ADaM Implementation Guide

Prepared by the CDISC ADaM Team

Notes to Readers

Revision History

Date	Version	Summary of Changes
May 30, 2008	1.0 Draft for Public Comment	

Note: Please see Appendix A for Representations and Warranties, Limitations of Liability, and Disclaimers.

Contents

1 Introduction

1.1 PURPOSE

This document comprises the Clinical Data Interchange Standards Consortium (CDISC) Version 0.2 Analysis Data Model Implementation Guide (ADaMIG), which has been prepared by the Analysis Data Model (ADaM) team of CDISC. This document is intended to guide the organization, structure, and format of analysis datasets and related metadata.

The ADaMIG should be used in close concert with the current version of the Analysis Data Model which is available at (web link). The Analysis Data Model describes key principles that apply to all analysis datasets, with the overall principle being that the design of analysis datasets and associated metadata facilitate explicit communication of the content, input, and purpose of submitted analysis datasets. Another important design goal for the ADaM standard is to support efficient generation of analysis results.

The user of the ADaMIG should also be familiar with the CDISC Study Data Tabulation Model (SDTM) and the Study Data Tabulation Model Implementation Guide (SDTMIG), both of which are available at http://www.cdisc.org/standards/index.html, since the ADaM concepts build on this model.

Both the SDTM and ADaM standards were designed to support submission by a sponsor to a regulatory agency, such as the U.S. Food and Drug Administration (FDA). However, the standards are also used to facilitate interchange of data among and within a broader range of organizations. For example, prior to submission, study tabulation and possibly analysis data may be provided to a sponsor by a university, a contract research organization, or other partner. Even within the same company, study tabulation data may be provided by a clinical data management organization to a separate statistical group.

Analysis datasets and associated metadata are one of the four types of data that can be submitted to the FDA, along with study tabulation datasets, subject profiles, and listings. Each of these types of data has a place in the folder structure of the Electronic Common Technical Document (eCTD). The SDTM datasets are not intended to fully support statistical analyses; therefore, the submission of ADaM datasets is advocated. The ADaM standard facilitates understanding of how the observed clinical data have been used to derive the variables (columns) and observations (rows) that are used for statistical analysis as well as how analysis results, such as p-values, were generated. Given that the ADaM standard has been developed as part of the larger family of CDISC standards, it is assumed that there is a relationship that can be described by metadata between the analysis datasets and the study tabulation data.

1.2 REFERENCES AND ABBREVIATIONS

The following are abbreviations for the documents referenced within this document and the links to the current versions:

1.3 ORGANIZATION OF THIS DOCUMENT

This document is organized into the following sections:

- Section 1, INTRODUCTION, provides an overall introduction to the importance of the ADaM standards and how they relate to other CDISC data standards.
- Section 2, THE ADaM DATA STRUCTURES, provides a review of the key principles that apply to all ADaM datasets and introduces two standard structures that are flexible enough to represent the great majority of analysis situations. Categories of analysis variables are defined and criteria that are deemed important to users of analysis datasets are presented.
- Section 3, STANDARD ADaM METADATA, defines standard variable metadata for analysis variables that

commonly would be used in the ADaM standard data structures.

- Section 4, IMPLEMENTATION ISSUES AND SOLUTIONS illustrates the use of the ADaM basic structure to address common analysis situations. The following subsections are presented:
    - 4.1 Creation of Derived Columns Versus Derived Rows – presents rules that dictate when a row versus a column should be created
    - 4.2 Inclusion of All Observed and Derived Records for a Parameter Versus the Subset of Records Used for Analysis – presents rationale for inclusion of all records
    - 4.3 Inclusion of Input Data That Are Not Analyzed But That Support a Derivation in the Analysis Dataset – expands on the concepts outlined in 4.2 and provides examples of how to present supportive versus analyzed data.
    - 4.4 Identifications of Rows Used for Analysis – presents general and specific examples of how to identify rows used for analysis versus those that are supportive
    - 4.5 Identification of Population-Specific Analyzed Records – presents solutions for how to identify records that are used for different population-level analyses, including both subject-level and record-level population analyses.
    - 4.6 Identifications of Records Which Satisfy a Predefined Criterion for Analysis Purposes – presents a solution that can be used to identify observations that fulfill one or more criteria.
    - 4.7 Other Issues to Consider – provides comment on other issues that may arise when creating analysis datasets.

1.4 DEFINITIONS

Input Data – The data used for the creation of analysis data sets. Example: The QS and EX domains from the SDTM study tabulation data, and the ADaM dataset ADSL, were the input data for the creation of the analysis dataset ADEFF, which was used for analyses of the primary efficacy measures.

Analysis value – (1) The character or numeric value described by the analysis parameter. The analysis value may be present in the input data or may be derived. Example: The analysis value of the parameter 'Average Heart Rate (bpm)' was derived as the average of the three heart rate values measured at each visit. (2) In addition, values of certain allowed functions are considered to be analysis values. Examples: baseline value, change from baseline.

Analysis parameter – A row identifier used to characterize uniquely a group of values that share a common definition. Example: The primary efficacy analysis parameter is '3-Minute Sitting Systolic Blood Pressure (mm Hg)'. Note that the ADaM analysis parameter contains all of the information needed to identify a group of like analysis values. In contrast, the SDTM --TEST column may need to be combined with qualifier columns such as --POS, --LOC, --SPEC, etc., in order to identify a group of like values.

Analysis timepoint – A row identifier used to classify values within an analysis parameter into temporal or conceptual groups used for analyses. These groupings may be observed, planned or derived. Example: The primary efficacy analysis was performed at the Week 2, Week 6, and Endpoint analysis timepoints.

Parameter-invariant – The property of being defined the same for all analysis parameters in an analysis dataset. A column is parameter-invariant if its metadata are not a function of analysis parameter. Example: The change from baseline column CHG is a parameter-invariant same-row function of the AVAL and BASE columns. The metadata (CHG = AVAL - BASE) do not vary according to parameter.

Traceability – The property that permits the user of an analysis dataset to understand the relationship of analysis values to the study tabulation datasets. Example: Based on the metadata and the content of the analysis dataset, the reviewer was able to trace how the primary and secondary efficacy analysis values were derived from the study tabulation data for each subject.

