PharmaSUG 2019 - Paper DS-173

Forewarned is forearmed or how to deal with ADSL issues

Anastasiia Oparii, Experis Clinical / Intego Group, LLC, Kharkiv, Ukraine

ABSTRACT

The Subject-Level Analysis Data Set is an important and essential part of each study which helps to review information about the patient across a clinical trial. Moreover, it provides traceability between all the analysis data sets and source data. Therefore, ADSL should be derived with special care. If you are really lucky, you even don't realize how many tricky questions it may cause when raw data is not clear enough or just something goes wrong.

This paper summarizes some common issues related to ADSL programming and suggests potential solutions to avoid problems in advance. In particular, it focuses on dealing with partial or completely missing dates and their connections with other derived variables along with additional validation for variables which are selected from SDTM domains. Furthermore, it walks through some useful examples and provides SAS® macros to identify issues.

KEYWORDS

ADSL, issues, Subject-Level Analysis Data Set, date-related issue, compliance, ADaM, SDTM, validation.

INTRODUCTION

The clinical trial data are collected using different sources, i.e. Case Report Forms, written questionnaires or electronic devices such as smartphones or tablets to provide data directly from patients. This variety of methods may lead to incorrect data entries and validation issues as well as a possibility to amend changes may potentially increase the opportunity for mistakes to be made. Moreover, it is not the only one reason for data issues - some of them may naturally come from incomplete dates of drug exposure, birth, start/end of adverse events, study completion/discontinuation, death date etc. which cause problems with corresponding analysis variables.

At the same time a huge number of issues appears due to incorrect CRF information handling, such as partially omitted data, misunderstanding of CRF items, a misinterpreted handwritten text, collecting data outside a required time window and collecting duplicates.

Programming issues are also a common problem in clinical studies that may occur because of different reasons. Despite numerous checks, which are usually provided by clinical trial teams during the data handling process, it is a good practice for statistical programmers to validate data before ADaM and TLG creation. While the study is ongoing, SDTM data sets are broadened, specifications often need to be changed, and furthermore, the new data may contain errors. Therefore, all these processes require special thoroughness from statisticians and programmers side to ensure that the data are complete, reliable and processed correctly. Considerable attention should be given to the Subject-Level Analysis Data Set due to its extensive use in statistical analysis.

The CDISC ADaM Implementation Guide claims that the Subject-Level Analysis Dataset (ADSL) "contains variables such as subject-level population flags, planned and actual treatment variables for each period, demographic information, stratification and subgrouping variables, important dates, etc. ADSL contains required variables plus other subject level variables that are important in describing a subject's experience in the trial." Moreover, this data set provides relationship between all the analysis data sets and the source data. Consequently, the quality of ADSL plays an important role in the outcome of the study.

The purpose of the paper is to highlight common problems with ADSL, find effective methods to ensure accurate and suitable subject-level data, improve the quality and efficiency of analysis and minimize time for proceeding the reports.

Section 1 describes how to work with the data related to incomplete or missing dates which are considered as issues. Section 2 suggests the algorithm how to deal with some special issues that occurred due to the difference between the planned and the actual treatments, features of patients' age derivation, adverse events related flags and other baseline subject's characteristics.

All assumptions and solutions are provided not for a guideline purposes, they are based on the author's experience and recommend how to deal with potential issues.

BACKGROUND. COMMON MISTAKES

In some cases, required ADSL variables appear to be missing even if other trial data say otherwise. An example of this might be a situation when a patient has received the study treatment, but is not considered as randomized in the randomized study. Or a patient has the Follow-Up period start date available, but ADSL.EOTSTT (End of Treatment Status) is missing.

Except for the various data errors, there are two main reasons for such issues. First of all, programmers have to ensure that certain criteria are met for the actual variable values used in the derivations. Despite mapping with separate code lists the following SDTM variables should be checked with a special attention while ADSL programming:

- To provide accuracy for the Dates of Randomization/Enrollment and Intent-To-Treat Population Flag variables DS.DSDECOD should be checked. The reason is that it may be equal to slightly different values such as "RANDOMIZATION" and "RANDOMIZED", "ENROLLMENT" and "ENROLLED" etc..
- It might be the case when DS.DSSCAT is equal to either "STUDY COMPLETION/EARLY DISCONTINUATION", or "STUDY COMPLETION/ DISCONTINUATION", or any other combination of these words. Non-standard DS.DSSCAT values may become a reason for the missing End of Study Date/Status/Reason.
- Numeric Date of Inform Consent variable derivation needs attention to Standardized Disposition Term value which should be equal to "INFORMED CONSENT OBTAINED" whereas Category for Disposition Event should be set to "PROTOCOL MILESTONE" and Subcategory for Disposition Event is required to contain "PROTOCOL". It is necessary to notice that generally it is equivalent to the Date/Time of Informed Consent from DM but in numeric format. In case of a partial date, no imputation is allowed to be performed, therefore RFICDT will be missing. However, talking about study specific Informed Consent flag, it should be set to "Y" based on the available character date.

The second common reason for programming issues is an incorrect handling of incomplete or missing dates and other variables depending on these dates. Statistical programmers should concentrate on converting date values from character (SDTM) to numeric (ADaM) type. According to CDISC compliance, all dates derived in ADaM data sets should be numeric. Assuming that the randomization date or the death date might be filled incompletely, they will be converted to numeric as missing and will finally disappear from the analysis. As a result, the ITT population or Death flags are most likely to be derived incorrectly. The solution of this problem may be to create the flags based not on the numeric date as sometimes described in the specification, but also to consider if the corresponding character date is non-missing.

