PharmaSUG 2017 - Paper AD05

An Efficient Solution to Efficacy ADaM Design and Implementation

Chengxin Li, Pfizer Consumer Healthcare, Madison, NJ, USA

Zhongwei Zhou, Pfizer Consumer Healthcare, Madison, NJ, USA

ABSTRACT

In clinical trial data processing, the design and implementation of efficacy datasets are often challenging. The efficacy datasets here refer to the analysis data subject level (ADSL) and efficacy endpoints datasets at specific study level (e.g., ADEFF). Those two types of datasets are also recommended for FDA submission. To achieve optimal programming with efficient and reusable codes, this paper investigates some standardization methods for the design and implementation of ADSL and ADEFF datasets.

For ADSL, based on rigid SDTM common domains, and ADSL components and functions, ADSL variables are further designated into categories of **g**lobal, **p**roject, and **s**tudy (GPS). The global variables (approximately 80% of all ADSL variables) in ADSL are specified, derived, and validated only once within a company; the project variables can be further managed within a therapeutic area or an indication; and study variables are handled at specific study level. A global macro is developed to implement the ADSL processing, where the macro is called for deriving "G" variables and "P" level variables. The "S" level variables are added from study programming team. Therefore the programming team can focus mainly on study specific variable derivations.

For ADEFF, this paper introduces a two-layer ADaM design method for generating the efficacy endpoints dataset. The first layer is an interim dataset developed with timing windows and imputation rules. Then derived from the first layer dataset, the second layer is an endpoints dataset holding either binary or continuous endpoints in a vertical or horizontal structure. In each layer, the derivation flows in sequential steps; the individual steps are maximally macrotized, e.g., for the derivation of LOCF. With this approach, the complicated concepts are divided into simpler manageable steps and then assembled together and further polished (i.e., aligning metadata with specifications). The second layer dataset is used for supporting all the efficacy endpoint analyses. For traceability, it is also recommended to submit the first layer dataset.

INTRODUCTION

The Clinical Data Interchange Standards Consortium (CDISC)[1] has defined a series of data models. CDASH (Clinical Data Acquisition Standards Harmonization) is a data collection standard harmonized with SDTM (Study Data Tabulation Model). SDTM should fully reflect the collected data (e.g., mapping for any collected data and deriving a limited number of variables, but no imputation for missing data). ADaM (Analysis Data Model) should only be derived from SDTM. The key endpoint analyses, inferential analyses, and complicated analyses should be designed in ADaM datasets. However, not every analysis needs to have a corresponding ADaM dataset. Some simple tables can be directly created from SDTM. The CDISC data processing models are illustrated in Figure 1.

Traceability and analysis-ready concepts are the two core features of the ADaM design process. ADaM datasets should fully support analyses and facilitate reviews. There are several dataset structures defined in the ADaM Standards such as: Subject-Level Analysis Dataset (ADSL), Basic Dataset Structure (BDS), and Occurrence Dataset Structure (OCCDS). The efficacy dataset (ADEFF) is often designed and implemented using a BDS structure and can be challenging to program. Safety analysis datasets using

[1] http://www.cdisc.org

either a BDS or an OCCDS structure tend to be more straightforward in terms of design and implementation.

This paper summarizes the practices of efficacy dataset generations including ADSL and ADEFF, introducing ADSL generation with GPS driven method and ADEFF generation with two-layer ADaM design method, respectively. Here ADSL functions as a supporting dataset for ADEFF.