Supportive – Enabling traceability. A column or row is supportive if it is not required in order to perform an analysis but is included in order to facilitate traceability. Example: the LBSEQ and VISIT columns were carried over from SDTM in order to promote understanding of how the analysis dataset rows related to the study tabulation dataset.

Analysis-enabling – Required for analysis. A column or row is analysis-enabling if an analysis cannot be performed without it. Example: the hypertension category column was added to the analysis dataset in order to enable subgroup analysis.

Observation – A row in a dataset; a record.

Variable – A column in a dataset.

2 The ADaM Data Structures

2.1 INTRODUCTION

Analysis datasets should adhere to four key principles as described in the Analysis Data Model:

- Analysis datasets should facilitate clear and unambiguous communication of the content of the datasets supporting the statistical analysis performed in a clinical study, should provide a level of traceability to allow an understanding of the relationship of analysis values to the input data, and should identify when analysis data have been imputed.
- Analysis datasets should be readily usable with available software tools.
- Analysis datasets should be linked to machine-readable metadata, because clear and unambiguous communication relies heavily on the availability of metadata. Machine-readable metadata facilitate software tool development.
- Analysis datasets should have a structure and content that allows statistical analysis to be performed with minimum programming. Such datasets are described as 'analysis ready'.

To assist review, analysis datasets and metadata should clearly communicate how the analysis datasets were derived. This requirement implies that the user of the analysis dataset ought have at hand the input data used to create the analysis dataset in order to be able to verify derivations. It is important to note that this concept is independent of the use of data standards since the relationship between the collected and the analyzed data is important in any submission. In the context of the use of CDISC standards, it follows that the relationship between SDTM and ADaM should be clear. This promotes the traceability between the input data (such as SDTM) and the analyzed data (ADaM) that is of high importance. If SDTM is not the input data used to create the analysis datasets, sponsors should provide documentation and adequate ADaM metadata that will help the user of the analysis dataset understand how the SDTM data could be used to recreate important analyses.

Although SDTM is a standard format for collected data, the SDTM also contains a few derived items, such as baseline flag, derived records, subject reference start date, study day, standardized results, and subject-level population flags. Given that SDTM requires a few derived variables and that sponsors may wish to include additional derived data in the SDTM domains, the problem exists of how to obtain the derived values for representation in SDTM and how to represent the necessary metadata, such as the computational method, in SDTM. In 2006, CDISC sponsored a special project, the SDTM/ADaM Pilot project, which explored the process of placing derived data into SDTM domains. In this project, a number of issues were uncovered regarding how to represent adequately the metadata associated with derived variables in SDTM within the Define.XML. The pilot project highlighted the need for more investigation into how to resolve the outstanding issues.

Though it may be useful for reviewers to have derived baseline records, baseline flag, and subject-level population flags in SDTM, the authoritative source for the unequivocal values and important explanatory metadata is the ADaM analysis datasets. If the identical derived data are represented in both SDTM and ADaM, sponsors would be prudent to ensure that the values are identical or provide explicit SDTM and ADaM metadata which when considered together explain why the values are different, and which ones were used for the reported statistical analysis. It is

also important to note that analysis datasets often combine information from multiple input domains. One observation within an analysis dataset may contain variables whose values are either copied directly from, or derived from, variables located within Events, Findings, and Interventions domains and, as well, may represent the combination of both safety and efficacy variables. This combining of variables from multiple domains is one advantage of analysis datasets. Attempting to place this derived information back into SDTM would dissociate these variables and no longer provide the linkage that is needed by reviewers. In addition, the metadata in SDTM would need to describe the relationship between the variables and the computational methods used for derivations.

The use of CDISC standards encourages software development and it is possible that in the future, software tools will facilitate making derived data in ADaM available together with the SDTM data. For example, when defining the required analysis dataset, ADSL, ADaM recognized the value of having a dataset with one record per subject that contains important subject-level variables, such as population flags. This data structure was selected because it supports simple merges with any other type of domain structure, a task that should be a simple feature to implement within a software solution.

2.2 THE ADAM REQUIRED DATA STRUCTURES

2.2.1 Introduction

Given that a key principle of analysis datasets is clear communication and that analysis datasets contain both input data and data that have been derived in the process of creating the analysis datasets, a central issue becomes communicating how the variables and observations were derived and how observations in the analysis datasets are used to produce analysis results. Optimally, the user of an analysis dataset should be able to identify clearly the data inputs and the algorithms used to create the derived information. If this information is communicated in a predictable manner, through the use of a standard data structure and metadata, then the user of an analysis dataset should be able to understand quickly how to use appropriately the analysis dataset to replicate results or to explore alternative analyses.

The ADaM team has spent considerable time discussing the relationship between the structure of analysis datasets and the type of analysis being conducted. The consensus was that many types of statistical analysis do not require a specialized structure. Said in another way, the structure of an analysis dataset does not necessarily limit the type of analysis that can be done, nor should it limit the communication about the dataset itself. Instead, if a predictable structure could be used for the majority of analysis datasets, then communication may actually be enhanced. In addition to supporting clear communication to the user of the analysis dataset, a predictable structure has other advantages. First, a predictable structure should ease the burden of the management of metadata that describe the observations and variables in the dataset because there will be less variability in the types of observations and variables that are included. Second, once a predictable structure is defined, software development can progress to support the management of the metadata and to support the development of software tools that aid in the review of the data, including tools that may allow restructuring of the data (transposing) based on known key variables. Lastly, predictability in structure will facilitate testing whether an analysis dataset conforms to ADaM standards, using a set of known conventions that can be verified to be present. Because of the nature of analysis datasets and the need to retain flexibility, it is unlikely that the ADaM model will support compliance testing as rigorous as that possible for the SDTM, in which there is a high degree of regularity and specificity of the content and structure. However, one goal of the ADaM model is to support compliance testing as much as possible.