DATES-RELATED ISSUES

Apart from the validity of date values, the order of events or assessments during the study ensures the trial data to be correct. Even though it was checked before the analysis process, statistical programmers should also keep all the date values logically consistent and minimize issues related to incorrect dates. They can face an incorrect order of the date variables which obviously indicates programming or data issues. For a proper comparison and further description all the dates during the study may be intuitively divided into the following groups:

- Dates prior to the clinical study start – Date of Birth is the most common representative of the group. It is certainly the earliest subject-related date included into the study data.

- Dates prior to the treatment start – Date of Inform Consent should be earlier than Date of Randomization or Enrollment; in some cases, Reference Start Date also could belong to this category; all baseline assessments definitely happened before intervention etc.
- Dates during the treatment period – Start/End Dates of Treatment, all Start/End Periods and Phases dates etc. which also need to be ordered logically. All the assessments and events dates collected during the treatment period belong to this category, however they are not related to ADSL.
- Dates after the treatment end – End of Study Date, Date of Death, all Follow-Up Dates, Date of Autopsy, Date Last Known to Be Alive. (Sometimes these dates are equal to End of Treatment Date, but it is convenient to analyze them in a separate category).

Looking at the categories above it makes sense to point out that any date from a certain block cannot be earlier than any date from the next block. Apparently, the end of induction date cannot be earlier than the date of randomization as well as a patient could not discontinue from the treatment before this treatment started. During the analysis phase it would be a good practice to check the order of values in date/time variables or at least the most important ones which are definitely taken into consideration further.

Besides the comparison between these categories, it is essential to check that, for example, Start of First Period Date could not occur later than End of the Second Period, End of Treatment could not be later than Study Discontinuation Date whilst Date of Death would be the latest available date per patient except for the date of autopsy or contact with relatives.

DATES PRIOR TO THE TREATMENT & DURING THE TREATMENT.

Date variables represent the major part of the clinical data, therefore it is a programmers' goal to make every effort in validating date variables. There are several dates on the start of the study which are considered to be non-missing:

The date of birth (imputed BRTHDT) is a valuable variable for the clinical study. Analysis Age variable which is directly derived from Imputed Birth Date plays a special role as a stratification factor for randomization and often defines the groups of patients to be analyzed. Nevertheless, according to some documentations and policies, the date of birth is considered to be an information lead to identification of a person and therefore it might be replaced by age. As an alternative approach it may be offered to provide only partial date of birth collected to DM.BRTHDTC, so the day and month part should be imputed. If there is no reference for such requirements and the date of birth variables exist in ADSL, they should be checked to be non-missing.

Since the SDTM demographic data set is considered to be a parent domain for ADSL, several variables which have a Record Qualifier role in DM are included to the analysis data set.

The first date, which indicates the patient's participation in the study, is the date when Inform Consent was obtained. It must follow non-missing rule if any other trial data have been collected after. The date of inform consent may be found in DS domain and moreover it is possible to have a multiple consent received during the study (in case of several study periods). For such cases ADSL.RFICyDT are provided and first of them is generally equivalent to DM.RFICDTC. As it was mentioned before, this variable should be handled with special attention to Standardized Disposition variable in order to provide validity of the study.

```
  %macro check_exist(dsin, var);

     %let dsid = %sysfunc(open(&dsin));

     %if %sysfunc(varnum(&dsid, &var)) =0 %then %do;

%put WARNING: Permissible variable %upcase(&var) does not exist in
%upcase(&dsin);

     %end;
     %else %do;
        %put NOTE: Variable %upcase(&var) exists in %upcase(&dsin);
     %end;
```

```
    %let rc = %sysfunc(close(&dsid));
  %mend check_exist;

  %check_exist(dm,rficdtc);


  data inf_consent(keep = studyid usubjid rficdt);
     set sdtm.ds;
     by studyid usubjid;
     if     upcase((scan(dsscat,1," "))) = "PROTOCOL"
        and upcase(dscat)               = "PROTOCOL MILESTONE"
        and upcase(dsdecod)             = "INFORMED CONSENT OBTAINED";

     if missing(dsstdtc) then
  put "WARN" "ING: Missing date of Inform Consent in DS. Please check USUBJID
= " USUBJID  ;

     else if length(dsstdtc) < 10 then
  put "WARN" "ING: Date of Inform Consent is partial. Please check USUBJID =
" USUBJID;

     else rficdt = input(dsstdtc,is8601da.);
  run;

  data adsl;
     merge dm          (in = indm)
           inf_consent (in = incons);
     by studyid usubjid;

     if ^indm or ^incons then
  put "USER WAR" "NING: Please check number of patients!";

     if missing(rficdtc) then
  put "WARN" "ING: Missing date of Inform Consent in DM. Please check USUBJID
= " USUBJID  ;

    else if length(rficdtc) < 10 then
  put "WARN" "ING: Date of Inform Consent is partial. Please check USUBJID =
" USUBJID;

    else if rficdt ^= input(rficdtc, is8601da.) then
  put "USER WAR" "NING: Inform Consent Date variables from ADSL and DM are
inconsistent. Please check USUBJID = " USUBJID;
  run;
```

As for Date of Randomization (RANDDT) and Date of Enrollment (ENRLDT), these dates are not allowed to have missing character values, if patient has at least one on-treatment record, as well as they should not be imputed if the respective character value is partially filled.

```
data randomiz;
    set sdtm.ds;
    by studyid usubjid;
    if upcase(dsdecod) = "RANDOMIZATION";
  /*in case of improper coding please check exact value of DSDECOD*/

    if missing(dsstdtc) then
```

```
   put "WARN" "ING: Has the subject" USUBJID = "been randomized? Please
   check!";

     else if length(dsstdtc)<10 then
   put "WARN" "ING: Date of Randomization is partial. Please check";

     else rficdt = input(dsstdtc,is8601da.);
run;
```

End of Participation date (RFPENDTC) represents the end of the subject's involvement in the study. In general, it should correspond to the last known date of contact. Although there are some examples where RFPENDTC may be the last date among all records for the subject in the database. In any case, End of Participation date should be filled if a patient has ended the study.

