

# 1 Table of Contents

1	Table of Contents .....	1
2	Description .....	1
3	Location .....	1
4	Syntax .....	2
4.1	Signature.....	2
4.2	Input Data Parameters .....	2
4.3	Output Data Parameters .....	2
4.4	Control and Log Parameters.....	2
5	Data Requirements.....	3
6	Output Produced .....	3
6.1	Merged dataset .....	3
6.2	DROP dataset .....	5
6.3	DUPP dataset.....	5
6.4	IGNOR dataset .....	5
7	Return codes.....	6
7.1	WARNING Return Codes.....	6
7.2	Termination Return Codes .....	6
8	Author .....	6

## 2 Description

supp2par\_v1.sas is a SAS macro that **merges SDTM supplemental qualifier data onto its SDTM parent domain** data transposing records in the supplemental data into *character* variables.

Generally only one output dataset is produced, being the dataset that contains the parent domain merged together with the supplemental qualifiers, refer to section 6.1.

The macro will always attempt to reach a result and only terminate without the creation of an output (merged) dataset if there is no other option, the output (merged) dataset may be created as a simple copy of the parent data. The log and return codes inform the user and/or a running program whenever the processing fails and/or assumptions have been made as shown in section 7. Note that the user can choose the name of the macro variable containing the return code, as detailed under section 4.4 for the **RC** parameter.

In some cases additional datasets will be produced, this only occurs when there is some issue in the data. See section 6.2, 6.3 and 6.4 below. These datasets are present only to have an easy reference for further improving the input data.

## 3 Location

```
P:\Drugs\Macros\Programs\supp2par_v1.sas
```

## 4 Syntax

### 4.1 Signature

```
%SUPP2PAR_v1(
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 */);
```

### 4.2 Input Data Parameters

**inlib:** Required, name of library containing datasets specified in **parent** and **supp**.

**parent:** Required, name of input dataset containing the parent domain data. Dataset must be SDTM compliant, some non-compliances lead to early termination of the macro, refer to section 5. No default.

**supp:** Required, name of input dataset containing the supplemental qualifier data. Dataset must be SDTM compliant, some non-compliances lead to early termination of the macro, refer to section 5. No default.

### 4.3 Output Data Parameters

**outlib:** Optional, name of library in which to write the output (merged) dataset specified by the parameter **outname**. Defaults to *work*.

**outname:** Optional, name of output (merged) dataset. Defaults to the name provided in the **parent** parameter suffixed with *FULL*.

### 4.4 Control and Log Parameters

**clean:** Optional, whether temporary datasets created by the macro in the work library should be deleted (Y) or not (N) upon completion. Valid values are Y and N, all other values revert to the default behaviour. Defaults to Y.

**dev:** Optional. Is this a development run? On a *non-development run* (N) messages arising from SAS options SYMBOLGEN, MACROGEN, SOURCE, SOURCE2, MLOGIC, and NOTES are turned off. For *development runs* (Y), these options are untouched, i.e. they remain in effect if these options are in effect when the macro is called. If the macro alters these options, then upon completion they are returned to their original settings. Valid values are Y and N, all other values revert to the default behaviour. Defaults to N (i.e. options turned off).

**info:** Optional, determines how the log informs the user about the processing. Valid values are 1, 2 and 3, other values revert to default behaviour. Three states: 1: all information provided, i.e. a start message, a summary message, and possibly messages in between informing on data/processing issues, 2: only start and summary provided, 3: only summary provided. Defaults to 1.

**RC:** Optional, name of the global macro variable to hold the return code. Defaults to *RC\_S2P*.

## 5 Data Requirements

The macro requires **SDTM compliant** input data: a parent dataset and a supplemental qualifier dataset that have both been constructed adequately for the data at hand with the correct keys available, valid values for important variables and so forth. Both datasets will require some variables to exist in order for the merge to be possible.

All data requirements are captured by the macro and the user will be informed if data assumptions are not being met. Refer to the list of Return Codes further down

Note: As this is purely SDTM implementation, not all data requirements will be listed here, nor will the requirements listed here suffice for SDTM compliance.

