PhUSE 2013

Paper DH09

Datacut Strategies: What, why and how

Hiren Naygandhi, Roche Products Ltd, Welwyn Garden City, UK

ABSTRACT

A datacut is when the data have been cut into subsets, according to a set of rules. This can be at a particular date, or a particular set of patients, or a significant point in time. The data in that subset are used for reporting purposes. Project teams can implement this in many ways, for example cutting at the SDTM or analysis level datasets. The paper describes the advantages and disadvantages of different datacut strategies and their implementation. Cutting the analysis data will mean the SDTMs remain unchanged, maintaining its status as source data. It also, however, means there may be additional data between the datacut and DB lock, within SDTMs that could result in unnecessary regulatory questions. The paper will also look at the possibility of removing the datacut process entirely: what if there was no datacut process?

INTRODUCTION

During the life cycle of a clinical trial, there could be several times at which analysis on the data may be required. Examples include: DSUR, Interim Analysis, regular IDMCs, Regulatory requests and so on. For each of those analyses, a datacut, would be required to subset the data based on the reporting period. The datacut process may involve additional rules, such as imputation of partial dates or how to handle data that may be present after the datacut cut-off. Project teams may decide to follow different strategies based on their requirements or experience. What is important is to consider all strategies, comparing the merits of each and to then make a decision.

The Author has used several terms within the paper that may not be widely used. For clarity, some definitions are provided here: *Source* refers to original raw or CRF (or non-CRF) data that is collected on a study; *Analysis* refers to analysis performed using source data that is used for reporting purposes; *Rollback* refers to re-deriving variables where they provide data that is considered to have occurred after the datacut.

WHAT IS A DATACUT?

A datacut is when the data have been cut into subsets, according to a set of rules. This can be at a particular date, or a particular number of patients, or a significant point in time. The data in that subset are used for reporting purposes.

Sounds simple enough! But why is it important?

WHY DO WE NEED IT?

A datacut enables data to be reported in a project; on all events that have occurred up to that point i.e. gives a true reflection of what has occurred up to the data cut-off. So why not just take everything on the data cut-off and then provide that? This would be ideal, however, it does not quite work in reality and the reason is: *dirty data*. Once a data cut-off is decided upon, there may be a period after that point where data needs to be *cleaned* i.e. check to see if any data is missing; values out of range; data entered incorrectly and so on. This is to ensure the data is as accurate as possible. So as an example, you could have a data cut-off on 15JAN2013, however, Data Management (DM) will need two weeks to clean the data, so we would need to take a snapshot of the data on 30JAN2013. The 15JAN2013 date would typically be referred to as the *clinical cut-off date* and the 30JAN2013 date would be the *snapshot date*. Based on this example, we cannot take all data as of the snapshot date, as there may be additional data on the study dated after the clinical cut-off date and DB lock date, as data entry may still continue after the clinical cut-off date. For this reason, data cutting rules need to be applied to ensure we include only the clean data in our analysis.

ADDTIONAL RULES

There are additional rules that need to be considered, when cutting data and also after it is cut:

1. Identify cut-off fields: if datacut is based on a date, domain specific date variables would need to be identified and used as the data cut-off comparator. If, however, a date is completely missing, further cutting rules would need to be in place e.g. for the Adverse Event (AE) domain, the AE start date is used to cut the data. If the AE start date is completely missing then use AE end date. These types of rules would need to be put in place for each domain.

2. Partial dates: data cut-offs based on dates, would typically use the event date of the particular domain when cutting e.g. Lab collection date, AE start date and so on. Whilst the data is cleaned as much as it can be, there may be cases where some dates are partial e.g. only a month is provided or only month and year (but date is missing), in which case date imputation rules may need to be applied to determine the imputed date of event. Discussion of possible date imputation rules are outside the scope of this paper, however, the important point to note here is the imputed date would be used to determine whether or not that particular data record is included within the datacut. Note: Date imputation rules would typically be identified within documentation for analysis of study e.g. sometimes referred to as a Statistical Analysis Plan.

3. Rollback: There may be instances where there is data that is based after the data cut-off e.g. using a cut-off date of 15JAN2013; a patient had an AE that started on 10JAN2013 and ended on 20JAN2013. The AE record is included, as it is dated within the data cut-off; however, data is available that gives more information after the cut-off date. Since the aim of the datacut is to report up to 15JAN2013, it could lead to bias, since it does not give a true reflection of the data as of that period. One approach could be to treat the AE as 'ongoing' and reset the end date to missing. The main reason for this modification of data is to give a true reflection of the study as of 15JAN2013 and it would be critical to ensure this is informed up-front and any future datacuts would have the modified data, changed back to what it was in the database. Further discussion will be made in the Analysis section on the merits of this approach.

HOW IS IT APPLIED

We now consider different examples of how a datacut can be applied in Table 1.

