PharmaSUG 2014 - Paper CC37

Reducing Variable Lengths for Submission Dataset Size Reduction

Sandra VanPelt Nguyen, inVentiv Health Clinical

ABSTRACT

The FDA has cited dataset size as one of the issues they commonly encounter for submitted clinical trial datasets and has found the allotted column variable lengths to have a high correlation to overall dataset size. Based on this analysis, the reviewing divisions have requested sponsors to reduce variable lengths to the minimum lengths needed to accommodate the values found within each variable. Since it may be difficult to identify or predict up front the longest potential value for every variable, not to mention that sponsors will want to avoid risk of truncation, this paper presents a macro which identifies the minimal lengths needed based on the actual data values which are present in a dataset and reassigns variable lengths accordingly as a post-processing step prior to submitting datasets to a regulatory agency.

INTRODUCTION

Recent industry guidance has urged sponsors to minimize dataset size by keeping variable lengths to the minimums needed to accommodate data values. Sponsors and the vendors they utilize must then determine not only an appropriate method to achieve this but also appropriate timing for setting or changing the variable lengths to meet this guidance. Care and caution should be taken to implement this in such a way that does not increase risk of truncating values and does not lead to additional effort or unnecessary rework.

Dataset size issues are linked closely to use of V5 SAS® Transport format currently required for submitting clinical trial datasets to the FDA as well as with the use of CDISC data standards, however reducing the variable lengths in order to help reduce dataset file sizes is a general "best practice" that can be applied to any dataset and file format.

Although the FDA has over the years been able to accept increasingly larger file sizes, the continued growth in dataset sizes has not only slowed down ease of review via various applications and tools but additionally poses risks in terms of stressing and overloading data management systems.

HISTORY

In 2011, FDA's Center for Drug Evaluation and Research (CDER) released a document[1] describing several commonly seen issues in datasets which had been submitted in standard format (i.e. CDISC). In addition to describing several common misunderstandings or misuses of the standards, the agency also included several requests to help them conduct their reviews as well as a section on general file size issues. Prior to this, there was only limited guidance from FDA regarding dataset file size. The Study Data Specifications[2] documents prior to 2012 only gave indication of maximum file size and some limited information on dataset splitting.

This dataset file size section in the common issues document directly requested the "allotted character variable length/size for each column in a dataset should be the maximum length used" in order to "significantly reduce dataset file sizes". This or similar text has since been added to the Study Data Specifications and was included in amendment 1 to SDTM Implementation Guide version 3.1.2[3].

Analysis conducted by CDER in 2011[4] using 20 randomly selected studies received during 2010-2011 (over 400 total datasets) showed significant file size differences due to:

- Dataset format (.jmp, .sas7bdat, and .xpt, in order from smallest to largest)
- Use of CDISC standards (legacy, ADaM, and SDTM, in order from smallest to largest)
- Number of records
- Column size (variable length)

As V5 SAS Transport format was still the required format for dataset files (and moving to another format has been under discussion), moving away from CDISC data standards would not be a move in the desired direction, and number of records was tied with use of CDISC standards as well as clinical trial patient populations, the "quick fix" to mitigate dataset file size issues was to reduce column sizes, which conveniently had a very direct and strong correlation to overall dataset file size:

