NOT EXIST, TERMINATED MACRO
ERROR:
******************************************************************************

******************************************************************************
* EARLY TERMINATION DUE TO: *
* Parent dataset does not exist
******************************************************************************
Informative

Finished Processing Successfully with Remarks:

CREATED: OD1.DMPLUS

* Outlib does not exist, defaults to WORK
* Supp dataset does not exist, output will be copy of parent
* Supp dataset is empty, output will be copy of parent
* Improper Outname specified, defaults to DMFULL
* Supplemental empty after selecting for RDOMAIN
* QNAM value in Supp is already a var on Parent. Suffixed:_2
* Records on Supp do not exist on the Parent, refer to: SUPPDM_DROP
* QNAM value in Supp is not a Valid SAS variable Name
* Records ignored: empty STUDYID USUBJID RDOMAIN or QNAM, refer to: SUPPDM_IGNOR

For CDISC compliance most of these messages must be fixed
Shortcomings

One call per pair of Parent - Supplemental

- Can easily be embedded in a ‘caller’ program
- Caller should use return codes and keep track for each pair

The Supplemental ‘variables’ are all character data

- Should be able to read the define.xml or other metadata
- Create supplemental ‘variables’ immediately as intended
Shortcomings

QEVAL and QORIG disappear

- On purpose
- Every QNAM could lead to 3 variables
- How would you name these?

SDTM 1.5: What about POOLID and APID

- Need to evaluate the requirements
- Possibly rework the macro
The Phuse Standard Scripts Repository

2013 Phuse/CSS in Washington: WG5 ‘Standard Scripts’
- Call for code
- Decided to share
- Uploaded to Google Code – NOW: GitHub

Collaboration: Entered comments on where to improve the code
- Comments not picked up
- Clear? Knowledge on Usage
- Too polite?
- ...

Clear?
In the Phuse Standard Scripts Repository

https://github.com/phuse-org/phuse-scripts/tree/504932bda0f297fd06fea85aacbc5839db0c66d9/lang/SAS/datahandle/supp2par

http://tinyurl.com/supp2par

The Phuse/CSS Working Group

http://www.phuse.eu/CS-Dashboard.aspx
Questions