

Using SAS® to Create Statistical CANDAs (SCANDA) Datasets from Clinical Trial Data

Rocco Brunelle, Eli Lilly and Company
Debbie Romjue-Bailey, Eli Lilly and Company
Bill Huster, Eli Lilly and Company
Michelle McNabb, Software Synergy, Inc.
Sharon Symanowski, Eli Lilly and Company
Ted Shaw, Eli Lilly and Company
Linda Bergin, Eli Lilly and Company
Kathy Koskowicz, Eli Lilly and Company

Abstract

The Food and Drug Administration (FDA) has stated that all New Drug Applications (NDA's) will have a computerized review by 1995. Consequently, all pharmaceutical companies are preparing computer assisted NDA's (CANDA's). This computerized submission aids FDA reviewers and often shortens the time of the review. Similarly, the Statistical Evaluation and Research Branch of the FDA often requests a Statistical CANDA (SCANDA) which includes the data and the computer code used to perform the statistical analysis. SCANDA's are typically an afterthought, constructed after the NDA is complete and validated with the results in the NDA. Because of this, the SCANDA's are often inadequate. The FDA's CANDA Guidance Manual (1992) stresses the need for *a priori* attention to the structure of clinical databases and the accessibility of suitable data subsets.

This paper reviews our experience in defining and creating a SCANDA database as SAS® datasets. The format and structure of these SAS SCANDA datasets are such that they can be used to easily generate the summary reports and analyses that comprise the NDA. Additionally, these datasets can be given to the FDA to facilitate the statistical review.

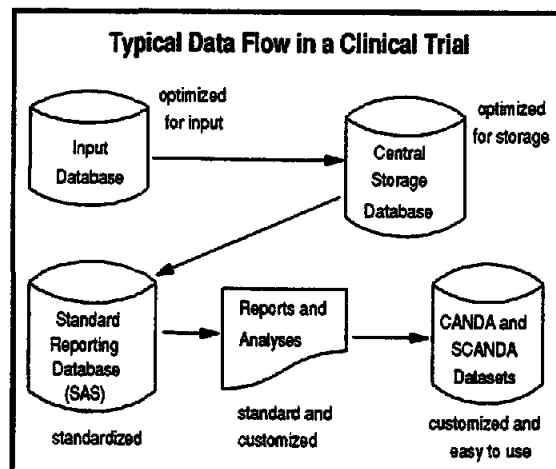
Introduction

In order to market a new medical therapy, one must perform many rigorous clinical trials evaluating safety and efficacy. After the trials are completed, a registration document is submitted to the appropriate governmental agencies. In the US, a New Drug Application (NDA) is sent to the Food and Drug Administration (FDA).

The data in a clinical trial originates with the protocol and case report forms. The protocol is a detailed document specifying the objectives and methods for the collection and analysis of the clinical trial data. In the pharmaceutical industry, the clinical trials typically compare a new treatment to one or more standard therapies. The case report form is a document for the investigators or the patients to record the clinical trial measurements. The case report forms are returned to the sponsoring agency and the data is entered into a computer. Recently, paper case report forms are being replaced with computers which can electronically send the clinical trial data to the sponsor.

After the clinical trial data is collected, it is often uploaded to a large database. Upon completion of the study, various reports and analyses are performed and a clinical report is written.

Typically, four databases are constructed during the course of a clinical trial. First, an input database is created which is optimized for data entry. This system is set up to evaluate the data as it is entered in order to identify suspicious values.



®denotes a registered trademark for USA registrations

The second database is usually a central storage database which contains data from many clinical trials in a standardized format. This central database is often designed for long-term storage and is thus optimized for cost efficiency.

A third database is often needed to put the data in a standardized system that can be used for reporting and analysis. The data is often put into a SAS library which is the standard database used in the pharmaceutical industry. This database contains all the information from a particular clinical trial but it is usually not optimized for analysis and reporting. Typically, the data is put into many small SAS files with the only duplicate information being the patient's identification information. Often it is difficult to interpret what data resides within each of the SAS files. Also the variable names and labels are usually difficult to understand. It often takes many lines of SAS code at the beginning of a program to prepare the data for listings and statistical analyses.