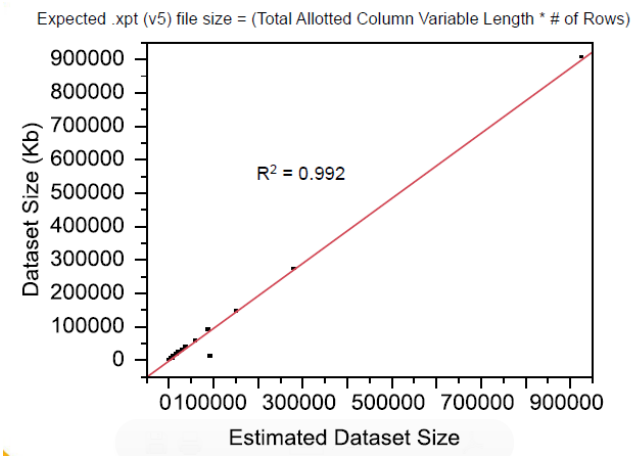


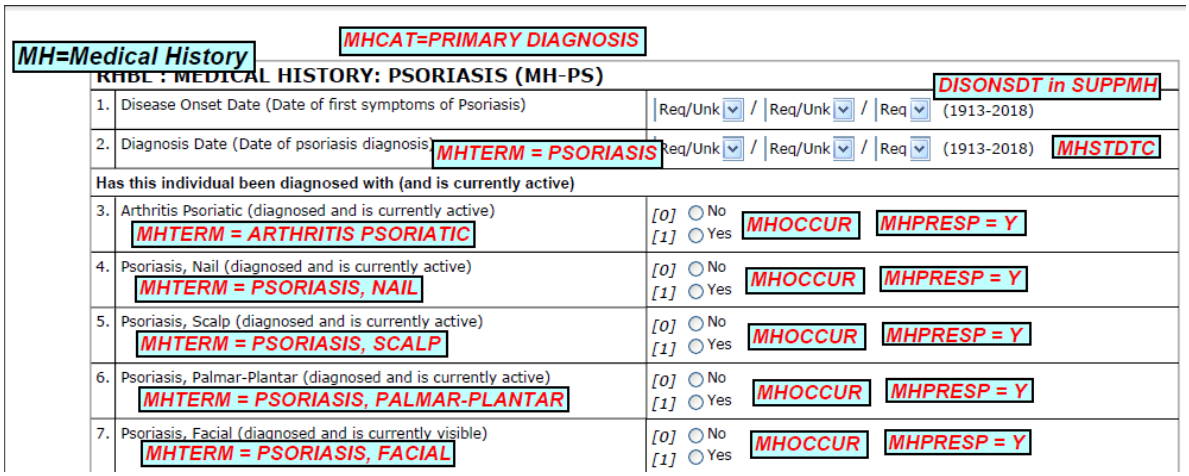
Figure 1: CDER Research: CDISC Submission Dataset Sizes[5]

After testing column size reduction in 20 studies, CDER found an average 70% overall (study-level) reduction in dataset file size. The impact is so significant that in a presentation in 2012[6], the FDA cited "Waste of Space" as issue #1 of their top 7 issues.

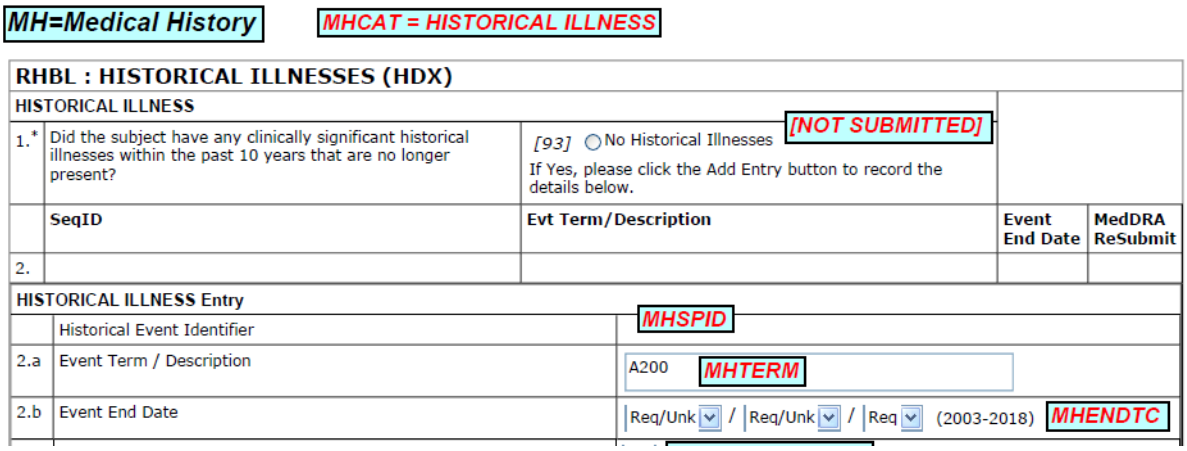
IMPLEMENTATION

So now that that column size reduction has specifically been requested by the agency (and is checked by OpenCDISC[7]), one must decide how and when to implement this within the dataset generation process. Variable lengths are typically designated up front within dataset programming (or mapping) specifications. When setting up the specifications, the "spec" writer should not default the length of a character variable to $200, the maximum allowed for version 5 SAS transport file format and for most variables within CDISC standards, but should base the length from the expected values to be mapped to each variable. For example, variable EGBLFL (ECG baseline flag in SDTM) will only contain values 'Y' or null, so that variable's length can be set as $1 up front.

Many variables will have CDISC controlled terminology[8] available which can be evaluated for the longest possible value in order to assign an appropriate variable length. Other variables may not use CDISC controlled terminology but will have a finite possible list of values which can similarly be reviewed up front in order to assign variable length.



In the CRF above, the maximum variables lengths needed for MHCAT, MHTERM, MHPRESP, MHOCCUR, and MHSTDTC can all be pre-determined based on the pre-specified values or formats in the CRF. In cases where possible values are not pre-specified or known, such as for free text fields (see MHTERM in the example below) or in external data, estimating the maximum length needed up front results in risk of truncated values, and defaulting to $200 goes against the FDA request and may result in unnecessarily large file sizes, so another approach must be considered.