Variables that must be present on the parent domain (*parent*):

- |           |  |
|-----------|--|
| - STUDYID | - USUBJID                                |
| - DOMAIN  | - <vars listed in IDVAR in <i>supp</i> > |

Variables that must be present on the supplemental qualifier dataset (*supp*):

- |           |            |
|-----------|------------|
| - STUDYID | - IDVARVAL |
| - RDOMAIN | - QNAM     |
| - USUBJID | - QLABEL   |
| - IDVAR   | - QVAL     |

## 6 Output Produced

### 6.1 Merged dataset

The merged dataset contains the parent dataset as is, with the supplemental qualifiers, transposed into variables, merged onto it on the right hand side (i.e. denormalized). The dataset is stored in the library specified by the **outlib** parameter and named as specified in the **outname** parameter.

The supplemental qualifiers will show the variable name as indicated in the QNAM variable on the supplemental dataset, the variable label will be as per the content of the QLABEL variable and the variable value as per the content in the QVAL variable.

The newly merged supplemental variables will always be text/character variables.

Note: The *Origin* and *Evaluator* information that may be present on the supplemental qualifier dataset are ignored. The macro will not notify the user if this is happening when data is present in these variables. Merging on this information – in a generic fashion – would imply a triplication of the amount of merged variables whereas it is unlikely needed for every QNAM. If the information is needed, the user will have to post-process.

#### 6.1.1 Merge through IDVAR - IDVARVAL

The most straightforward way for a supplemental qualifier to merge back onto a parent dataset is through the sequence number (--SEQ) so that one record in the parent dataset receives a variable with a value as found in one record on the supplemental dataset. The macro however uses the information provided in the supplemental dataset in the variables IDVAR and IDVARVAL. Respectively these provide the identifying variable on the parent dataset that identifies the related record and the value of that identifying variable. Often the sequence number will be indicated here, but it can be a different variable, such as for example a category (--CAT). In the latter case, one record for a subject (see Note 2 below) in the supplemental dataset will merge its value onto every record for that subject of the parent dataset which has the specified IDVARVAL for the category.

Note: When IDVAR and IDVARVAL are left empty, the macro will attempt to merge through USUBJID.

Note 2: Supplemental qualifier datasets are subject-based by definition, therefore an SDTM-compliant supplemental qualifier dataset cannot provide any information “*in general*”, i.e. disregarding the subject. It is thus impossible for all records of a category to receive the content for a supplemental variable through just one record in the supplemental dataset. The supplemental dataset must always have the information specified for every subject for which the content applies.

### 6.1.2 Simplified Example of Merging a Supplemental onto its Parent Domain

The following is a simplified representation of a parent dataset, in this case AE. Notice that only a few variables are shown here, which is indicated by [..p].

PARENT DATASET			
USUBJID	AESEQ	AECAT	[..p]
0001	1		[..p]
0001	2	NEURO	[..p]
0002	1		[..p]
0002	2		[..p]
0003	1		[..p]
0003	2	NEURO	[..p]
0003	3	NEURO	[..p]
0003	4	NEURO	[..p]

The following is a simplified representation of a supplemental qualifier dataset, in this case SUPPAE. Here a few variables such as QLABEL have been left out as well, which is indicated by [..s].

SUPPLEMENTAL QUALIFIERS DATASET							
USUBJID	RDOMAIN	IDVAR	IDVARVAL	QNAM	[..s]	QVAL	QORIG
0001	AE	AESEQ	1	RELCMP	[..s]	Y	
0001	AE	AECAT	NEURO	DSM	[..s]	4	
0003	AE	AESEQ	1	RELCMP	[..s]	Y	
0003	AE	AESEQ	4	LABVAL	[..s]	12,45	CRF
0003	AE	AECAT	NEURO	DSM	[..s]	4	

The following is the merged output dataset where the supplemental qualifiers appear as variables.

PARENT DATASET				SUPPLEMENTAL VARIABLES		
USUBJID	AESEQ	AECAT	[..p]	RELCMP	DSM	LABVAL
0001	1		[..p]	Y		
0001	2	NEURO	[..p]		4	
0002	1		[..p]			
0002	2		[..p]			
0003	1		[..p]	Y		
0003	2	NEURO	[..p]		4	
0003	3	NEURO	[..p]		4	
0003	4	NEURO	[..p]		4	12,45

The values for QNAM from the supplemental dataset now appear as variables with values as given in QVAL in the supplemental dataset.

Also notice that all the variables present in the parent dataset also appear in this merged dataset. This is also true for the number of observations. In fact, one can easily observe that the entire parent dataset is still present in the merged output dataset. This is not true for the supplemental dataset, the variable QORIG is not present on the merged output dataset, nor are the variables indicated by [..s].