The user of any data standard likely will recognize that future advances in software development and computing environments may someday obviate the need for some of the requirements set forth by the standard. This is true for analysis datasets since the value of the analysis dataset does not reside within the definition of the structure or content but rather the metadata associated with each variable and observation. By thinking of variables with an object-oriented mind-set, one begins to understand that knowing where a variable or observation resides is not as important as knowing how the variable or observation was created. This dissociation between location and creation, coupled with the use of structured metadata, may impact the future state of statistical computing environments. The use of structured metadata may someday allow analysis datasets to be virtual in the sense that any variable or observation could be recreated upon request and then be associated with other variables and observations.

2.2.2 The ADaM Basic Data Structure

The ADaMIG presents metadata for two standard structures:

- Subject-level dataset ADSL
- Multiple-record-per-subject basic data structure

The ADAMIG focuses mainly on the standard multiple-record-per-subject structure, referred to as the ADaM basic data structure, the ADaM basic structure, or simply as the basic structure.

The ADaM basic data structure is flexible and yet predictable. The majority of analyses, regardless of the therapeutic area or type of analysis, can use this standard structure. However, it is recognized that there may be some analysis situations that cannot be adequately represented with this model. It is incumbent on the user of this model to understand and appreciate when an alternate structure for the analysis dataset is truly needed. If an alternate structure is needed, then the resulting analysis dataset would not be considered ADaM-compliant; however, it should still adhere to the principles discussed in the ADaM V2.1 document.

The ADaM basic structure is a normalized design that can be loosely described as one or more records per subject per analysis parameter per analysis timepoint. Other variables, such as the analyzed record flag, population flags, and record derivation type may be necessary to uniquely identify an observation. From a broad conceptual point of view, however, the variable(s) describing the subject, analysis parameter, and analysis timepoint can be considered the most important variables in understanding the basic structure.

Metadata for the standard ADaM variables are presented in Section 3, and the ADaM basic data structure and variables are discussed and illustrated in Section 4. Briefly, the ADaM variables can be categorized in the following groups:

Variable Group	Description
Subject Identifiers	Variables that uniquely identify a subject, such as USUBJID
SDTM Identifiers	Variable(s) from SDTM, such as --SEQ, VISITNUM, and VISIT, that can be used to trace data in the analysis observation back to SDTM.
ADaM Timing Identifiers	Variable(s) to describe the observation with respect to the timing of the analysis parameter, such as AVISIT, AVISITN, ADY. Analysis timepoints can be absolute, relative, or conceptual.
ADaM Parameter Identifiers	Variable(s) to describe the analysis parameter, such as PARAM and PARAMCD.
ADaM Analysis Values	Variables containing character or numeric analysis values, such as AVAL and AVALC. Also includes variable BASE (Baseline Value), and any variable that is a parameter-invariant function of BASE and AVAL on the same row, such as CHG and PCHG..
Analysis Enabling Variables	Variables that are required for performing a statistical analysis. For example, indicator variables, such as population flags or analyzed record flag (e.g. ANLFL), are needed to identify the observations that are used in an analysis; and variables that are used in statistical model statements, such as treatment variables (e.g. TRTP) and covariates, are needed in order to perform the analysis. Also includes variables such as ACAT, SHIFT, and CRIT that group analysis values for categorical analysis.
Supportive Variables	Variables such as the SDTM Identifiers that are provided to support traceability back to the input data, and any other variables that are included to support understanding of how the analysis variables and observations were derived.

Whereas the ADaM basic structure may upon first glance be considered similar to an SDTM Findings domain, it is important to dismiss this similarity since the use of this structure is not limited to findings data. Instead, in the ADaM model, an observation in the analysis dataset can represent any observed or derived value required for analysis. For example, it might be a time to an event, such as the first visit a score became greater than a predefined value, or the relative day of discontinuation. Or it might be a highly derived quantity such as a surrogate for tumor growth rate derived for each subject by fitting a regression model to that subject's chemical marker laboratory data. This concept is elaborated in Section 4 with additional schematic examples.

This structure was chosen because it is flexible and contains a standard set of variables with standard variable names that can be used to represent the most frequent analysis concepts. Most importantly, a consistent set of variables, coupled with prudent addition of indicator variables, lends itself well to the specification of selection criteria (e.g., SQL or WHERE statements) that can be used within software programs to identify observations of interest and/or replicate analyses.

One drawback of this structure is that one column may be used to store values from numeric results that are obtained from multiple parameters and these results may have different levels of numeric precision. Since all values are stored in the one column, many software tools will pad all numeric results to the maximum level of precision in the column when displaying results. For example, a true value of 82 may be represented as 82.00 if this column variable had another parameter that was significant to 2 decimal places. However, value-level metadata can indicate the correct degree of precision, and structured metadata may be used by compliant software to represent analysis values according to the desired precision.

2.2.3 ADaM Criteria

Four criteria are of frequent interest to reviewers who use the analysis datasets to verify and validate submitted results. These are the following:

- Identify observations that exist in the submitted study tabulation data (e.g. SDTM).
- Identify observations that are derived within the ADaM analysis dataset.
- Identify the method used to create derived observations.
- Identify observations used for analyses, in contrast to observations that are not used for analyses yet are included to support traceability or future analysis.

Creators of ADaM datasets should provide variables and metadata to fulfill these criteria. An illustration of the use of variables to satisfy these criteria is presented in Table 2.2.3.1 in abbreviated form. The column variables are defined in Section 3; their use is demonstrated in the examples in Section 4.

Table 2.2.3.1

USUBJID	QSSEQ	VISIT	AVISIT	PARAM	AVAL	DTYPE	ANLFL
1001	198	WEEK 8	WEEK 8	Symptom Score for Item 1	134		
1001	198	WEEK 8	WEEK 10	Symptom Score for Item 1	134	LOCF	Y

In this example, the dataset is designed to support a Last Observation Carried Forward (LOCF) analysis at Week 10. The subject discontinued after the WEEK 8 visit, so SDTM does not contain a Symptom Score for Item 1 at the WEEK 10 VISIT. As indicated in DTYPE (Derivation Type), an analysis timepoint (AVISIT) WEEK 10 observation was derived in the ADaM dataset by the LOCF method. As indicated by the Analyzed Record Flag (ANLFL), the collected AVISIT=WEEK 8 observation is not analyzed, but is included to support traceability.