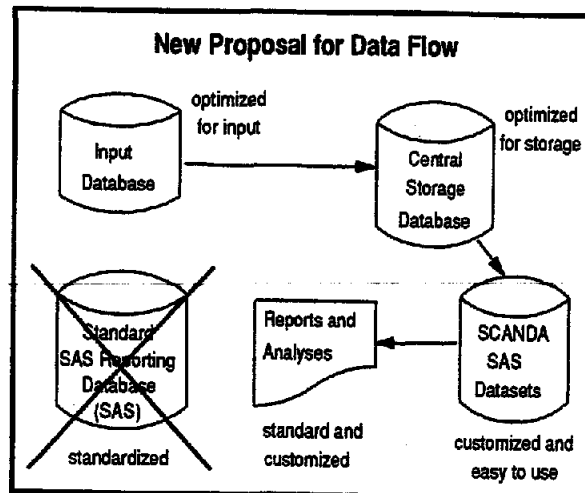
After the data is analyzed and the clinical report is finished, a fourth database is often created which will be sent to the FDA. The medical reviewer at the FDA often requests the clinical trial data in an electronic form that will aid his or her review. This is called a CANDA (Computer Assisted New Drug Application). A CANDA is useful as an aid in reviewing large, complex reports and in examining large amounts of data. The FDA guidelines recommend the use of CANDA's and states that all NDA's will have CANDA's by 1995.

Besides submitting the data as a CANDA, the statistical branch of the FDA often requests all the data that was collected in a clinical trial in electronic form along with the code that was used to create the reports and analyses. A SCANDA (Statistical CANDA) is put together which includes a SAS library of all the clinical trial data, the SAS code used to create the reports and analyses and the final report in electronic form.

The CANDA's and SCANDA's are usually customized for each clinical trial in order to make it easy for the FDA to use and, hopefully, speed up the review process.

Our proposal is to construct the SCANDA's and CANDA's earlier so they can be used by both the statisticians and systems analysts responsible for the final report and for the FDA. Our paper focuses on the early development of the SCANDA; however,

many of the same concepts will apply to the early development of a CANDA.



Objective

The objective of a SCANDA is to create an optimized database to meet the reporting and analysis needs for the NDA and other registration requirements. The SCANDA database should be easy to use by in-house statisticians and systems analysts, and the statisticians at the FDA. Also, the SCANDA should be sufficiently standardized so that one can use in-house standard reporting programs.

These datasets should be designed in such a way that they anticipate the reporting and analysis needs. They should reduce the number of merges required for analysis, store variables that will be analyzed together in the same datasets and have derived and summarized variables ready for analysis.

Many regulatory agencies, including the FDA, have detailed guidelines for the reporting and analysis of data from clinical trials which can be used in the design of a SCANDA. The clinical report includes listings of all the data collected in the clinical trial, summary tables of the primary and secondary efficacy and safety measurements, and tables of the analysis results. The listings should include identification variables such as project, investigator number, patient number, treatment group and visit. Summarization's should be made by treatment group and visit, and the analyses typically compare the treatment groups at each visit.

Analyses are also conducted for selected derived and summarized parameters. For example, a study in a drug to treat hypertension may have multiple blood pressure measurements at each visit which are averaged for analysis.

Also, the various regulatory agencies require subgroup analyses. Subgroup analysis evaluate the treatment effects for various demographic subgroups that can be affected by the study treatments. For example, the subgroups can be gender (males and females), race and weight.

Design

There are three main points to consider when creating a SCANDA:

1. SCANDA Users
2. Requirements Document
3. Database Implementation

Input is needed from everyone that will either use this SCANDA database or will influence the reports and analyses. The primary group should include the systems analysts, statisticians, physicians and the paramedical personnel responsible for conducting the trial. Additionally, the group can include medical writers, individuals from health economics and marketing, and other individuals from areas that may use the data in the SCANDA.

The next step is to put together the requirements document. This is a detailed document defining the elements and structure of the SCANDA database.