Reference Start Date/Time (RFSTDTC) and Reference End Date/Time (RFENDTC) variables usually display the time points when a patient was first and last exposed to the study drug, and thus they are assumed not to be missing for all randomized subjects. Generally, it is necessary to understand a time frame of the patient's involvement in the study in order to analyze the events during this period. In a certain study the start of reference period may be defined differently, such as having a washout before the randomization or a medical procedure required during a screening. In these cases, RFSTDTC is set to the enrollment date. For the studies with designs that involve many subjects who are not expected to be treated, a different protocol milestone may be chosen as a starting reference point. Only in any of the following cases RFSTDTC (as well as RFENDTC) will be null: when a subject had a screen failure (is ineligible for treatment), a subject who was enrolled but not assigned to any treatment arm or a subject who was randomized but did not receive the study treatment.

Dates related to the study treatment are widely used in the clinical trial data since it is very important to know when the medication has been administered and to calculate the duration of this treatment.

Unlike subject reference date variables, RFXSTDTC (RFXENDTC) always represents the date/time of the first (last) study exposure of protocol-specified treatment or therapy, and it is consistent to the value from SDTM.EX.

Starting from Date of First Exposure to Treatment (TRTSDT) which is derived in ADSL if there is an investigational product. At least one treatment variable is required even in a non-randomized study. To maintain consistency, the treatment values should be derived carefully and stored only once in ADSL standard variables to be used later throughout the study. Variables TRTSDT and DM.RFXSTDTC both reflect the concept of the first exposure, although TRTSDT is not required to have the same value as the DM first treatment date variable. Such scenario is possible when the first intervention is not an investigational (i.e. supportive) product. Therefore, treatment start date is not equal to the earliest date per patient in SDTM EX data set and not equal to RFXSTDTC being first date of exposure on study. The same thing might happen to the treatment end date TRTEDT. Derivations of First/Last Exposure Dates only include observations when a patient received a valid dose and date part of character variable is complete.

Dates related to the study treatment appear across the entire study including periods and phases if they are available and related to the medication. In case of multiple exposure periods or phases during the study TRTxxSDT (TRTxxEDT) with the index incremented by one should be implemented in ADSL. According to the standards it is straightforward that TRT01SDT variable has the same value as TRTSDT and TRTEDT is equal to the latest date of all TRTxxEDT. It is worth mentioning the need to check the logical order of all the period/phase treatment start/end variables.

DATES AFTER THE TREATMENT.

End of study date variables.

Except for the End of Treatment, it is highly recommended to check the End of Study Date. The definition of the end of study should be determined in the protocol. Basically it is either date of the study completion or date of the study discontinuation, date of the last visit or completion of any follow-up monitoring and data

collection or data cut-off date for interim analyses. It was mentioned above that this variable is created on the basis of DS data and some conditional selections. If a patient completed or discontinued the study regardless of the outcome this date should be non-missing and should not be imputed if it appears to be partial. It is clear that the end of study date must be later than or equal to the end of treatment date and obviously later than any other date related to the ongoing study process.

There are some specific date/time variables which require much more attention and thoroughness to derive.

DTHDTC

Working on any oncology study programmers have to take great care in order to observe accuracy and validity of death-related variables since they influence some important endpoints (e.g. Overall Survival, Event-Free Survival) and they might raise a lot of issues. Let us consider the following set of variables in details.

If DD domain is anticipated, it may be used as an additional source of Death information. Nevertheless, referring to SDTM IG 3.3., DD domain is not intended to collate data that are collected in standard variables in other domains, such as AE.AEOUT (Outcome of Adverse Event), AE.AESDTH (Results in Death) or DS.DSTERM (Reported Term for the Disposition Event). Although the death date is rarely available in SDTM DD, it will help to confirm the date of death using the information about death reasons.

According to CRF death information is collected from the Adverse Events form in case of the Serious AE resulted in death and it is going to be retained in SDTM AE domain. Besides that, it is taken from the Study Discontinuation form where “DEATH” is considered to be a reason of the Study Discontinuation which is provided in DSDECOD or DSTERM variables in DS domain. Moreover, FA data set may contain the information related to death as additional finding. Date/Time of Death (DTHDTC) and Subject Death Flag (DTHFL) are supplied to ADSL data set from DM domain and ADSL.DTHDT variable is set to the numeric date part of DTHDTC afterwards. Certainly death-related variables could not be collected with all the demographic information from CRF and in general practice they are derived by clinical programmers from AE and DS domains. It would be useful to remember that the death date from the Adverse Events data set is collected to AEDTHDTC variable while Outcome is “FATAL” and Results to Death Flag (AESDTH) = “Y” and all values of these variables have to be consistent.

```
*NOTE: Be sure that all three variables are available in SDTM.AE! Please
check using CHECK_EXIST macros;
  data ae_death;
    set sdtm.ae;
    if aeout = "FATAL" or aesdth = "Y" or ^missing(aedthdtc);

    if cmiss(aeout,aesdth,aedthdtc) < 3 then
put "WARN" "ING: Death-related variables in SDTM.AE are inconsistent.
Please check!";
  run;
```

From the Disposition data set information about death date is set as a Start Date/Time of Disposition Event (DSSTDTC) where DSDECOD = “DEATH”. If a patient died due to adverse event and the date of death comes from AE, information about “DEATH DUE TO AE” should be included in DS domain too.