SDTM variables that are typical candidates for traceability are --SEQ, VISIT, and VISITNUM. In the above example, the SDTM variable --SEQ (e.g. QSSEQ) is sufficient to satisfy traceability and is a more predictable input

variable since it is a required variable in SDTM, as opposed to VISIT, which is permissible. However, VISIT is useful also, because it is more meaningful in terms of the protocol and study context.

In many analysis situations, data values from multiple domains are used to define one analysis parameter value. To maintain traceability back to multiple SDTM domains, it would be necessary to retain the SDTM identifying variables from each input domain. For example, if an analysis parameter is defined as being the earliest relative day on which a subject had a specific adverse event, a laboratory value greater than some threshold, or a questionnaire assessment score greater than a given value, then it would be necessary to retain SDTM sequence variables from AE, LB, and QS. Note that the situation may occur where multiple records from the same domain are used to define one analysis parameter, in which case the name of the SDTM identifying variable would need to be changed in order to have both values on the same analysis record. For example, if an analysis parameter were based on the first occurrence of two disposition events, the variable DSSEQ would have to be renamed for one of the events. There are situations where traceability via variables becomes impractical, for example where an analysis timepoint is defined as the average of the last 21 days of e-diary data, in which case there might be 21 or more values of --SEQ that are relevant. In these cases, communication can best be supported through clear metadata.

In this document, Section 4 illustrates recommended solutions to commonly occurring analysis situations using the ADaM basic data structure.

3 Standard ADaM MetaData

This section defines the required characteristics of standard variables (columns) that are frequently needed in analysis datasets. The ADaM standard assumes that these variable names will be used when a variable that contains the content defined in Section 3 is included in an analysis dataset.

Sections 3.1 through 3.8 describe metadata for the ADaM basic data structure. Section 3.9 describes the standard subject-level dataset ADSL.

The metadata tables in Section 3 contain a "Core" column that describes whether a variable is required, conditionally required, or permissible:

Values of ADaM Metadata "Core" Column

- **Req** = required
- **Cond** = required if applicable (conditionally required)
- **Perm** = permissible

For the variables that are defined as required or conditionally required, the variable label that is specified below is required. For other variables, the labels specified below are suggestions, and may be changed to be more specific. The metadata for these variables would be sponsor defined.

Any ADaM variable whose name is the same as an SDTM variable name is assumed to be a copy of the SDTM variable, and its label and values may not be modified. ADaM adheres to the principle of harmonization known as "same name, same meaning, same values."

All ADaM variable names must be no more than 8 characters in length, start with a letter or underscore, and be comprised only of letters, underscore, and digits. All ADaM variable labels must be no more than 40 characters in

length.

In Section 3, an asterisk (*) is sometimes used as a variable name prefix or suffix. The asterisk should be replaced in an actual variable name by a suitable character string, so that an actual variable name is meaningful and complies with the above restrictions.

Similarly, a lower case letter 'x' or 'y' in a variable name should be replaced in an actual variable name by one or more digits.

Additional variables not defined in Section 3 may be necessary to enable the analysis or to support traceability and should be added to ADaM datasets, providing that they do not violate the intent of the structure.

Section 3 is meant to serve as a metadata dictionary and contains few examples. Section 4 contains many examples. The rules specified in Section 4.1 govern whether derived data are to be added as columns or as rows.

3.1 SUBJECT IDENTIFIER VARIABLES

All ADaM datasets must contain the SDTM STUDYID and USUBJID variables as a minimum requirement. SDTM identifiers such as SITEID and SUBJID may optionally be included in ADaM analysis datasets. If used in analyses, sponsors should add derived identifying variables such as a pooled site variable to analysis datasets.

Table 3.1.1

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
STUDYID	Study Identifier	Char		Req	SDTM DM.STUDYID
USUBJID	Unique Subject Identifier	Char		Req	SDTM DM.USUBJID
SUBJID	Subject Identifier for the Study	Char		Perm	SDTM DM.SUBJID
SITEID	Study Site Identifier	Char		Perm	SDTM DM.SITEID. SITEID is required in ADSL but permissible in other datasets.
SITEGRP	Pooled Site Group	Char		Perm	Character description of the grouping of clinical sites for analysis purposes
SITEGRPx	Pooled Site Group x	Char		Perm	Description of a grouping of clinical sites when there are multiple types of grouping. "x" represents an integer.

3.2 TREATMENT VARIABLES

Table 3.2.1

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
TRTxP	Planned Treatment for Period x	Char		Req	The planned treatment for the subject for period x. "x" represents an integer. In a one-period

					randomized trial, TRT1P would be the treatment to which the subject was randomized. See Table 3.9.1.
TRTxPN	Planned Treatment Number for Period x	Num		Req	The numeric code for TRTxP. One-to-one map to TRTxP. Orders treatments for analysis and reporting.
TRTxA	Actual Treatment for Period x	Char		Cond	TRTxA is required if a situation occurred in the conduct of the trial where a subject received a treatment other than what was planned. Permissible in trials where all subjects received the planned treatment. See Table 3.9.1.
TRTxAN	Actual Treatment Number for Period x	Num		Cond	The numeric code for TRTxA. One-to-one map to TRTxA. If either TRTxA or TRTxAN is included, both must be included.
TRTP	Planned Treatment	Char		Cond	The planned treatment for the subject. TRTP is required when there is an analysis need for a variable representing how treatment varies by record within a subject, for example to support analysis of cross over and other designs. TRTP is optional when it would be constant within subjects, for example in a simple one-period parallel design, but in that case, at least one TRTxP, e.g. TRT1P, and its corresponding numeric equivalent TRTxPN, would be required to be copied from ADSL. See Table 3.9.1.
TRTPN	Planned Treatment Number	Num		Cond	The numeric code for TRTP. One-to-one map to TRTP. Orders treatments for analysis and reporting. If either TRTP or TRTPN is included, both must be included.
TRTA	Actual Treatment	Char		Cond	The actual treatment that the subject was given. It may be derived from the SDTM EX.EXTRT variable or some other treatment assignment data such as an IVRS database. The actual subject treatment may exist outside of clinical data inputs. Required when TRTP is included, unless actual treatment is always equal to planned. TRTA is optional when it would be constant within subjects, for example in a simple one-period parallel design, but in