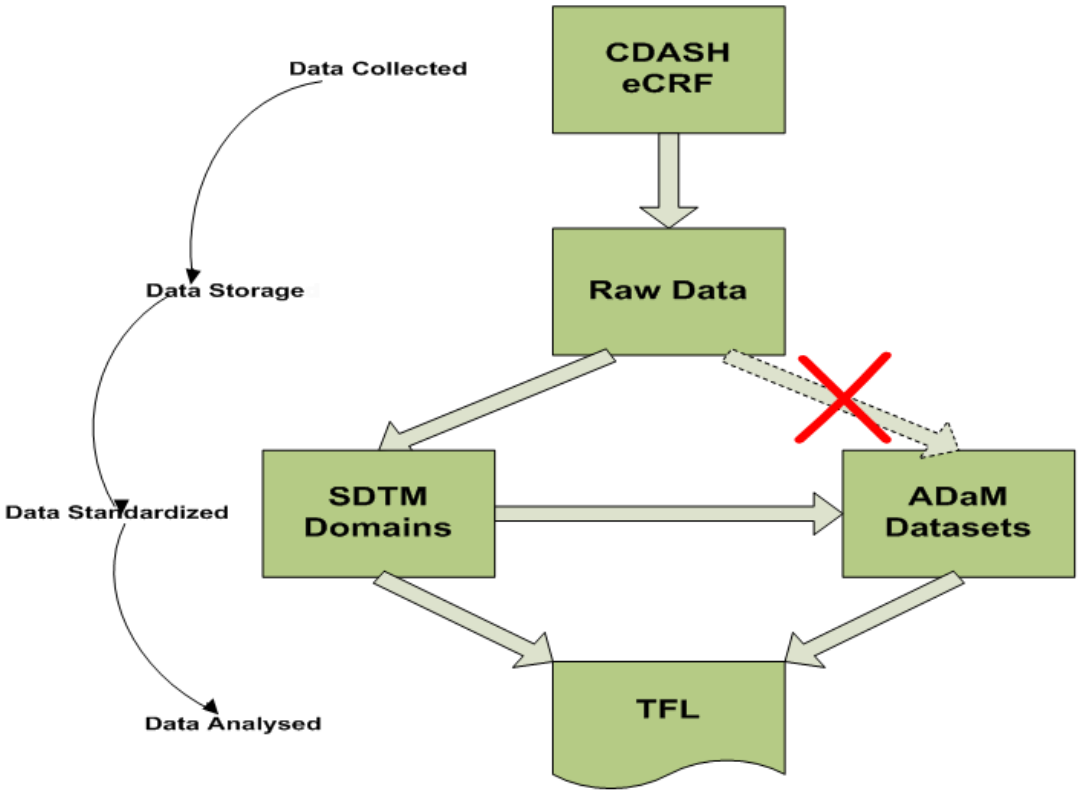

Figure 1 CDISC Data Processing Model

For illustration purpose, the trial example, if applied, is simplified as a randomized population and parallel trial design. For missing values, last observation carried forward (LOCF) approach is assumed. The SDTM QS, LB domains are assumed to derive ADaM efficacy endpoints.

The design and implementation of integrated summary of efficacy (ISE) datasets are beyond the scope of this paper.

ADSL DESIGN AND IMPLEMENTATON

ADSL dataset is a required submission dataset, structuring one record per subject and describing attributes of a subject not varying over visits during the course of a study.

ADaM Implementation Guide (IG) version1.1 specifies standard variables of subject identifiers, demographics, population indicators, treatment, trial dates, and trial level experience variables like disposition and overall compliance. Additionally, FDA Study Data Technical Conformance Guide[2] further requires important baseline subject characteristics, and covariates presented in the study protocol should also be listed in ADSL and other ADaM datasets.

[2] http://www.fda.gov/downloads/ForIndustry/DataStandards/StudyDataStandards/UCM384744.pdf

From the above components of ADSL, ADSL is able to support key subject evaluations and also provide source variables to other ADaM datasets. The key subject evaluations may include the demographic table, baseline table, disposition table, and optionally the overall exposure table and the overall treatment compliance table. In the ADaM occurrence data structure (e.g., ADAE), the denominator used for percentage calculation in the analysis is also directly summarized from ADSL.