Furthermore, it may occur that date of death is still missing after such derivations and consequently FA and SS domains should be additionally checked in order to cover all the related dates.

Based upon the author’s experience, there is a great number of potential issues associated with inconsistency of death information within all SDTM data sets and eventually in ADSL.

Obs	STUDYID	USUBJID	AEDECOD	AEOUT	AESDTH	AEDTHDTC	DSTERM	DSDECOD	DSSTDTC	DTHDTC
1	AA123	AA123-001-112	PNEUMONIA	FATAL	Y	2016-04-02	DEATH DUE TO ADVERSE EVENT	DEATH	2016-04-02	2016-04-02
2	AA123	AA123-001-122					DEATH DUE TO PROGRESSION OF DISEASE	DEATH	2017-11-01	2017-10-29
3	AA123	AA123-002-114					DEATH	DEATH	2018-06-22	2018-06-22
4	AA123	AA123-003-312	ENCEPHALITIS	FATAL	Y	2019-02-05				2019-02-05
5	AA123	AA123-003-416	SEPSIS	FATAL	Y	2017-04-03	DEATH	DEATH	2017-06-05	2017-04-03

Table 1. Death Date information from SDTM.AE and SDTM.DS data sets with possible issues.

Therefore some checks make sense to be implemented to ensure that the date of collection from the Subject Status data set, Findings About information as for the autopsy and any other odd dates have not caused any issue in derivation of DM.DTHDTC variable and ADSL.DTHDT as a result.

Numeric version of Death Date in Subject-Level Data Set is derived as a converted date part of the corresponding character value from DM domain. It should be emphasized that the death date is stored in a date format and time is not included. In case of partial DTHDTC, numeric variable appears to be missing (if no imputation is going to be performed). Therefore, all further analysis related to death information should be based on the character date, death flag etc.

Given that Subject Death Flag (DTHFL) is also derived from DM domain and may lead to discrepancies, it is highly recommended not to use this flag for further analysis.

To bring a traceability to the analysis in the ADaM structure, Domain and Sequence variables have been added to the Subject-Level Analysis Data Set as they are usually included in BDS structures in case of composite parameters. Supposing there is at least one observation from AE, where non-missing Death Date AEDTHDTC is available, then DTHDOM should be set to “AE” and DTHSEQ filled with the respective sequence number. Otherwise if Standardized Disposition Term DSDECOD = “DEATH” and Reported Term for Disposition Event DSTERM contains “DEATH DUE TO” then DTHDOM is set to “DS” and DTHSEQ is set to the DS sequence number.

Cause of Death (DTHCAUS) variable is connected to the DTHSEQ number and is set to AEDECOD in case of DTHDOM = “AE”, or to DSTERM value in case of “DS” source. In other scenario information about cause of death may come from DD (is equal to DDSTRESC if DDTEST = “Cause of Death”) or SUPP domains. If multiple death observations are found, the one with the earliest death date should be taken.

As mentioned before, in some clinical trials SDTM.SS may be included in investigations. In a standard way it is called Subject Status and it provides information about a patient during the Follow-Up period. From this data set programmers are able to receive patient’s “DEAD” status but not the death date value. The only date available in SS domain is SSDTC which contains the date of contact and may be collected as a call to relatives after the patient has died, so it does not have to be equal to the real death date. Be careful to check that the death date should be the latest date in ADSL and the latest among SDTM available dates except for Date of Contact in SS domain or any other dates of collection.

LSTALVDT

There are various study-specific derivation rules for Date Last Known to be Alive variable. It usually differs within the Oncology and Non-Oncology studies. First of all, LSTALVDT is a numeric variable which is derived as a combination of different dates. Although it considers either numeric or character date variables. Different approaches might be used regarding the death date in derivation of the date last known to be alive. It should be discussed beforehand and if the derivation for subjects who have died differs from the derivation for subjects who are not known to have died, the differences should be noted in the metadata.

In general, this date is set to the numeric value of the last date when a patient has been documented in clinical data to show him/her alive. But this derivation must exclude the dates of documentation, collection or the date of contact if such have been filled in.

In accordance with the existing practice, if a patient is reported to have died, LSTALVDT is equal to the date of death (or the date of death minus 1 day in rare occasions). Otherwise, different options are available. If SS domain is included in analysis, the last complete survival follow-up date with patient status 'ALIVE' is frequently used. For some kind of derivations observations with status “DEAD” are also included, but as

well as any death-related information, it should be previously described. Nevertheless, as Subject Status information may not be collected directly from a participant, there are no sufficient reasons to consider it as relevant. This approach may lead to problem when Date Last Known to be Alive appears to be later than the death date.

Obs	STUDYID	USUBJID	DTHDTC	SSTESTCD	SSSTRESC	SSDTC	LSTALVDT
1	AA123	AA123-123-111		SURVSTAT	ALIVE	2017-01-16	2017-01-16
2	AA123	AA123-134-124	2018-03-04	SURVSTAT	DEAD	2019-01-13	2018-03-04
3	AA123	AA123-141-132	2018-08-02	SURVSTAT	ALIVE	2018-09-14	2018-09-14
4	AA123	AA123-142-139	2019-04-02				2019-04-02

Table 2. Relation between SDTM.SS and Date Last Known Alive information with possible issues.

Among all the dates of assessments/events data from DS domain certainly may be used for LSTALVDT. Several derivations, which have been analyzed, state that DSTERM and DSDECOD with particular values should be considered: if available DSTERM is equal to “ALIVE” or DSDECOD is equal to “WITHDRAWAL BY SUBJECT” or “LOST TO FOLLOW-UP”. This option is reasonable since the date of consent withdrawal or lost to follow-up date are most likely to be the latest in the available subject’s data. But the fact of the successful study completion may be mistakenly skipped, since the latest date in the participant’s history is DS observation with DSSCAT = “STUDY DISCONTINUATION” and DSDECOD = “COMPLETION” or any other final DS record may be not taken into account.