					that case, at least one TRTxA, e.g., TRT1A, and its corresponding numeric equivalent TRTxAN, would be required to be copied from ADSL.
TRTAN	Actual Treatment Number	Num		Cond	The numeric code for TRTA. One-to-one map to TRTA. If either TRTA or TRTAN is included, both must be included.
TRTSEQP	Planned Treatment Sequence	Char		Cond	Required when there is a sequence of treatments that are analyzed, for example in a cross over design. TRTSEQP is not necessarily equal to ARM, for example if ARM contains elements that are not relevant to analysis of treatments. Whenever applicable, TRTSEQP is required, even if identical to ARM.
TRTSEQPN	Planned Treatment Sequence Number	Num		Cond	The numeric code for TRTSEQP. One-to-one map to TRTSEQP.
TRTSEQA	Actual Treatment Sequence	Char		Cond	TRTSEQA is required if a situation occurred in the conduct of the trial where a subject received a sequence of treatments other than what was planned. Permissible in trials where all subjects received the planned sequence of treatments.
TRTSEQAN	Actual Treatment Sequence Number	Num		Cond	The numeric code for TRTSEQA. One-to-one map to TRTSEQA. If either TRTSEQA or TRTSEQAN is included, both must be included.
TRTPGy	Planned Pooled Treatment y	Char		Perm	Planned pooled treatment y. "y" represents an integer corresponding to a particular pooling scheme. Useful when planned treatments (TRTP) are pooled together for analysis, for example when all doses of Drug A (TRTPG1=All doses of Drug A) are compared to all doses of Drug B (TRTPG1=All doses of Drug B). Each value of TRTP is pooled within at most one value of TRTPGy. May vary by record within a subject.
TRTPGyN	Planned Pooled Treatment Number y	Char		Perm	The numeric code for TRTPGy. One-to-one map to TRTPGy. If either TRTPGy or TRTPGyN is included, both must be included.
TRTAGy	Actual Pooled	Char		Cond	Actual pooled treatment y. "y" represents an integer corresponding

	Treatment y				to a particular pooling scheme. Required when TRTPGy is present and actual pooled treatments differ from planned for at least one subject. May vary by record within a subject.
TRTAGyN	Actual Pooled Treatment Number y	Char		Cond	The numeric code for TRTAGy. One-to-one map to TRTAGy. If either TRTAGy or TRTAGyN is included, both must be included.

See Section 3.9 for description of required and permissible subject-level treatment variables, any or all of which may be copied in to basic structure datasets to support traceability or enable analysis.

3.3 TIMING VARIABLES

Any SDTM timing variables (including, but not limited to, EPOCH, --DTC, --DY, VISITNUM, VISIT, and VISITDY) may and should be carried forward into analysis datasets if they would help to support data traceability back to the SDTM input data.

Table 3.3.1 defines derived timing variables for analysis datasets. Note that the timing variables whose names start with "A" are the topical analysis variables, or in other words, the timing variables directly associated with the AVAL and AVALC variables in the analysis dataset.

Names of supportive timing variables should be prefixed by a character string instead of the placeholder asterisk shown in Table 3.3.1, so that their actual names comply with the variable naming conventions described at the beginning of Section 3. In many cases, the prefix for supportive date and time variables would match that of an SDTM --DTC, --STDTC or --ENDTC variable name; for example if a supportive numeric date variable were created from --STDTC, then it would be named --STDT. However, if --STDTC is the topical analysis date (the date that characterizes AVAL and AVALC), its numeric equivalent should be named ADT.

In Table 3.3.1, imputation should be understood as including either imputation of a missing date or time component, or any change resulting in a difference between a date or time component in the ADaM numeric date, time or datetime variable, and its counterpart in the SDTM variable. Any differences must be reflected in a corresponding date or time imputation flag value.

Numeric dates and times should be formatted, such as with standard SAS date formats, so as to be human readable with no loss of precision.

The date and time imputation flag variables are not binary, since they have multiple possible values. Yet the date and time imputation flag variables do function as flags because a null value indicates no imputation, while a non-null value indicates imputation. The precise value indicates the degree of imputation.

In ADaM as in SDTM, a variable whose name ends with DY is a relative day variable in which there is no day 0. If there is a need to create a relative day variable that does include day 0, then its name must not end in DY. ADaM relative day variables need not be anchored by SDTM RFSTDTC.

Table 3.3.1

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
ADT	Analysis Date	Num		Perm	The topical analysis date in numeric format.
ADTM	Analysis Date/Time	Num		Perm	The topical analysis date/time in numeric format. When ADTM is present, it must be consistent with ADT and ATM if present.

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
ATM	Analysis Time	Num		Perm	The topical analysis time in numeric format.
ADY	Analysis Relative Day	Num		Perm	The topical analysis relative day. The number of days from a reference date (not necessarily DM.RFSTDTC) to another date. ADY is calculated in the same fashion as DY variables are in the SDTM, except that the reference date is not necessarily DM.RFSTDTC. The reference date is relative day 1 and the day before it is relative day -1. As in SDTM DY variables, there is no ADY 0.
ADTF	Analysis Date Imputation Flag	Char	Y, M, D	Cond	The level of imputation of the ADT variable based on the source SDTM DTC variable. ADTF = Y if the entire date is imputed. ADTF = M if month and day are imputed. ADTF = D if day is imputed. ADTF = null if ADT equals the SDTM DTC variable date part equivalent.
ATMF	Analysis Time Imputation Flag	Char	H, M, S	Cond	The level of imputation of the ATM variable based on the source SDTM DTC variable. ATMF = H if the entire time is imputed. ATMF = M if minutes and seconds are imputed. ATMF = S if only seconds are imputed. ATMF = null if ATM equals the SDTM DTC variable time part equivalent. For a given SDTM DTC variable, if only hours and minutes are ever collected, and seconds are imputed in ATM as 0, then it is not necessary to set ATMF to 'S'. However if seconds are generally collected but are missing in a given value of the DTC variable and imputed as 0, or if a collected value of seconds is changed in the creation of ATM, then the difference is significant and should be flagged in ATMF. Both ADTF and ATMF are needed to describe the level of imputation in ADTM.
AVISIT	Analysis Timepoint Description	Char		Cond	The topical analysis timepoint description. AVISIT may contain the same values as SDTM VISIT, but in addition, or instead, it may contain derived visit names, or time window names, or conceptual timepoint descriptions. AVISIT is a derived field and does not have to map to VISIT from the SDTM. AVISIT represents the