Table 1

Scenario	Cut applied at Source or Analysis	Imputed dates	Rollback
1	Source	Dropped	No (flags created in analysis datasets)
2	Source	Retained	Yes
3	Analysis	Retained	No

ANALYSIS

We now compare and contrast the different scenarios.

SOURCE OR ANALYSIS LEVEL

Scenarios 1 and 2 apply the cut at the source datasets; however, Scenario 3 applies it at analysis level. The impact of applying a cut at source level allows the data to be carried through to analysis i.e. there would be no mismatch between the subset of data in both source and analysis datasets. Cutting at analysis level means source level data does not need to be changed i.e. the source data retains its integrity. However, doing so may

provide discrepancies between source and analysis data i.e. source may have had more data that occurred after the datacut. Whilst it may not be an issue, if both source and analysis data were to be submitted to Regulatory Authorities (RA), it could lead to further questions.

RETAINING IMPUTATION RULES

In Scenario 1, the derived dates for imputation of partial dates are dropped in the cut datasets, however, in Scenarios 2 and 3 they are retained. The main reason to drop the imputed dates, after processing has been done, is to ensure the source data stays intact and no derivations or analysis are applied there. The drawback of this approach is date imputation programming will have to be done twice: initially at the source level (where it is subsequently dropped) and then at analysis level for reporting purposes (where it would be retained).

Note: Scenario 2 cuts the data at the source and retains the imputed dates. This approach ensures imputations only need to be made once and can be re-used in analysis. There are, however, a couple of drawbacks with this:

1. Derivations are being added within the source level datasets (instead of doing this at the analysis level datasets).

2. This approach will not work if using SDTMs, as the structure is defined by standards and adding or updating date variables cannot be done in this case.

ROLLBACK

Scenarios 1 and 3 do not apply any rollback, whereas Scenario 2 does. The reason to apply rollback is to give an accurate as possible reflection of the data as of the cut-off. The biggest drawback of it is that it involves manipulation of data i.e. in Scenario 2, source data events that occurred after the datacut would be changed to null or ongoing. Provided this is documented upfront, it is a valid approach. What should be considered very carefully is what would happen if there was another datacut, based on the reporting period from the previous cut to the latest one? This would mean, those variables that may have been updated, will need to be reverted back to their original values if they fall within the latest cut. This approach would need to be checked carefully to ensure it doesn't become convoluted.

One reason to not apply rollback is to avoid modifying any values of variables and keep them as is; only manipulation when cutting is: imputation of dates and subsetting, nothing more. Another reason is how much of an impact does applying rollback have on the overall picture? There may only be a hand few observations that may be required to be updated for rollback, in which case, is it worth applying if it doesn't make a significant impact on the analysis?

Scenario 1 provides an alternative to rollback, by deriving flags in each domain where an event has occurred after the datacut, within the analysis level datasets. This would clearly identify observations that fall after the datacut and would display transparency to RA. It also provides a safety net, as these flags could be utilised in derivations should you want to discard data from derivations if there is a requirement to use them e.g. dated before or on the clinical cut-off date.

PROGRAMMING MADE EASIER

Moving to standard processes means being able to follow the same process across projects and possibly even industry. It also enables flexible programming. Take SDTMs for example. The nature of their structure and naming conventions makes it much easier to develop a datacut program e.g. all event date variables are typically named as *DOMAIN*.*DTC* i.e. LBDTC or AESTDTC (with start or end dates they would have *ST* or *EN* in between). This makes creating a generic macro for a datacut fairly straightforward. Figure 1 shows a snippet of how it could be done in SAS®.

FIGURE 1

```
%macro g_sdtmvcut(macroname=&sysmacroname,donotcut=);

  *** Identify the date fields;
  %macro cut;
    %do i = 1 %to &tot;
      %str_util_exist(objtype=&&dsin&i, objname=&&dsin&i..stdtc, outmvar=_foundsdat);
      %str_util_exist(objtype=&&dsin&i, objname=&&dsin&i..dtc, outmvar=_founddat);

      %if &_foundsdat > 0 %then %let __var = &&dsin&i..stdtc;
      %else %if &_founddat > 0 %then %let __var = &&dsin&i..dtc;

      *** Apply impuation rules to date fields;
      data __&&dsin&i;
        set &&dsin&i;
        %str_util_dttm_impute(datein   = &__var,
                              hour     = 23,
                              minute   = 59,
                              second   = 59,
                              imputed  = __&&dsin&i..dtm,
                              dateflag = ,
                              timeflag =
        );
      run;

      *** Cut data;
      %g_datacut(dsin    = __&&dsin&i,
                 dsincut  = ,
                 compdt   = __&&dsin&i..dtm,
                 cutdate  = &__cutdt,
                 donotcut = ,
                 dsout_cut = ana.&&dsin&i..cut,
                 dsout     = _&&dsin&i) ;

   %end;

   %mend cut;
   %cut;

%mend g_sdtmvcut;
```

This datacut macro example uses a date as the cutting approach. The first part searches for a variable names ending in *STDTC* and *DTC*. It then sets the macro variable *__var* to the start date variable if that exists and if not, to the event date variable, within each domain. The next steps include the calls for the imputation and datacut macros. These apply the imputation rules to get the imputed dates and then use those dates against the datacut date reference, macro variable *__cutdt*, to subset the data required. This makes programming much more flexible and means the datacut process can be centralized.