Obs	STUDYID	USUBJID	LSTALVDT	DSTERM	DSDECOD	DSCAT	DSSCAT	DSDTC	DSSTDTC
1	XYZ12345	XYZ12345-1435-2222	19NOV2018	COMPLETED	COMPLETED	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION	2017-01-23	
2	XYZ12345	XYZ12345-1234-9876	29SEP2017	WITHDRAWAL BY SUBJECT	WITHDRAWAL BY SUBJECT	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION		2017-09-29
3	XYZ12345	XYZ12345-1111-1357	16MAY2017	DISEASE RELAPSE	DISEASE RELAPSE	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION	2016-12-15	
4	XYZ12345	XYZ12345-2468-1111	04MAY2017	LOST TO FOLLOW-UP	LOST TO FOLLOW-UP	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION		2017-07-30

Table 3. Relation between SDTM.DS and Date Last Known Alive information.

NOT DATES-RELATED ISSUES

In addition, variables in ADSL not only identify when the event occurred, but also provide the flags to show whether the event occurred, as well as baseline characteristics, stratification factors, the treatment received and other important information about subject’s participation in the study. Similar to date variables there is a huge number of potential issues which affect accuracy of the study data and in perspective may cost time, money and lives. Some errors might look obvious such as impossible values which are inconsistent with life signs, while others might be identified only during the data analysis being the collecting issues. There is no matter if they caused due to forgotten details, misunderstanding of questions, measurement or randomization problems, statistical programmers are responsible for high quality analysis process and have to do their best to perform proper procedures using as cleanest data as possible. To keep in mind some simple checks can help to avoid serious problems during reporting. Furthermore, understanding the source of possible errors is the half way to success.

Referring to ADaM IG 1.1. all ADSL variables may be divided into following categories:

Identifier Variables

The actual set of ADSL variables may vary but the core data items including identifier variables are required and included in all data sets across the whole study.

Study and Subject identifiers are basic variables which provide relationship between all data. For all general-observation-class domains STUDYID and USUBJID comprise the key of the data set. Study identifier connects all observations related to trial process and, moreover, plays a major role for the pooling. Each trial subject has a unique individual identifier which is the same across all study processes. Furthermore, if a subject participates in more than one study across the project, the same USUBJID should be followed across all studies. It is necessary to admit that there must be a one-to-one correspondence between USUBJID and SUBJID values. Unique subject identifier value is not allowed to have spaces and usually it concatenates study, site identifier and subject identifiers. From programmer’s side it is required to

check proper implementation and valid values of the USUBJID in all data sets to maintain correct merge and traceability among all SDTMs and ADSL as a result.

As is well known, all demographic participants' information in ADSL are based on DM data. For each USUBJID across SDTMs further checks should be additionally implemented to ensure that all subjects are included in DM and there is no extra patient received medication or had an adverse event for example. Besides that, a programmer may dig deeply and check for duplicates not only within the study, but also within the site.

Obs	USUBJID
1	ABC12345-231-9876
2	ABC12345-111-1366
3	ABC12345-UNK-9876
4	ABC12345-555-2365
5	ABC12345-001-4444

Table 4. Issue in USUBJID variable.

In addition to ID variables REGION (COUNTRY) is a part of the identification data which should be filled with meaningful and correctly coded value without typos. The good idea may be to check the values of REGION (COUNTRY) variable against the CDISC Terminology.

Demographics Variables

According to ADaM Implementation Guide, the Demographic category consists of age-related, race and sex variables.

As mentioned before, Age variable is important for statistical analysis since it is widely used as a stratification factor for the randomization. It is often considered for subgroup analysis in different reporting events and moreover it might be a covariate variable. Age Group variable plays special role for these purposes too. For example, AGEGRxx is provided for DSUR analysis, so usually ADSL contains this variable. In general, any age group variables are allowed to be added to the Subject-Level Data Set on demand.

Patient's age is collected in CRF and stored in DM domain. If possible, it is a good practice to derive ADSL variables required for analysis, because there are various reasons why pre-defined SDTM variables should be re-derived in ADaM data sets. Depending on the protocol requirements, Analysis Age (AAGE) variable is assumed to be created based on either the inform consent date, or the date of randomization, or the date of the first treatment instead of using age information from CRF for further investigations. Original variables also should be copied to ADSL. While implementing AAGE, it should be verified that age values fall into real human age interval and are non-missing. In case of missing age variable, the date of birth and the reference point should be additionally checked on completeness. Analysis Age Unit values are pre-specified in advance and AGEU generally may be equal to "DAYS", "MONTHS" or most likely "YEARS".

```
 * The following macro variables are available for optional calculation:  *;
 *     unit     - units to calculate AAGE variable                        *;
 *                default: Y                                              *;
 *                permitted values: Y M D                                 *;
 *     ref_date - reference variable to calculate age.                    *;
 *                default: RANDDT                                         *;
 *                permitted values: RANDDT TRTSDT RFICDT                  *;

options minoperator;
%macro age_deriv ( unit     =
                  ,ref_date = ) / minoperator;
```

```
  %if not(%upcase(&unit.) in (Y M D YEARS MONTHS DAYS)) %then
      %put %str(WARN ING: Impossible Units);

  %else %if not(%upcase(&ref_date.) in (RANDDT TRTSDT RFICDT)) %then
      %put %str(WARN ING: Improper reference variable) ;

  %else %do;

   /* Choose certain values according to macro options */
   %if %upcase(&unit.) = Y or %upcase(&unit.) = YEARS %then %do;
       %let factor  = 365.25;
       %let _ageu   = "YEARS";
   %end;

    %if %upcase(&unit.) = M or %upcase(&unit.) = MONTHS %then %do;
       %let factor  = 30.4375;
       %let _ageu   = "MONTHS";
    %end;

    %if %upcase(&unit.) = D or %upcase(&unit.) = DAYS %then %do;
       %let factor  = 1;
       %let _ageu   = "DAYS";
    %end;

     data adsl;
         set adsl;
         length ageu $6. aage;
         ageu = &_ageu.;

         if missing(&ref_date.) then
          put "WARN" "ING: Reference date is missing. Please check USUBJID =
" USUBJID;

         else aage = int((&ref_date. - brthdt +1)/&factor.);

       if aage ^= age then
       put "NOTE: Analysis Age and Age var from DM are different for USUBJID
= " USUBJID ;

     run;
   %end;
%mend age_deriv;
```