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
					analysis timeframe of the record, but it does not mean that the record was analyzed; ANLFL serves that purpose. There are often multiple records for the same subject and parameter that have the same value of AVISIT. ANLFL and other flags are needed to identify the records selected for any given analysis. See Section 3.7 for metadata about flag variables. AVISIT should be unique for a given analysis timepoint window and should not be blank. In the event that a record does not fall within any predefined analysis timepoint window, AVISIT should be populated with explanatory text, e.g. "Not Windowed". If either AVISIT or AVISITN is present, both must be present.
AVISITN	Analysis Timepoint Number	Num		Cond	The topical analysis timepoint number representation of AVISIT. This may be a protocol visit number, a week number, an analysis timepoint number, or any other number logically related to AVISIT or useful for sorting that is needed for analysis. There is a 1:1 correspondence between AVISITN and AVISIT. If either AVISIT or AVISITN is present, both must be present.
Supportive date variable metadata follows:					
*DT	Date of …	Num		Perm	Supportive analysis date in numeric format.
*DTM	Date/Time of …	Num		Perm	Supportive analysis date/time in numeric format. When *DTM is present, it must be consistent with *DT and *TM if present.
*TM	Time of …	Num		Perm	Supportive analysis time in numeric format.
*DY	Relative Day of …	Num		Perm	Supportive analysis relative day. The number of days from a reference date to another date. *DY is calculated in the same fashion as DY variables are in the SDTM except that the reference date is not necessarily DM.RFSTDTC. The reference date is relative day 1 and the day before it is relative day -1. As in SDTM DY variables, there is no *DY 0. Note that if the resulting variable name is the same as an existing SDTM variable name, then the label and values must be

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
					the same. For a given prefix (represented by *), if *DT is imputed (*DTF is non-null on any row), then *DY must be renamed if needed to avoid such a conflict.
*DTF	Date Imputation Flag of …	Char	Y, M, D	Cond	The level of imputation of the *DT variable based on the SDTM DTC variable. *DTF = Y if the entire date is imputed. *DTF = M if month and day are imputed. *DTF = D if day is imputed. *DTF = null if *DT equals the SDTM DTC variable date part equivalent.
*TMF	Time Imputation Flag of …	Char	H, M, S	Cond	The level of imputation of the *TM variable based on the SDTM DTC variable. *TMF = H if time is imputed. *TMF = M if minutes and seconds are imputed. *TMF = S if only seconds are imputed. *TMF = null if *TM equals the SDTM DTC variable time part equivalent. For a given DTC variable, if only hours and minutes are ever collected, and seconds are imputed in *TM as 0, then it is not necessary to set *TMF to 'S'. However if seconds are generally collected but are missing in a given value of the DTC variable and imputed as 0, or if a collected value of seconds is changed in the creation of *TM, then the difference is significant and should be flagged in *TMF. Both *DTF and *TMF are needed to describe the level of imputation in *DTM.
*VISIT	Analysis Timepoint of …	Char		Perm	Supportive analysis timepoint description. See also the description under AVISIT.
*VISITN	Analysis Timepoint Number of …	Num		Perm	Supportive analysis timepoint number representation of *VISIT. See also the description under AVISITN.

3.4 ANALYSIS PARAMETER VARIABLES

Table 3.4.1 defines analysis parameter variables for analysis datasets.

402 **Table 3.4.1**

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
PARAM	Parameter Description	Char		Req	The description of the analysis parameter. Examples include: “Supine Systolic Blood Pressure (mm Hg)”, “Log10 (Weight (kg))”, "Time to First Hypertension Event (Days)", "Estimated Tumor Growth Rate", etc. PARAM should be sufficient to describe AVAL uniquely. PARAM must include test, units, specimen type, location, position, and any other applicable qualifying information needed, any additional information such as transformation function, and indeed any text that is needed, in order that PARAM describes AVAL uniquely. PARAM may be longer than 40 characters in length.
PARAMCD	Parameter Code	Char		Req	The short name of the analysis parameter in PARAM. Values of PARAMCD should follow SAS 5 variable naming conventions (8 characters or less, starts with a letter or underscore, contains only letters, digits, and underscore). There must be a one-to-one mapping with PARAM. Examples: SYSBP, LWEIGHT, HYPEREVT.
PARAMN	Parameter Number	Num		Perm	Useful for ordering and programmatic manipulation. There must be a one-to-one mapping with PARAM. Must be an integer.
PARAMTYP	Parameter Type	Char	DERIVED	Perm	Indicator of whether the parameter is derived as a function of one or more other parameters.
PARAMCAT	Parameter Category	Char		Perm	Used to group parameters into categories. For example, to identify the parameters having to do with a particular questionnaire or lab specimen type or area of investigation.
AVAL	Analysis Value	Num		Req (at least one)	Numeric analysis value described by PARAM.
AVALC	Character Analysis Value	Char			Character analysis value described by PARAM. AVALC can be a formatted text version of AVAL but if so there should be a one-to-one map between AVAL and AVALC.
BASE	Baseline Value	Num		Cond	Baseline analysis value. If used for a given PARAM, should be populated for all records of that PARAM. Required if

