

Slim Down Your Data

October 13, 2015

Mickael Borne



Glossary

- ❑ FDA: US Food and Drug Administration
- ❑ **CRT: Case Report Tabulation** (Datasets (SDTM, ADaM,...), corresponding define.xml, Analysis programs,...)
- ❑ ICH: International Conference on Harmonisation of Technical Requirements for Registration of Pharmaceuticals for Human Use
- ❑ CTD: Common Technical Document (a set of specifications for the dossier used in the registration of medicines)
- ❑ CDISC: Clinical Data Interchange Standards Consortium
- ❑ CDER/CBER: FDA's Center for Drug/Biologics Evaluation and Research

Introduction

- ❑ A package of SAS® macro-programs that was developed to automatically resize character variables of all SAS datasets in a project directory.
- ❑ The allotted character variable length/size can significantly impact the size of the corresponding SAS dataset file.
- ❑ Adjusting this length to the maximum length observed is recommended for electronic submission of clinical trial data.
- ❑ For example SUBJID variable has been defined with a length of 200 characters in all your datasets, but SUBJID has a maximum length of 10 characters after using these macros all the SUBJID will be declared with a length of 10 characters.
- ❑ Agenda:
 - The context: Electronic submission of clinical data
 - Dataset file size & Character variable length
 - SAS® programming topic: Call Execute statement

The Context: Electronic Submission of Clinical Data



3.3.3 Dataset Column Length

The allotted length for each column containing character (text) data should be set to the maximum length of the variable used across all datasets in the study. This will significantly reduce file sizes. For example, if USUBJID has a maximum length of 18, the USUBJID's column size should be set to 18, not 200.

Dataset File Size and Length of Character Variables

- The correlation between file size and the length of variables can be demonstrated by the following simple empirical experiment.

```
data WORK.TEST1 (label='Test1 Dataset') ;
  length MyKey 8. VarA VarB VarC VarD VarE VarF $ 2 ;
  delete ;
run ;
```



test1.sas7bdat

5 KB

```
data WORK.TEST2 (label='Test2 Dataset') ;
  length MyKey 8. VarA VarB VarC VarD VarE VarF $ 200 ;
  delete ;
run ;
```



test2.sas7bdat

17 KB

```
data WORK.TEST3(label='Test3 Dataset') ;
  length MyKey 8. VarA VarB VarC VarD VarE VarF $ 200 ;
  label MyKey='Identifier' VarA='Parity' ;
  do MyKey=1 to 50000 ;
    if (mod(MyKey,2)) then VarA="Uneven" ;
    else VarA="Even" ;
    output ;
  end ;
run ;
```



test3.sas7bdat

61 553 KB

```
data WORK.TEST4(label='Test4 Dataset') ;
  length MyKey 8. VarA $ 6 VarB VarC VarD VarE VarF $ 1 ;
  label MyKey='Identifier' VarA='Parity' ;
  do MyKey=1 to 50000 ;
    if (mod(MyKey,2)) then VarA="Uneven" ;
    else VarA="Even" ;
    output ;
  end ;
run ;
```



test4.sas7bdat

1 197 KB

➔ File size reduced by more than 98% without losing any information.

SAS Programming Topic: Call Execute statement

- ❑ Call Execute allows using the values in a SAS dataset as parameters of a macro.
- ❑ The process generated and executed by Call Execute will be repeated from every observation of the input dataset
- ❑ In a new document released in June 2015, the CDER recommends that the length "should be set to the maximum length of the variable used **across all datasets in the study**".
- ❑ Now when a variable is in more than one dataset the length should not be set to the maximum length of the variable in the dataset but **across all datasets in the study**
 - Using COLUMN dictionary we create call execute input dataset
 1. Identification of variables existing in more than one dataset
 2. Keep a row for each dataset/variable with a length to modify

	Library Name	Member Name	Column Name	LENGTH
1	WORK	DS	VISIT	6
2	WORK	DV	VISIT	6

SAS Programming Topic: Call Execute statement

- Using Call Execute functionalities we modify the length of variables identified by each row of the input dataset

```

%** Resize variable using Call execute and SQL procedure
** ;
data _NULL_ ;
  set WORK.&Tmp.TODO ;
  length COMMAND $ 1000 ;
  COMMAND=catx(' ', "proc sql ;alter
table", cats(LIBNAME, '.', MEMNAME), "modify", NAME, cats("char(
", put(LENGTH, best12.), ")"), ";quit ;") ;
  call execute(COMMAND) ;
run ;

```

NOTE: CALL EXECUTE generated line.

```

1  + proc sql ;
1  +           alter table WORK.DS modify VISIT char(6) ;

```

NOTE: Table WORK.DS has been modified, with 1 columns.

```

1  +                                           quit ;

```

Overall summary

- ❑ Why do you have to adjust the length of variables ?
 - ❑ Because it is recommended by FDA
 - ❑ To save place and to not have to split your dataset
- ❑ Why do you, SAS programmers or users, have to keep in mind this relationship between dataset file size and variable length ?
 - ❑ To reduce file size and storage costs
 - ❑ To resolve resource limitations
 - ❑ And to facilitate completing long SAS jobs
- ❑ This SAS package for adjusting the lengths of character variables to fit the longest value may be the solution needed for many users wishing to reduce dataset file size.
- ❑ Link: [Study Data Technical Conformance Guide](#)

Questions?



CONTACT INFORMATION

Mickael Borne

Work Phone: +33 1 42 86 64 57

Email: mborne@4clinics.com