The multiplicity of information in ADSL requires multiple domains as the sources to the ADSL. However, most variables can be directly copied or derived from common SDTM domains, e.g., DM, DS, EC/EX, and VS. Other variables such as study specific baselines, strata, covariates make the derivations more flexible. The source data may come from LB, QS, or other therapeutic area SDTM domains.

ADSL for common variables based on the common domains are formalized as “global” variables across all studies, thus specified, derived, and validated only once but used across all the studies. Approximately 80% of ADSL variables can be defined as “global”. The other ADSL variables such as indication and study specific baselines and covariates can be designated as “therapeutic area”, “project” or “study”. For instance, in Virology, numeric Baseline HCV Viral Load Value (IU/mL) is a project variable, consistently derived from non-missing LB.LBSTRESN before or on treatment start date with LB.LBTESTCD='HCVVLD'. However, the variable for Statin Usage is only used in a specific study, derived from scanning CM.CMDECOD, thus specified only at study level.

A global macro is developed to implement the ADSL “global” variables processing, and optionally, another global macro is further developed for “project” variable derivations. The “study” level variables are specifically added by the study programming team. The method is named as “GPS” navigation method. Based on “GPS” navigation, the ADSL processing flow is illustrated in figure 2.

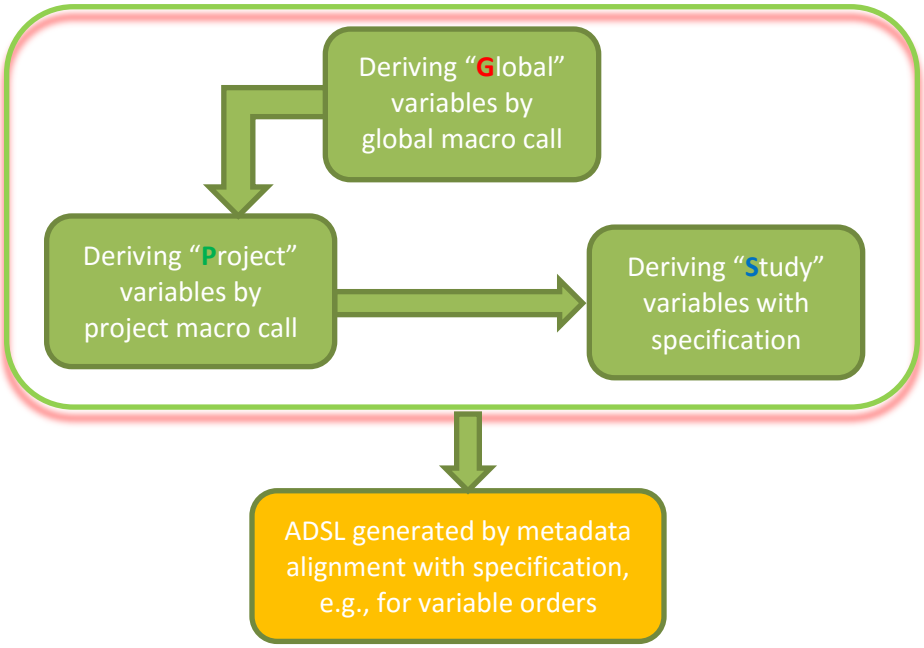

Figure 2 ADSL Generation with GPS Navigation Method

There are multiple ways to manage ADSL GPS metadata for the ADSL generation in production such as with MDR (metadata repository) or Excel sheet in DEFINE.xml style designated with “G”, “P”, and “S” in one additional column.

The horizontal structure of ADSL makes the GPS navigation method feasible and operational. It may not be applied to other vertical ADaM structures such as OCCDS or BDS. The method introduced here is still semi-automated in the sense that “S” variables are still being specifically handled by the study team.

However, from the project management perspective, the solution is simple, efficient, and easy to implement in production.