CONSIDERATIONS

One needs to evaluate the mapping process and method for assigning the variable lengths in order to have the least impact and risk and minimize effort, rework, and issues. There are several questions to consider:

- At what point in the mapping process should the variable lengths be determined?
  - Standards/metadata library
  - Study-specific programming specifications
  - Mapping programs
  - XPT file generation
- At what point during the study should the variable lengths be finalized?
  - Initial programming
  - Following each data generation cycle
  - Database lock
  - Regulatory submission
- What method should be used to assign the variable lengths?
  - Hard coding
  - Automated from metadata library or programming specifications
  - Automated based on the data

OUR SOLUTION

To allow maximum flexibility but minimal effort, we decided to reduce variable lengths via a SAS macro as a post-processing step, which would determine in a dynamic fashion the maximum length needed for each variable based on the actual values present in the dataset. By taking this approach, we would not have to continuously monitor new data coming in to check for longer values and repeatedly update programming specifications and programs. We could assign the maximum possible length (up to $200) for each variable in our programming specifications up front and then allow the macro to reset the variable lengths at the end of each dataset generation cycle. This also reduced risk of truncation of values and the reduced dataset file size would be in place at any point in the study. A quick PROC COMPARE is used to verify that the processed, reduced-size datasets are no different than the original datasets apart from variable lengths.

Step 1: SAS dictionary tables can automatically determine the datasets and variables for your study:

```
proc sql noprint;
       create table DSETS as
              select MEMNAME label="Dataset Name",
              MEMLABEL
              from dictionary.tables
              where LIBNAME = "SDTMDATA"
              order by MEMNAME;
       create table VARLIST as
           select MEMNAME label="Dataset Name",
              NAME label="Variable",
              TYPE,
              LENGTH,
```

```
              VARNUM
              from dictionary.columns
              where LIBNAME = "SDTMDATA";
quit;
```

Step 2: Loop through the datasets and variables using PROC SQL and LENGTH function to determine the maximum length present in each dataset/variable combination:

```
   %if &vartype = char %then %do;
      proc sql;
        create table temp as
        select "&dset" as dataset,
               "&varname" as variable,
                 "&vartype" as type,
                 max(length(&varname)) as maxlen
        from sdtmdata.&dset;
      quit;
   %end;
```

*Note that for character variables which are not populated (all values are null), then the length will be set to $1.

**Numeric variables are not reduced.

Step 3: Gather the maximum length for each variable and reset the variable lengths in each dataset:

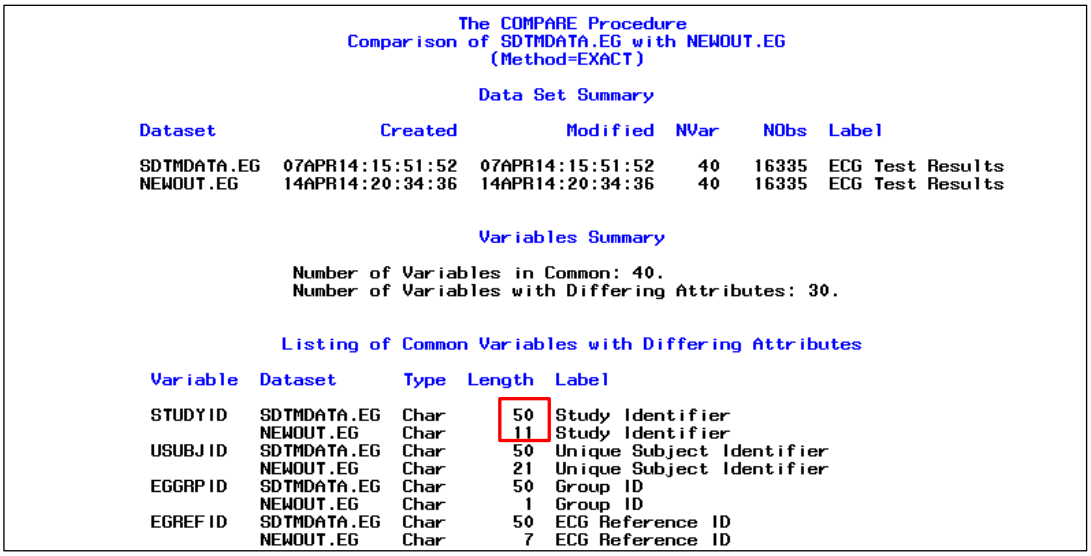
```
       data all_len;
         set all_len (where = (dataset = "&dset"));
         if type = 'char' then newlen = '$' || trim(left(maxlen));
         else newlen = trim(left(put(maxlen,8.)));
         templen = trim(variable) || ' ' || trim(newlen);
       run;

       proc sql noprint;
         select templen
         into : alllength separated by ' '
                from all_len;
       quit;


       data newout.&dset (label = &dslbl);
         length &alllength;
         set sdtmdata.&dset;
       run;
```