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
					dataset supports analysis or review of baseline value or functions of baseline value such as change from baseline.
BASEC	Character Baseline Value	Char		Perm	Baseline value of AVALC or BASE as character. May be needed when AVALC is of interest. If used for a given PARAM, should be populated for all records of that PARAM.
BASETYPE	Baseline Type	Char		Cond	Sponsor-defined text describing the definition of baseline relevant to the value of BASE on the current record. Required when there are multiple ways that baseline is defined. If used for a given PARAM, should be populated for all records of that PARAM. Refer to Section 4.1.1, Rule 6, for an example.
CHG	Change from Baseline	Num		Perm	Change from baseline analysis value. Equal to AVAL-BASE. If used for a given PARAM, should be populated for all records of that PARAM.
CHGC	Change from Character Baseline	Char		Perm	May be needed when AVALC is of interest. This must be a one-to-one map with each unique combination of AVALC and BASEC. If used for a given PARAM, should be populated for all records of that PARAM.
PCHG	Percent Change from Baseline	Num		Perm	Percent change from baseline analysis value. Equal to ((AVAL-BASE)/BASE)*100. If used for a given PARAM, should be populated (when calculable) for all records of that PARAM.
R2BASE	Ratio to Baseline	Num		Perm	AVAL / BASE
R2ULN	Ratio to Upper Limit of Normal	Num		Perm	AVAL / ULN
R2*	Ratio to …	Num		Perm	AVAL / * where *= the name of a numeric column that contains a constant, a parameter-specific constant, or a subject-specific constant. For example, if the variable D2 is a parameter-specific constant, then the ratio of AVAL to D2 would be named R2D2. R2ULN is an exception, since the denominator ULN (upper limit of normal) may vary over time as the subject ages, and may also vary by

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
					laboratory or test method or machine, so is not necessarily a constant, nor a subject-specific constant, nor a parameter-specific constant.
SHIFT	Shift from Baseline	Char		Perm	Change in ACAT from baseline to current observation. Useful for shift tables. For example, 'NORMAL to HIGH'. ACAT is described in Section 3.6. See the example of ACAT and SHIFT in Table 4.1.1.9.
SHIFTN	Shift from Baseline	Num		Perm	Numeric version of SHIFT. One-to-one map with SHIFT.

Note that additional variables may be added that are parameter-invariant functions of AVAL and BASE on the same row. Refer to Section 4.1 for the rules governing when derivations are added as rows, and when they are added as columns.

3.5 ANALYSIS DESCRIPTOR VARIABLES

Table 3.5.1 defines analysis descriptor variables for analysis datasets:

Table 3.5.1

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
DTYPE	Derivation Type	Char	LOCF WOCF AVERAGE <others as needed>	Cond	Analysis value derivation method. DTYPE is used to denote when the value of AVAL or AVALC (and thus the entire record) has been imputed or derived from other record(s) within the same value of PARAM. DTYPE is **not** used to denote that an analysis parameter is derived. PARAMTYP may be used to indicate that an entire parameter is derived. See Section 4 for examples of DTYPE. LOCF = last observation carried forward. WOCF = worst observation carried forward. AVERAGE = average of values. DTYPE is required if any row in the dataset is derived within a parameter.

If analysis timepoints are defined by relative day windows, then the variables in Table 3.5.2 may be used along with ADY to clarify how the observation representing each analysis timepoint was chosen from among the possible

413 candidates. The observation chosen is indicated by the analyzed record flag ANLFL (see Table 3.7.1). Note that
414 the variables in Table 3.5.2 may not be applicable in all situations and are presented as an option..

415 **Table 3.5.2**

Windowing Variable Name	Windowing Variable Label	Type	Controlled Terminology	Core	Definition
AWRANGE	Analysis Window Valid Relative Day Range	Char		Perm	The range of analysis relative day (ADY) values that are valid for a given analysis timepoint (a given value of AVISIT). For example, "5-9 DAYS"..
AWTARGET	Analysis Window Target Day	Num		Perm	The target or most desired analysis relative day (ADY) value for a given value of AVISIT..
AWTDIFF	Analysis Window Diff from Target Day	Num		Perm	Absolute difference between ADY and AWTARGET. It will be necessary to adjust for the fact that there is no day 0 in the event that ADY and AWTARGET are not of the same sign. If the sign of the difference is important, then AWTDIFF might have to be used in conjunction with ADY and possibly AWTARGET when choosing among records.

416

417 ## 3.6 CATEGORICAL VARIABLES

418 The following table defines categorical analysis variables for analysis datasets:

419 **Table 3.6.1**

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
ACAT	Analysis Category	Char		Perm	A categorical representation of AVAL. Not necessarily a 1:1 map to AVAL. For example, ACAT may be used to categorize AVAL with respect to upper and lower normal limits, into low, normal, high categories. See also the example of ACAT in Table 4.1.1.9.
CRIT	Analysis Criterion	Char		Perm	A text string identifying a criterion, for example, SYSBP > 90. In some cases, the presence of the text string indicates that the criterion is satisfied on this observation, while a null value indicates that the criterion is not satisfied. In other cases, the text string identifies the criterion being evaluated, but whether or not the criterion is satisfied is indicated by the value of the variable CRITFL. See CRITFL and CRITFN in Section 3.7. Refer to Section 4.6 for a detailed

					discussion of CRIT, CRITFL and CRITFN.
CRITx	Analysis Criterion x	Char		Perm	A text string identifying a criterion, for example SYSBP > 90, when, for at least one parameter, there is more than one criterion to evaluate. In some cases, the presence of the text string indicates that the criterion is satisfied on this observation, while a null value indicates that the criterion is not satisfied. In other cases, the text string identifies the criterion being evaluated, but whether or not the criterion is satisfied is indicated by the value of the variable CRITxFL. See CRITxFL and CRITxFN in Section 3.7. Refer to Section 4.6 for a detailed discussion of CRITx, CRITxFL and CRITxFN.

3.7 INDICATOR VARIABLES

Character date imputation and time imputation flag variables are discussed in Section 3.3. Although as mentioned in that section they can be used as binary flags, their names and values do not comply with the conventions in Section 3.7.

In Section 3.7, the terms "flag" and "indicator" are synonymous, and "flag variables" are sometimes referred to simply as "flags".

Names of all character flag variables end in FL, and names of all numeric indicator variables end in FN.

Population flags must be included in the dataset if the dataset is analyzed by the given population. At least one population flag is required. All applicable subject-level population flags must also be present in ADSL.

Character and numeric subject-level population flag names end in FL and FN, respectively. Similarly, parameter-level population flag names end in PFL and PFN, and record-level population flag names end in RFL and RFN.