First, there should be separate SAS datasets for different types of data. For example the SCANDA might have the following datasets:

- Efficacy Dataset
One record for each patient and visit
- Adverse Events Dataset
One record for each adverse event
- Dosage Information Dataset
One record for each patient and visit
- Habits Dataset
One record for each patient (eg, smoking and alcohol use)

Additional SAS datasets may be needed in the SCANDA database. For example, an investigator dataset could include one observation for each

investigator such as the investigator's name and address. Also, a study dataset could be useful. This could include just one observation containing the date the SAS library was updated, the title of the study, as well as other study specific information.

The exact structure of the SCANDA datasets should also be defined in the requirements document. Each SAS dataset should have global variables and specific variables. The global variables include the patient identification variables and the subgroup variables. The specific variables include the original measurement variables as well as summarized and derived variables.

The structure of the variables should also be carefully documented. The variable names should be carefully chosen so that they are easily understood by everyone involved in the project. Also, the variable labels should be very specific and well defined. The storage length for character variables should be set to the length of the longest possible value. For numeric variables, we suggest one use the SAS default storage length.

Variable output formats should also be predefined. Often there are standard output formats for specific variables which are useful when listing the data. For example, the variable AGE at the start of the study, which is computed from the study start date and the date of birth, could have a predefined output format of 5.1. Often, one can use the case report form as a reference to determine good output formats.

Finally, the variables should contain values that make it easy for the user to interpret and one should try to minimize the use of codes. For example, the variable SEX should contain the values "Male" and "Female", or "M" and "F", instead of codes 1 and 2.

Below is an example of a Habits Dataset within the SCANDA Database.

Habits File

	Variable Name	Label	Output Format
ID	PROJ	Project Code	\$8
	INV	Investigator Number	\$6
	PATIENT	Patient Number	5.0
	TRT	Treatment	\$12
Subgroup	AGE	Age in Years	5.1
	SEX	Sex	\$6
Habits	SMOKING	Patient Smokes?	\$1
	ALCOHOL	Patient Uses Alcohol?	\$1

It is acknowledged that these SCANDA datasets contain a great deal of duplicate data. However, this structure aids in creating reports and performing analyses.

For example, the following SAS code,

```
PROC PRINT DATA=libname.EFFICACY ;  
RUN ;
```

will produce a logical listing of the data within the EFFICACY dataset. Notice that this procedure did not need the use of VAR or FORMAT statements.

One can easily produce a fancy listing with better labels by using the following SAS code:

```
PROC PRINT DATA=libname.EFFICACY LABEL ;  
RUN ;
```

The order of the data within the datasets should be considered. The variables in each of the SCANDA datasets should appear in a predefined order. The ID variables should be first, followed by the subgroup variables and the data specific variables. The end user should know where to look to find a specific variable in either the SAS dataset or in a simple listing of the data. Also, the observations in each of the datasets should be presorted in a logical, predefined order.

The SCANDA now is more than a database - it has become information. It is also easier for the end user to construct reports and perform analyses.

One last point is that the SCANDA is not static. It is a dynamic database. One should expect new variable definitions especially for subgroup, summarized and derived variables right up to the

writing of the final report. Often, the analysis uncovers the need to summarize the data in new ways. However, most of the structure in the SCANDA's can be defined before the data is reported and analyzed.

Implementation

The systems analysts responsible for creating the SCANDA SAS library need to have a good understanding of the clinical trial and the data being collected. They should also be familiar with the structure of the central storage database.

The systems analysts first need to construct logical mappings of the elements in the central storage database to the SAS SCANDA library. Also, they must write and test the code to create the SAS SCANDA's. Finally, they should spot check the SCANDA data and compare it with the original clinical trial data. One way to do this is to randomly select a few patients and then carefully check all of their data.

The structure of the SAS program that creates the SCANDA's should be comprised of macro units. One macro should exist for each SAS dataset defined in the requirements document. (See Example of Macro Units on MVS.)

The SAS dataset macros include global macro calls, dataset specific information, and the summarized and derived variables. The global macro captures the global variables which are common across all the SCANDA datasets. The global variables macro insures consistency of variable names, variable labels and variable formats. Also, this global macro allows for easy maintenance of the SAS code. (See Example of Dataset Macro.)