ADEFF DESIGN AND IMPLEMENTATON

In addition to ADSL, a limited number of SDTM domains as input dataset are needed to develop the efficacy ADaM dataset (ADEFF). The required SDTM domains would be commonly LB, QS, or therapeutic area (TA) specific domain(s) (mostly called SDTM efficacy domain(s)). For efficacy analyses, timing windows and imputations are widely defined in a statistical analysis plan (SAP) along with endpoints definitions. ADEFF should comply with ADaM implementation guide and agency requirements (e.g., FDA Study Data Technical Conformance Guide).

To achieve better implementation, a structured design technique, consisting in dividing a complex task or concept into several simple modules (procedures), then inter-relating those modules (procedures), should be performed. With this approach, the programming codes become easier to implement, understand, debug, and maintain. Readability with less complexity is a very important factor in programming (e.g., less macro layers, appropriate comments). Structured design facilitates readability, making the implementation easy to understand and review.

Incorporating the above efficacy ADaM design factors, one two-layer design method is developed. Figure 3 illustrates the architecture of ADaM efficacy dataset design.

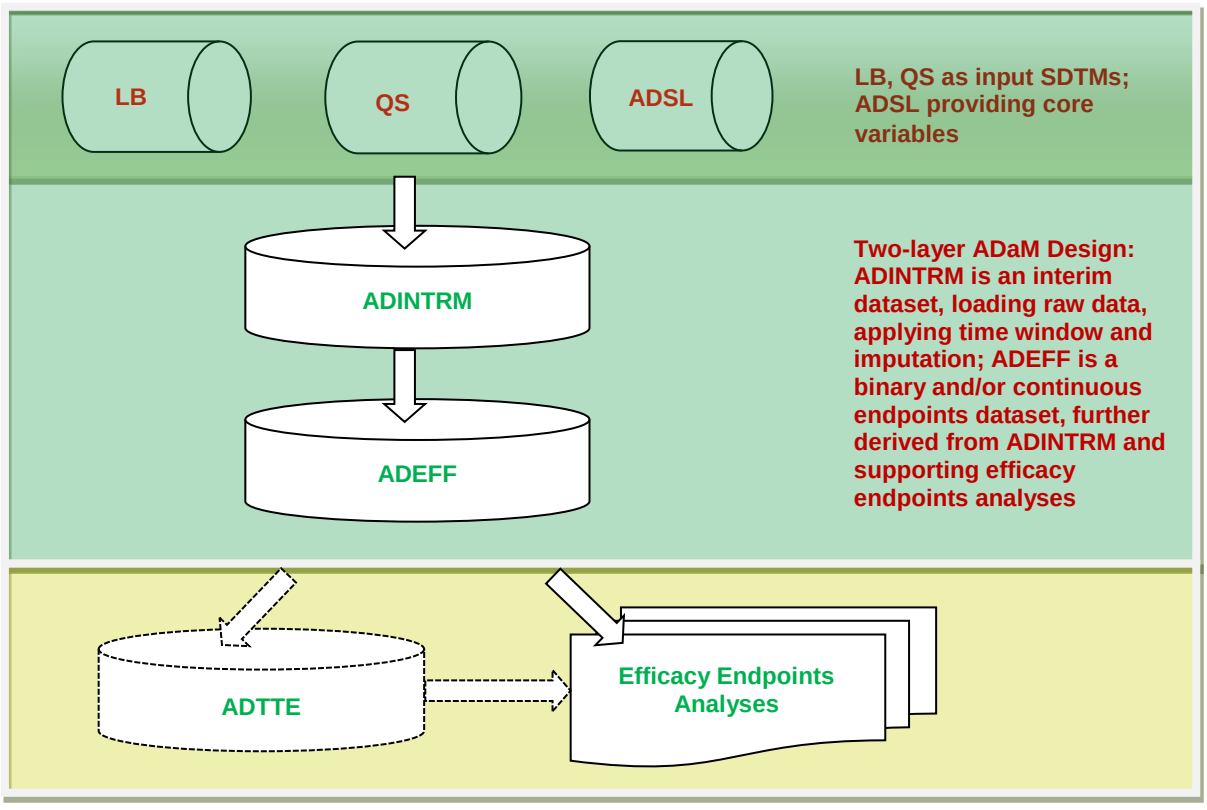

Figure 3 Architecture of Efficacy Datasets Design

ADSL functions as a driver dataset to the efficacy dataset, providing core variables of demographics and baseline characteristics, analysis population set flags, planned treatment groups, covariates, sub-groups. Even though the interim ADaM dataset ADINTRM is not designed to support any analyses, but rather support the derivations of further endpoints, ADINTRM should be submitted for traceability purpose. Additionally ADINTRM may support listings.