In ADSL as well as in any other ADaM data set there are several variables which may have limited list of values. Such as SEX variable should have values “F” or “M” and it should be remained for any conditional steps to use these options, not “FEMALE” or “MALE”. Race and Ethnic variables have bigger range of valid values. They usually vary and coded using CDISC Terminology. In general, these variables come from DM domain, but some data related to categories or clarifications may also arise from the SUPP DM data set.

Similar to age variable race, ethnic and sex are required for subgroup analysis, demographic outputs and as stratification factors. Same to AGEGR variable, RACEGR may be included in ADSL. Moreover, these variables may have numeric analogues with a one-to-one correspondence between character and numeric values.

Population Indicator Variables

There is a requirement for every clinical trial to have at least one population flag in the Subject-Level Analysis Data Set. ADaM structure gives an opportunity to add any indicator variables to define analysis populations. However, some of them are standard and they are highlighted in ADaM IG. Intent-to-treat, safety, enrolled and less often randomized, full analysis population flags are usually included in specifications. As admitted before, all these variables are derived on the basis of the SDTM data, mostly Disposition variables. For example, randomization or intent-to-treat flags results depend on the valid and meaningful values of Start Disposition Date and Standardized Term. It is important to remember that the patient's date of randomization from DS domain may appear to be incomplete. Therefore, intent-to-treat flag should be checked carefully to be sure that source variable has non-missing valid value. And vice-versa, if all indicator flags are filled with "Y", the corresponding dates are not necessary to have full and logic values.

Obs	STUDYID	USUBJID	ITTFL	RANDDT	DSDECOD	DSCAT	DSSTDTC
1	MN114	MN114-465-3488	Y	26MAR2015	RANDOMIZATION	PROTOCOL MILESTONE	2015-03-26
2	MN114	MN114-265-1156	Y	15MAY2019	RANDOMIZATION	PROTOCOL MILESTONE	2019-05-15
3	MN114	MN114-345-3233	Y	03SEP2016	RANDOMIZATION	PROTOCOL MILESTONE	2016-09-03
4	MN114	MN114-365-1265	N	.	RANDOMIZATION	PROTOCOL MILESTONE	2019-03
5	MN114	MN114-153-1763	Y	03NOV2015	RANDOMIZATION	PROTOCOL MILESTONE	2015-11-03

Table 5. Derivation of ITTFL using SDTM.DS information with possible issues.

Safety Population flag is known to be based on information about any valid dose from EX. The concept of valid dose should be defined previously, but generally EX information is assumed to be verified on non-missing EXDOSE value greater than zero in case of study drug or zero in case of placebo.

Full Analysis Set population flag is assumed to be set to "Y" for all patients. Nevertheless, depending on required analysis it may incorporate patients who were enrolled, randomized or signed informed consent.

Per-protocol population flag may be used for efficacy or any other required analysis. However, protocol may define other populations which differ from PPROTFL.

For some types of analysis numeric version of flags is also added to the database. Character and numeric subject-level population flag names end with FL and FN respectively. For both of them NULL value is not allowed and one-to-one mapping between –FL and –FN should be adhered.

Treatment Variables

In comparative clinical trials the randomized treatment allocation is a study foundation. Quality of the results depends on correctness of the randomization to any per-protocol treatment or intervention. There is different arm information in ADSL which should be carefully derived and checked with the SDTM domains data. As known treatment variables can show planned and actual arms and sometimes their values may differ. The information about the planned treatment comes from the IxRS (Interactive Voice/WEB Response System) and it is derived from DM or TA, while the actual treatment is taken from the CRF Exposure page. In spite of the derivation performed in DM actual arm may be compared with values from Exposure SDTM domain. Moreover, there is a chance that treatment information in EX (and EC if available) will be controversially changed per patient during the trial. Therefore, additional checks for EX and EC should be made to ensure that all data have been collected correctly and all patients received their treatment.

First of all, it is useful to focus on treatment-related variables across ADSL: ARM/ARMCD variables should be equal to TRT01P (in general, even in case planned treatment is changed for the next period) as well as ACTARM/ACTARMCD should be equal to TRT01A on the same principle. For subjects randomized in treatment groups but not treated, the planned arm variables should be populated, but the actual treatment arm variables should be left blank. In case of missing actual treatment variables in DM, it is worth to check SDTM EX domain to find any information about the treatment for a certain patient. The next step is to

compare EX with the actual and planned treatment from DM. According to the programmer's practice it would be useful to confirm that all treatment variables are valid and correctly spelled.

Then, if the multiple periods available, it should be confirmed that all period variables across the trial are associated with corresponding treatment provided in ADSL.

