

Paper PO101

Ouch, How did that get there? The pitfalls of merging.

Nancy McGarry, Abt Associates Inc., Cambridge, MA

ABSTRACT

This poster is a short slide show presentation of the pitfalls of merging. It highlights some common problems of the unwary, looking at 5 common missteps in merging data in hopes of preventing errors by making the audience more aware of things that can go wrong during a DATA step merge.

INTRODUCTION

Frequently when doing analysis it is necessary to combine multiple sources of data. In SAS®, one of the ways to do this is using a DATA step with a merge statement. Use of the merge statement can be fraught with minefields. The poster (and paper) examine some very common mistakes that occur.

**COMMON MISTAKE #1: NOT USING A BY STATEMENT**

If there is a common identifier, sort each (or all) datasets by that variable and remember to use the identifier in the by statement:

```
PROC SORT DATA = file1;
  BY id;

PROC SORT DATA = file2;
  BY id;

DATA file3;
  MERGE file1 file2;
  BY id;
RUN;
```

COMMON MISTAKE #2: IMPLICIT RETAIN STATEMENT AND/OR CARRYING DATA FROM PREVIOUS RECORDS

Sometimes SAS® carries information over in implicit retain statements. So don't try to do too much in the data step. It is better to do a second data step after a merge if creating new variables:

```
DATA file3;
  MERGE file1 file2;
  BY id;
```

```

RUN;

DATA file3a;
  SET file3;
  c=SU<(a,b);
  new_d = CONCAT(x,y);
RUN;

```

COMMON MISTAKE #3: ERROR IN UPDATING FILES WITH A COMMON VARIABLE

Assuming that file 2 contains complete information for all records, remember to drop the variable on the dataset that is being updated:

```

PROC SORT DATA = file1;
BY id;
PROC SORT DATA = file2;
BY id;

DATA file3;
  MERGE file1 (DROP=partner1) file2;
  BY id;
RUN;

```

All fields are updated.

COMMON MISTAKE #4: MIXING CHARACTER AND NUMERIC FIELDS WITH THE SAME VARIABLE NAME

If 2 datasets have the same variable name but one is character and the other is numeric then rename one of the variables and resolve/update the other in a later datastep:

```

DATA file3;
  MERGE file1 (RENAME=(date=date1)) file2;
  BY id;
RUN;

```

COMMON MISTAKE #5: NOT REALIZING ORDER IS IMPORTANT WHEN MERGING

Be aware of which file is replacing data. The last file mentioned will overwrite earlier files:

```

DATA file3;
  MERGE file1 file2;
  BY id;
RUN;

```

DOES NOT GIVE THE SAME RESULTS AS

```
DATA file3;  
    MERGE file2 file1;  
    BY id;  
RUN;
```

CONCLUSION

The merge statement can be very powerful but to perform successful merges always consider the following:

- 1) Know your data
- 2) Sort your data and use the by statement to control matching
- 3) Check a sample of your output to make sure you are not carrying information from one record to the next
- 4) If you aren't getting all the information you expected, check to see if you are replacing or adding a variable
- 5) Make sure any variable you want replaced is the same type as its replacement
- 6) Make sure the file that is doing the updating is the last file on your merge statement



CONTACT INFORMATION

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

Name: Nancy McGarry

E-mail: Nancy.McGarry@abtassoc.com

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.