Depending on the analyses requested in the SAP, an efficacy ADaM time-to-event (ADTTE) dataset may be further derived from the ADEFF dataset.

Under two-layer efficacy ADaM design architecture, figure 4 depicts the sequential derivation flow of ADINTRM. With the listed steps in figure 4, the interim ADINTRM can be easily developed. Considering a randomized population set as the base in the first step (step ❶), screen failure subjects are excluded from the dataset in the beginning. The supportive core variables are included only at the final step (step ❺). With this method, the temporary datasets are "cleaner" and easier to manipulate. If a follow-up study visit comes in an on-treatment phase illogically, apply time window separately by prior to or on treatment (TRTSDT as the reference) and after treatment (TRTEDT as the reference) in step ❷. The LOCF (step ❸), which is one example of common imputation methods, is a much formalized process in programming. It can be coded as a macro. When defining a visit shell, one needs to follow the planned visit schedule defined in protocol. It is preferable to carry forward timing variables as well from previous non-missing AVAL if those variables are currently missing, e.g., VISIT, VISITNUM, VISITDY, ADT, ADY. Technically, the derivations of parameter-invariant variables (step ❹) should be put at a later programming stage.

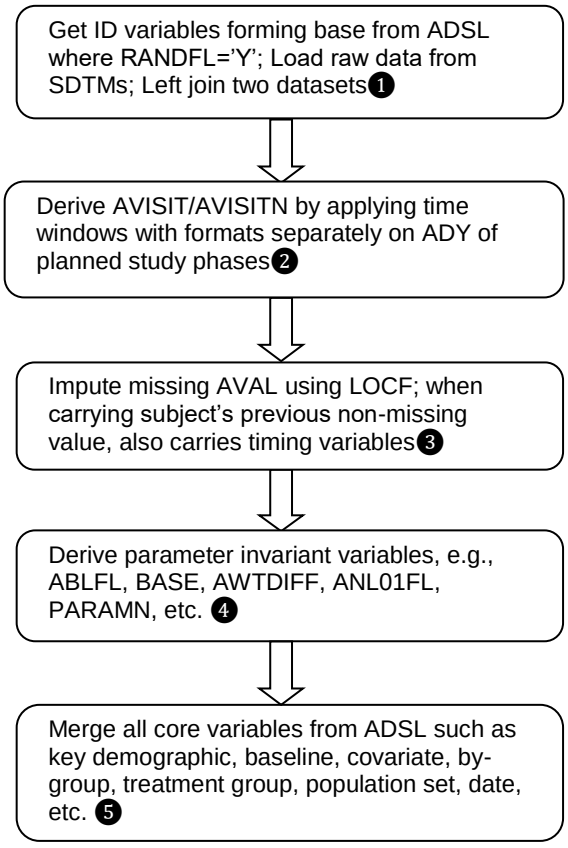

Figure 4 Derivation flow of ADINTRM

The endpoint is a function of selected items in ADINTRM. Based on ADINTRM, the generation of the endpoint dataset becomes straightforward by applying the endpoint definition to AVAL/PARAMCD in addition to copying other variables. Therefore, the processing is not repeated here. Instead, some notes are provided like the ones below.