Trial Experience Variables

This category depends on Disposition Trial data. All variables require acceptable and valid DSDECOD, DSSDTC, DSCAT values. In general, except for the start and end dates-related variables which have been discussed before, this block may be conditionally divided into two information types: about end of the study and end of the treatment.

End of Study Status (EOSSTT) variable is directly based on DSDECOD values where DSSCAT is equal to "STUDY DISCONTINUATION". If the date of disposition event is non-missing, it should be filled with "COMPLETED", "DISCONTINUED" or "ONGOING" otherwise. This status also may be provided for the data cut-off. According to EOSSTT value, End of Study Date is set to the date of completion, discontinuation or the cut-off date. Reason for the study discontinuation comes from DS domain too and should be filled only in case of DSDECOD is not equal to COMPLETED and valid date value.

Similar variables for Treatment Discontinuation are also provided for each study period if available. In addition to standard checks for the DS data, other SDTM data sets which include information about the treatment and all study assessments should be verified.

The permissible TRTDUR-related variables describe the duration of the treatment of the whole study or periods measured in different units. If a subject is allocated to the treatment, duration variable is count from the reference start of treatment date as Day 1. All duration values are numeric. Thus to calculate Treatment Duration in days:

```
trtdur = adsl.trtedt – adsl.trtsdt + 1;
```

If the treatment dates are missing, TRTDUR should be set to missing as well.

Moving from standards, many variables which are not described in ADaM Implementation Guide, may also impact the validity of analysis if they are not handled correctly. The team keens on having all stratification variables, treatment or therapy flags, autopsy, baseline assessments and other important information in place and seeks to add all of them to the Subject-Level Data Set.

ADSL may be intended to include autopsy flag (e.g. ADSL.ADTHAUT). The values of the flag may be either "Y" if autopsy performed, or missing otherwise. There are multiple options to get information for this variable referring to Source Domain of Death. If DTHDOM = "AE" and data of autopsy is available there, then ADSL.ADTHAUT should be equal to AE.AEDTHAUT.

Obs	STUDYID	USUBJID	AEDECOD	AEOUT	AESDTH	AEDTHDTC	AEDTHAUT	DTHDOM	ADTHAUT
1	XYZ12345	XYZ12345-1283-5543	SEPTIC SHOCK	FATAL	Y	2019-02-03	N	AE	N
2	XYZ12345	XYZ12345-1134-1133	SEPSIS	FATAL	Y	2016-11-08	N	AE	N
3	XYZ12345	XYZ12345-2234-6543	MYOCARDIAL INFARCTION	FATAL	Y	2016-07-07	U	AE	U
4	XYZ12345	XYZ12345-2222-1256	SEPSIS	FATAL	Y	2018-12-23	Y	AE	Y

Table 6. Relation between SDTM.AE and Autopsy variable with possible issues.

In case of DTHDOM = "DS", autopsy-related variable from DS should be used if available.

Obs	STUDYID	USUBJID	DSDECOD	DSCAT	DSSTDTC	DTHDOM	ADTHAUT	DSDTHAUT
1	XYZ12345	XYZ12345-8642-3488	DEATH	DISPOSITION EVENT	2017-06-11	DS	N	N
2	XYZ12345	XYZ12345-8776-3233	DEATH	DISPOSITION EVENT	2018-10-24	DS	N	N
3	XYZ12345	XYZ12345-7442-1265	DEATH	DISPOSITION EVENT	2017-04-23	DS	U	U

Table 7. Relation between SDTM.DS and Autopsy variable with possible issues.

If previous options are not sufficient, autopsy is a procedure, so the data about it may come from Findings Data Set. Hence flag will be set to “Y” if FA.FADTC is not missing where FA.FATESTCD equals to “AUTOPSY” and FA.FASTRESC = “Y”. Such information can be also available in PR or any other SDTM.

Obs	STUDYID	USUBJID	FATESTCD	FATEST	FAOBJ	FASTRESC	FADTC	ADTHAUT	DTHDOM
1	XYZ12345	XYZ12345-1655-1654	AUTOPSY	Autopsy Performed	DEATH	N	2017-06-11	N	AE
2	XYZ12345	XYZ12345-8554-1487	AUTOPSY	Autopsy Performed	DEATH	N	2018-09-14	N	AE
3	XYZ12345	XYZ12345-9843-1123	AUTOPSY	Autopsy Performed	DEATH	N	2017-06-15	N	DS

Table 7. Relation between SDTM.FA and Autopsy variable with possible issues.

Similar to autopsy indicator, flags related to Adverse Events are widely used in ADSL.

“Subject Discontinued Study Due to AE” flag is usually derived from DS data if DSDECOD is equal to “ADVERSE EVENT”. To ensure consistency of this information between DS and AE data it is worth looking into variable related to OTHER ACTION TAKEN which often provides corresponding details of the event.

Obs	STUDYID	USUBJID	AEDECOD	AEACNOT	DSDECOD	DSCAT	DSSCAT	DISCAE
1	PO8865	PO8865-001-112	CYTOPENIA		ADVERSE EVENT	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION	Y
2	PO8865	PO8865-005-154	ALANINE AMINOTRANSFERASE INCREASED		ADVERSE EVENT	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION	Y
3	PO8865	PO8865-012-312	PLEURAL EFFUSION	SUBJECT DISCONTINUED FROM STUDY	ADVERSE EVENT	DISPOSITION EVENT	STUDY COMPLETION/EARLY DISCONTINUATION	Y

Table 8. Relation between SDTM.AE and SDTM.DS in derivation DISCAE variable.

AE Leading to Drug Withdrawal Flag is derived from SDTM AE data set. Variable related to Action Taken to Study Treatment displays the required information about this data to create AEWITHFL. However, the treatment discontinuation event should also be documented in DS domain.