Conclusion

A well designed reporting and analysis SAS library is not only useful to the FDA and other regulatory agencies to speed the review process of a new drug application, but it is also very useful to speed the reporting and analysis of the study results. The same well design SAS library can be used by many different areas to perform listing, summarizes and analyses.

Example of Macro Units on MVS

```

/JOBNAM JOB(ACCT#),.....
/.....
/SASSTEP EXEC SAS6,OPTIONS='MAUTOSOURCE'
/SASAUTOS DD DSN=A.X.SASMACRO,DISP=SHR
/.....
/IN DD DSN=WW.SAS.UU,DISP=SHR
/OUT DD DSN=WW.SAS.YY,DISP=SHR
/.....
/SYSOUT DD DUMMY
/SYSIN DD *
/.....
* ADVERSE EVENTS
/.....
%EVENTS(INPUT=IN.EVTTBL,OUTPUT=OUT.EVENTS);
/.....
* PATIENT SUMMARY
/.....
%SUMMARY(INPUT=IN.PATSUM,OUTPUT=OUT.SUMMARY);
/.....
* DOSAGE
/.....
%DOSE(INPUT=IN.THERDS,OUTPUT=OUT.DOSE);
/.....
* EFFICACY
/.....
%EFFICACY(INPUT=IN.LABTBL,OUTPUT=OUT.EFFICACY);

```

References

Guideline for the Format and Content of the Clinical and Statistical Sections of New Drug Applications, U.S. Department of Health and Human Services, Public Health Service, Food and Drug Administration, Office of Drug Evaluation, 5600 Fishers Lane, Rockville, Maryland, July 1988.

CANDA Guidance Manual, U.S. Department of Health and Human Services, Public Health Service, Food and Drug Administration, Office of Drug Evaluation, 5600 Fishers Lane, Rockville, Maryland, 1992.

SAS is a registered trademark of SAS Institute Inc. in the USA and other countries.

Rocco Brunelle and Debbie Romjue-Bailey, Eli Lilly and Company, Lilly Corporate Center, Drop 2233, Indianapolis, IN 46285, voice (317) 276-7081, fax (317) 277-3220.

Example of Dataset Macro

```

%MACRO DOSE (INPUT=, OUTPUT=);
%*-----*;
%* INPUT DOSAGE DATA FROM CENTRAL DATABASE*
%*-----*;
PROC SORT DATA=&INPUT OUT=_ONE;
    BY &IDVARS;
RUN;
%*-----*;
%*MERGE IN OTHER DESIRED DATA.
%*-----*;
DATA _TWO;
    MERGE _ONE _XXX;
    ...
RUN;
%*-----*;
%* MERGE DOSAGE DATA WITH THE GLOBAL VARIABLES*;
%*-----*;
%GLBVAR(OUTPUT=_GLBS);
PROC SORT DATA=_GLBS;
    BY &IDVARS;
RUN;
DATA _FIVE (KEEP=&IDVARS VISIT THER DOSE TIME
                AGE SEX WEIGHT);
    MERGE _FOUR (IN=DOSE)
          _GLBS (IN=GLBS);
    BY &IDVARS;
    IF DOSE;
RUN;
%*-----*;
%* OUTPUT PERMANENT SAS SCANDA LIBRARY MEMBER*;
%*-----*;
DATA &OUTPUT (KEEP = &IDVARS VISIT THER DOSE TIME
                AGE SEX WEIGHT);
%*-----*;
%* ORDER OF VARIABLES IN LENGTH STATEMENT
    DETERMINES ORDER IN SAS MEMBER
%*-----*;
%*-----*
    LENGTH PROJECT $6
            INVSTGR $6
            PATIENT $8
            VISIT 8
            AGE 8
            SEX $1
            WEIGHT 8
            THER $20
            DOSE 8
            TIME 8;
    FORMAT DOSE 3.1
            TIME TIME.;
SET _FIVE;
    LABEL DOSE = 'Daily Dose'
           TIME = 'Time of Dose';
RUN;
%MEND DOSE;

```