FURTHER CONSIDERATIONS

EVENT DATE

The discussion on dates, up to this point, has been around partial event dates and at which level to impute them, source or analysis, however, what if we didn't apply any imputation rules? If we didn't, that may result in certain observations missed off within the cut due to partial dates. An alternative (and slightly controversial) approach would be to use the date of database entry date instead of event date. This would mean instead of using the event date e.g. AE start date, Vital Signs assessment date etc. we would use the date entry of when that event was entered into the database (assuming this is available within source data).

The advantage of using such an approach is that it eliminates the need to impute for partial dates. The date of entry in the database would always be non-missing and therefore there will never be partial. It also means you would take an accurate reflection of what is in the database as the time of the cut-off. The risk, however, is there could be observations that are missed off the cut due to a delay in data entry e.g. the cut-off date is 15JAN2013, the lab collection date is 10JAN2013, however, the date of entry on the database is 17JAN2013. This would mean those lab records will not be present in the cut, as their date of entry was after the cut-off date. Even though it would take an accurate reflection of what is in the database, it may not take an accurate reflection of what has happened within the study i.e. the example above with the lab collection date.

There is no right or wrong approach here and in fact both approaches could be used depending on the purpose of the cut e.g. for a DSUR, emphasis is not placed as much on cleaning data, as the objective is to provide a

safety update on a regular timely basis. Here, the approach of using the date of entry as the cut-off makes sense as any data missed off in the cut will be captured in the next DSUR reporting. For a poster publication, there will be a greater emphasis placed on ensuring the data is as clean as possible and may mean the event dates are a more appropriate choice.

DO WE NEED DATACUT PROCESS?

Do we need a datacut process? The key word in the question is *we* and the discussion around the responsibility. The author is a Statistical Programming Analyst (SPA). Much of the work within a datacut has been performed by SPA (within the author's experience). DM is responsible for delivery study source data to SPA. What if DM delivered data to SPA that was already cut? If this transfer included cut data, it would enable the process to be removed from SPA and passed on to DM. DM may be able to use a script to apply a datacut (similar if not the same as Figure 1) before delivering the data.

It can be argued, that applying a datacut at that stage may be considered to be performing analyses, which might not be part of DM roles and responsibilities. Whilst this is simply a proposal, in reality, it might be difficult to achieve as:

1. The definition of a datacut process needs to be aligned across all projects. For such a definition it's likely other functions will need to be involved i.e. Science, Stats and so on, as SPA may not be the final decision makers. If this cannot be done, or separate rules defined for different projects, then it may lead to complexities in the technical aspects of implementing rules for individual projects, meaning it may not be worth implementing a generic datacut process.

2. It would require agreement from DM. DM may argue that this is not part of their responsibility and therefore not be keen on taking on an additional task on top of their existing ones.

CONCLUSION

This paper has described the various strategies and rules that can be used when applying a datacut. Whilst there are pros and cons for these suggestions, it is important to ensure there is a defined process. Efforts should be made to make this a generic process across all projects; however, even if the approach is different from project to project, there should be a standard process that defines what should be done as a standard in each project. Statistical Programming may not be the final decision makers and as such it will require much collaboration between other functions such as Statistics and DM. One factor to also consider is providing clarity for regulatory submissions, to assist the FDA with reproducing analyses from source data. This is an important factor to consider, whichever approach is considered.

The author's recommendation is to consider using multiple strategies depending on the reporting purpose of the datacut. If the purpose is for interim analysis, the recommendation would be to use Scenario 1 described earlier i.e. cut data at source level and to not apply rollback. If the purpose is for DSUR reporting, we know it is not critical to have the data as clean as possible. In this instance the recommendation would be to cut the data based on the database date of entry and to not consider any imputation rules.

Whichever approach is taken, it should be made consistent across all projects, to ensure the same process is being applied.

Special thanks to Hinal Patel, Vijay Reddi and Chris McKenna for their review and insights of the topics covered.

CONTACT INFORMATION

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

Author Name:	Hiren Naygandhi
Company:	Roche Products Ltd.
Address:	6 Falcon Way, Shire Park
City / Postcode	Welwyn Garden City, AL7 1TW, UK
Work Phone:	+44 (0) 1707 366160
Email:	hiren.naygandhi@roche.commailto:hiren.naygandhi@roche.com