For both subjects 0001 and 0003, the supplemental dataset has a record with value '4' (QVAL) for QNAM = 'DSM'; the IDVAR for these records is set to 'AECAT' and the IDVARVAL to 'NEURO'. In the merged output dataset, the variable DSM shows up with value '4' more often than provided in the supplemental qualifier dataset. This is due to the merge taking place on the *identifying variable* (IDVAR) being the category (AECAT): whenever AECAT's value equals the *identifying variable value* (IDVARVAL), namely 'NEURO', the record is written out to the output dataset. The other supplemental qualifiers, RELCMP and LABVAL have a more straightforward merge taking place on the sequence number.

Note also that subject 0002 has no records in the supplemental qualifier dataset, in the merged output dataset this subject will simply never have a value for any of the supplemental variables.

## 6.2 DROP dataset

If records exist on the supplemental that could not be merged, then these are presented in this dataset. The dataset is always stored in the WORK library, and will be named according to the value in the **supp** parameter suffixed with **\_DROP**. This will occur whenever the identifying variable value (IDVARVAL) present in the supplemental is not present on the parent dataset, refer to section 6.1.1, which details the merging principle.

Note: in this case the macro will still produce the output (merged) dataset and the return code -128 will capture the event.

## 6.3 DUPP dataset

If duplicate records exist on the supplemental then these are written into this dataset, one record for each duplicate key occurrence. The dataset is always stored in the WORK library, and will be named according to the value in the **supp** parameter suffixed with **\_DUPP**.

Note: in this case the macro will terminate without producing an output (merged) dataset, the return code 10 captures this event.

## 6.4 IGNOR dataset

If records exist on the supplemental that have an empty value for USUBJID, RDOMAIN and/or QNAM, then these are presented in this dataset. They are ignored for further processing. The dataset is always stored in the WORK library, and will be named according to the value in the **supp** parameter suffixed with **\_IGNOR**.

Note: in this case the macro will still produce the output (merged) dataset and the return code -512 will capture the event.

## 7 Return codes

Conditions which are considered ERRORS lead to macro termination and produce positive return codes. Conditions which are considered WARNINGS produce negative return codes. These latter may require the input data to be altered or may just inform the user of default choices or data topics.

Note: Negative return codes are summed as binary numbers so that all WARNINGS can be caught in one return code, e.g. -5 occurs when OUTLIB does not exist while at the same time the supplemental dataset is empty. In the binary sum, the value -5 can only result whenever these two issues occur.

Note: The return code is always produced in the macro variable as specified in the parameter **rc**. Whenever a termination or warning occurs, this is always detailed in the summary in the log when the macro is being completed. The "in-between" messages (i.e. when **info = 1**) will be more detailed than the summary messages which are shown in following overviews.

### 7.1 WARNING Return Codes

RC	Issue and action
-1	Outlib does not exist, defaults to WORK
-2	Supplemental dataset does not exist, output will be copy of parent
-4	Supplemental dataset is empty, output will be copy of parent
-8	Improper Outname specified, defaults to [PARENT]FULL: [...]FULL
-16	IDVAR and IDVARVAL were empty substituted by USUBJID
-32	Supplemental empty after selecting for RDOMAIN
-64	QNAM value in Supplemental is already a variable on Parent. Suffixed with _2
-128	Records on Supplemental do not exist on the Parent domain, refer to: [...] _DROP <u>Note</u> : In this case the DROP dataset is created, as described in section .
-256	QNAM value in Supplemental is not a Valid SAS variable Name
-512	Records ignored: empty STUDYID USUBJID RDOMAIN or QNAM, refer to: [...] _IGNOR <u>Note</u> : In this case the IGNOR dataset is created, as described in section .

### 7.2 Termination Return Codes

RC	Error
1	Inlib does not exist
2	Parent dataset does not exist
3	Parent dataset is empty
5	Required variables not present on the Parent dataset
6	Required variables not present on the Supplemental dataset
7	Same QNAM has different labels
8	IDVAR empty while IDVARVAL not empty
9	IDVAR(s) not present on PARENT domain
10	Supplemental dataset not unique on USUBJID RDOMAIN IDVAR IDVARVAL QNAM. <u>Note</u> : In this case the DUPP dataset is created as described in section 6.3.
500	Data Step Error trapped in the macro
599	Data Step Error trapped in the macro: Cannot write output dataset
600	Procedure Error trapped in the macro
1000	Running the macro under the current SAS version and/or OS is not allowed
9999	Unexpected error

## 8 Author

Dirk Van Krunkelsven