Obs	STUDYID	USUBJID	AEDECOD	AEACN	DSDECOD	DSCAT	DSSCAT	AEWITHFL
1	PO8865	PO8865-456-003	NEUROPATHY PERIPHERAL	DRUG WITHDRAWN	ADVERSE EVENT	DISPOSITION EVENT	TREATMENT A	Y
2	PO8865	PO8865-456-014	HEPATOTOXICITY	DRUG WITHDRAWN	ADVERSE EVENT	DISPOSITION EVENT	TREATMENT A	Y
3	PO8865	PO8865-456-108	ENDOCARDITIS	DRUG WITHDRAWN	PHYSICIAN DECISION	DISPOSITION EVENT	TREATMENT A	Y

Table 9. Relation between SDTM.AE and SDTM.DS in derivation AEWITHFL variable.

CONCLUSION

Nowadays ADaM standards and different tools certainly help us to produce outputs without special efforts and reduce wasting time on data cleaning. But unfortunately absolutely all issues and tricky cases cannot be excluded. Therefore, it would be useful to be forewarned and ready to identify inconsistencies and report them to Data Managers ahead of time. The paper demonstrates potential pitfalls in the Analysis Subject-Level Data Set programming and provides basic guidance on checking issues which worth programmer’s attention during analysis process.

REFERENCES

1. CDISC Analysis Data Model Team, 2016, “Analysis Data Model Implementation Guide Version 1.1.”. (https://www.cdisc.org/standards/foundational/adam)

2. CDISC Submission Data Standards Team, 2018, “Study Data Tabulation Model Implementation Guide: Human Clinical Trials Version 3.3.”(https://www.cdisc.org/standards/foundational/sdtmig)

3. FDA Study Data Technical Conformance Guide, 2018.

4. CDISC ADaM Compliance Sub-Team, 2015, “ADaM Validation Checks Version

5. Shostak, Jack. 2014. *SAS® Programming in the Pharmaceutical Industry, Second Edition*. Cary, NC: SAS Institute Inc.

CONTACT INFORMATION

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

Anastasiia Oparii
Experis Clinical / Intego Group, LLC
+380 (44) 500 7020 (ext.2444)
anastasiia.oparii@intego-group.com
https://intego-group.com/

APPENDIX

```
* Macros to validate date variables                  *;
* The following macro variables are available:       *;
*     dsin     - user defined input data set name    *;
*     date_var - character variable in ISO8591       *;

%macro check_valid_date ( dsin     =
                        , date_var =  );
%let today = today();

data &dsin. (drop = _:);
  set &dsin.;

  if missing(&date_var.) then
     put "WARN" "ING: MISSING DATE. PLEASE CHECK USUBJID = " USUBJID;
  else do;
     _year  = scan(&date_var.,1,"-");
     _month = scan(&date_var.,2,"-");
     _day   = scan(&date_var.,3,"-");


 * Check YEAR part to be valid ;                                  *;
 * There are following possible options for valid year:           *;
 *   first digit (y1) may be equal either 1 or 2                  *;
 *   second digit (y2) may be equal either 9 or 0 respectively    *;

     if length(_year) ^= 4 then
put "WARN" "ING: INVALID LENGTH OF YEAR PART. PLEASE CHECK USUBJID = "
USUBJID;
     else do;

* set possible combinations of valid year to compare with YEAR part parsed
from the analyzed date;

        _y1 = strip(ifc(substr(_year,1,1)="1","1","2"));
        _y2 = strip(ifc(substr(_year,1,1)="1","9","0"))   ;

* compare first two digits from YEAR part with first digits which are valid
("19" or "20");

        if  substr(_year,1,2) ^= strip(_y1)||strip(_y2) then
put "WARN" "ING: INVALID YEAR VALUE. PLEASE CHECK USUBJID = " USUBJID;
         else do;


        if _year > scan(put(&today.,is8601da.),1,"-") then
put "WARN" "ING: INVALID YEAR VALUE. PLEASE CHECK USUBJID = " USUBJID;
         else do;

* Check MONTH part to be valid ;

* month number is greater or equal to 1 and less or equal to 12. *;

         if input(_month,best.) < 1 or input(_month,best.) > 12 then
put "WARN" "ING: INVALID MONTH VALUE FOR USUBJID = " USUBJID ".PLEASE CHECK!"
;
```

```
         else do;

 if missing(_day) then
put "WARN" "ING: MISSING DAY VALUE FOR USUBJID = " USUBJID ".PLEASE CHECK!";
  else do;

       * check for FEB depending on leap year;

  if input(_month,best.) = 2 then do;

           if intck("day",
mdy(2,1,input(_year,best.)),mdy(3,1,input(_year,best.))) = 28 then do;

            if input(_day,best.)>28 then
put "WARN" "ING: INVALID DAY VALUE FOR USUBJID = " USUBJID ".PLEASE CHECK!";
       end;
       else if  input(_day,best.)>29 then
put "WARN" "ING: INVALID DAY VALUE FOR USUBJID = " USUBJID ".PLEASE CHECK!";
          end;

    * check for months with 30 or 31 days respectively;

    else if input(_month,best.) in (4,6,9,11) and input(_day,best.) > 30 then
put "WARN" "ING: INVALID DAY VALUE FOR USUBJID = " USUBJID ".PLEASE CHECK!";

    else if input(_day,best.) > 31 then put "WARN" "ING: INVALID DAY VALUE
FOR USUBJID = " USUBJID ".PLEASE CHECK!";

      *  re-check that date if earlier or equal to today;

   else if  input(&date_var.,is8601da.)> &today. then put "WARN" "ING: DATE
IS LATER THAN TODAY";

              end;
        end;
        end;
    end;
end;
end;
run;
%mend check_valid_date;
```