Step 4: Use PROC COMPARE to verify no unanticipated changes occurred:

```
    proc compare base = sdtmdata.&dset compare = newout.&dset listall;
      title "Compare - &dset";
   run;
```



The COMPARE Procedure
Comparison of SDTMDATA.EG with NEWOUT.EG
(Method=EXACT)

Data Set Summary

Dataset	Created	Modified	NVar	NObs	Label
SDTMDATA.EG	07APR14:15:51:52	07APR14:15:51:52	40	16335	ECG Test Results
NEWOUT.EG	14APR14:20:34:36	14APR14:20:34:36	40	16335	ECG Test Results

Variables Summary

Number of Variables in Common: 40.
Number of Variables with Differing Attributes: 30.

Listing of Common Variables with Differing Attributes

Variable	Dataset	Type	Length	Label
STUDYID	SDTMDATA.EG	Char	50	Study Identifier
	NEWOUT.EG	Char	11	Study Identifier
USUBJID	SDTMDATA.EG	Char	50	Unique Subject Identifier
	NEWOUT.EG	Char	21	Unique Subject Identifier
EGGRPID	SDTMDATA.EG	Char	50	Group ID
	NEWOUT.EG	Char	1	Group ID
EGREFID	SDTMDATA.EG	Char	50	ECG Reference ID
	NEWOUT.EG	Char	7	ECG Reference ID

Output 1: PROC COMPARE results (partial)

CONCLUSION

Given the strong correlation between variable lengths and overall dataset size, it only makes sense to take efforts to reduce variable lengths wherever possible, but with this request applying across studies and with potential risks for truncation or rework, it additionally makes sense to approach this using a process which is as dynamic and automated as possible. Reducing the variable lengths can be implemented in more than one way. The code presented in this paper is just one example and there are other methods which have been publicly shared as well. I encourage you to research and evaluate various methods and consider what will work best within your process while minimizing risk/impact to data as well as effort involved and potential for rework during the project. The SAS code in our solution can easily be modified to exclude particular variables from length reduction (e.g. ARMCD, --TESTCD, --TEST which have defined maximum lengths in SDTM), to look across datasets for determining maximum lengths (such as when a domain has been split), or look across studies for determining maximum lengths (useful when integrating data), as well as incorporate SAS Transport file generation. Implementing dataset file size reduction is not only important for regulatory submissions, but can also help reduce system disk space requirements within your daily work environment.

REFERENCES

[1] "CDER Common Data Standards Issues Document." *CDER*. December 2011. Available at http://www.fda.gov/downloads/Drugs/DevelopmentApprovalProcess/FormsSubmissionRequirements/ElectronicSubmissions/UCM254113.pdf.

[2] "Study Data Specifications." *FDA*. Available at http://www.fda.gov/downloads/ForIndustry/DataStandards/StudyDataStandards/UCM312964.pdf.

[3] "Study Data Tabulation Model Implementation Guide 3.1.2." *CDISC*. Available at http://www.cdisc.org/sdtm.

[4, 5] Chhatre, Dhananjay. "SDTM Column Resizing: Background and Industry Testing Results." Available at http://www.cdisc.org/stuff/contentmgr/files/0/4f05d8426369051905a247002c87e38e/files/dhananjay_chhatre__session_9.pdf.

[6] Chattre, Dhananjay; Malla, Amy. "CDER/CBER's Top 7 CDISC Standards Issues." *FDA*. Available at http://www.fda.gov/downloads/Drugs/DevelopmentApprovalProcess/FormsSubmissionRequirements/ElectronicSubmissions/UCM291752.pdf.

[7] OpenCDISC. Available at www.opencdisc.org.

[8] CDISC controlled terminology. Available at http://www.cancer.gov/cancertopics/cancerlibrary/terminologyresources/cdisc.

RECOMMENDED READING

- PhUSE Data Sizing Best Practices Recommendation (http://www.phusewiki.org/wiki/index.php?title=Data_Sizing_Best_Practices_Recommendation)

CONTACT INFORMATION

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

Sandra (Sandy) VanPelt Nguyen
inVentiv Health Clinical
sandra.vanpeltnguyen@inventivhealth.com