1. Avoid the floating decimal issue for binary endpoints, e.g., using CHG=input((AVAL-BASE), best.) instead of CHG=AVAL-BASE, similarly, PCHG=input((AVAL-BASE)/BASE*100,best.) instead of PCHG= (AVAL-BASE)/BASE*100;

2. Add IMPUTFL (Imputation Flag) variable. AVAL in an endpoint dataset is the function of selected items in ADINTRM, and the imputation method specified with DTYPE is only traceable in ADINTRM. Very often, the descriptive analyses are required to be based on observed values. Therefore, the IMPUTFL variable is needed in the endpoint dataset to facilitate those analyses. The IMPUTFL can be derived as: sum (ADINTRM.DTYPE) ^='')>0) then IMPUTEFL='Y' per AVISIT per PARAMCD per SUBJECT, regardless of the VISIT.

With the macrotized derivations, the codes reusability is maximized. In the first layer dataset ADINTRM, the time windows deriving AVISIT and AVISITN can be easily achieved by utilizing the SAS® PROC FORMAT. The imputation process is highly standardized as well and can be implemented with a macro incorporating LOCF, BLOCF (baseline observation carried forward), WOCF (worst observation carried forward), or any other common methods. The derivation of the parameter invariant variable--ANL01FL, used for specific unique analysis record selection from multiple records within the same time window, is also straightforward with a macro and parameterized with the method of either closest to targeted day (e.g., VISITDY) or worst value or some other predefined method. In the second layer dataset ADEFF, the endpoint derivations can be implemented with a macro by parameterizing the input dataset name, endpoint name, (PARAM/PARAMCD), data selection conditions, and the endpoint definition. The below SAS® macro depicted in figure 5 demonstrates the implementation of binary response endpoints.

```
/*** binary endpoints derivations ***/
%macro BINEPT(INDS=, BEPT=, PARAM=, PARAMCD=, RESPDEF=, WHERE=);
  proc sql;
    create table &BEPT. as
    select distinct STUDYID,USUBJID,AVISIT,AVISITN, &PARAM. as PARAM, &PARAMCD. as PARAMCD,
           case when sum(IMPUTEFL='Y')>0 then 'Y'
                 else ''
           end as IMPUTEFL,
           case when &RESPDEF. then 1
                  else 0
           end as AVAL
    from &INDS.
    where &WHERE.
    group by STUDYID, USUBJID, AVISIT
    order by STUDYID, USUBJID, AVISITN;
  quit;
%MEND BINEPT;
```

Figure 5 Macro for Binary Endpoint Derivation

Very often the validation of an efficacy endpoint dataset requires independent double programming. The common challenge is to reach an exact match within competitive timeline. From our practices, the two-layer ADaM design method for developing efficacy dataset improves not only implementation and review, but validation as well. The two layer design allows the validation process for second layer dataset in parallel even if the first layer dataset is not a full match yet. The architecture (Figure 3) can be optionally documented in the Data Guide to facilitate reviews.

DISCUSSION

Practically, for less complexity, the ADSL implementation can be further divided by trial design types, e.g., parallel vs. cross-over. One specification and program for parallel and another one for cross-over make the implementation and maintenance simple.

It would be possible to implement an efficacy endpoint dataset with an automated concept under this two-layer design architecture. However, the multiplicity of endpoints of study by study and TA by TA would make the implementation too complex to manage. At this point, it would be sufficient just to keep the programming in the individual study level but with the same design architecture and coding structure across studies/projects with this two-layer ADaM design method.

The methods discussed in this paper are focused on the study level and exclude integrated summary of efficacy (ISE). Further investigations are needed to efficiently develop efficacy dataset for the ISE analyses using similar optimal programming philosophy.

ACKNOWLEDGEMENT

The authors would like to thank Anne LeMoigne for the paper review and invaluable inputs.

CONTACT INFORMATION

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

Name: Chengxin Li
Enterprise: Pfizer Consumer Healthcare
Address: 1 Giralda Farms
City, State ZIP: Madison, NJ 07940
Work Phone: 793 4014046
E-mail: chengxin.li@pfizer.com