For character population flag variables: N = no, Y = yes. Null values are not allowed.

For numeric population flag variables: 0 = no, 1 = yes. Null values are not allowed..

In addition to the population flag variables defined in Table 3.7.1, other population flag variables may be added to ADaM analysis datasets as needed, and must comply with these conventions.

For character flags that are not population flags, nulls may be allowed, and a scheme of Y/N/null, or Y/null may be specified. As indicated in Table 3.7.1, some common character flags use the scheme Y/null. Corresponding 1/0/null and 1/null schemes apply to numeric flags that are not population indicators.

Additional flags that are not population flags may be added if their names and values comply with these conventions.

See the end of Section 3.7 for a discussion of the differences between ADaM population and baseline flags and the flags in SDTMIG 3.1.1.

443 **Table 3.7.1**

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
ABLFL	Baseline Record Flag	Char	Y, null	Cond	Character indicator to identify the baseline record for each parameter, or if there is more than one baseline definition, for each parameter and baseline type (BASETYPE). See BASETYPE in Table 3.4.1. ABLFL is required if BASE is present in the dataset.
ANLFL	Analyzed Record Flag	Char	Y, null	Cond	Character indicator of whether the record was used for analysis or not. ANLFL or ANLxFL are required if all records in the dataset are not used in an analysis. See examples in Section 4.
ANLFN	Analyzed Record Flag, Num	Num	1, null	Perm	Numeric indicator of whether the record was used for analysis or not.
ANLxFL	Analyzed Record Flag x	Char	Y, null	Cond	Character indicator of whether the record was used for the xth analysis or not. "x" represents an integer. ANLFL or ANLxFL are required if all records in the dataset are not used in an analysis. See example in Section 4.5.
ANLxFN	Analyzed Record Flag x, Num	Num	1, null	Perm	Numeric indicator of whether the record was used for the xth analysis or not. Used when one analyzed record flag is not sufficient. "x" represents an integer.
ONTRTFL	On Treatment Record Flag	Char	Y, null	Perm	Character indicator of whether the observation occurred while the subject was on treatment.
ONTRTFN	On Treatment Record Flag, Num	Num	1, null	Perm	Numeric indicator of whether the observation occurred while the subject was on treatment.
LVOTFL	Last Value On Treatment Record Flag	Char	Y, null	Perm	Character indicator of the last non-missing value on treatment for each parameter.
LVOTFN	Last Value On Treatment Record Flag, Num	Num	1, null	Perm	Numeric indicator of the last non-missing value on treatment for each parameter.
ITTFL	Intent-To-Treat Population Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the intent-to-treat population.
ITTFN	Intent-To-Treat Population	Num	0, 1	Perm	Numeric indicator of whether the subject was in the intent-to-treat

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
	Flag, Num				population.
ITTRFL	Intent-To-Treat Record-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the intent-to-treat population for the specific record.
ITTRFN	Intent-To-Treat Record-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the intent-to-treat population for the specific record.
ITTPFL	Intent-To-Treat Parameter-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the intent-to-treat population for the specific parameter.
ITTPFN	Intent-To-Treat Param-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the intent-to-treat population for the specific parameter.
SAFFL	Safety Population Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the safety population.
SAFFN	Safety Population Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the safety population.
SAFRFL	Safety Population Record-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the safety population for the specific record.
SAFRFN	Safety Population Record-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the safety population for the specific record.
SAFPFL	Safety Population Parameter-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the safety population for the specific parameter.
SAFPFN	Safety Population Param-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the safety population for the specific parameter.
FASFL	Full Analysis Set Population Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the full analysis set population.
FASFN	Full Analysis Set Population Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the full analysis set population.
FASRFL	Full Analysis Set Record-	Char	N, Y	Cond	Character indicator of whether the subject was in the full analysis set

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
	Level Flag				population for the specific record.
FASRFN	Full Analysis Set Record-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the full analysis set population for the specific record.
FASPFL	Full Analysis Set Parameter-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the full analysis set population for the specific parameter.
FASPFN	Full Analysis Set Param-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the full analysis set population for the specific parameter.
PPROTFL	Per-Protocol Population Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the per-protocol population.
PPROTFN	Per-Protocol Population Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the per-protocol population.
PPROTRFL	Per-Protocol Record-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the per-protocol population for the specific record.
PPROTRFN	Per-Protocol Record-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the per-protocol population for the specific record.
PPROTPFL	Per-Protocol Parameter-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the per-protocol population for the specific parameter.
PPROTPFN	Per-Protocol Parameter-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the per-protocol population for the specific parameter.
COMPFL	Completers Population Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the completed subjects population.
COMPFN	Completers Population Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the completed subjects population.
COMPRFL	Completers Record-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the completed subjects population for the specific record.
COMPRFN	Completers Record-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the completed subjects population for the specific record.
COMPPFL	Completers Parameter-Level Flag	Char	N, Y	Cond	Character indicator of whether the subject was in the completed subjects population for the specific parameter.

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
COMPPFN	Completers Parameter-Level Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the subject was in the completed subjects population for the specific parameter.
CRITFL	Criterion Evaluation Result Flag	Char	N, Y	Perm	Character indicator of whether the criterion defined in CRIT was met. See CRIT in Section 3.6. Refer to Section 4.6 for a detailed discussion.
CRITFN	Criterion Evaluation Result Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the criterion defined in CRIT was met.
CRITxFL	Criterion x Evaluation Result Flag	Char	N, Y	Perm	Character indicator of whether the criterion defined in CRITx was met. "x" represents an integer. See also CRITx in Section 3.7. Refer to Section 4.6 for a detailed discussion.
CRITxFN	Criterion x Evaluation Result Flag, Num	Num	0, 1	Perm	Numeric indicator of whether the criterion defined in CRITx was met. "x" represents an integer.

3.7.1 Differences Between SDTM and ADaM Population and Baseline Flags

In the SDTM Implementation Guide Version 3.1.1, Section 10.3.4 gives controlled terminology for some subject-level population flags. The conceptual mapping from those terms to ADaM indicator variables is presented in Table 3.7.1.1.

Table 3.7.1.1

SDTM QNAM	SDTM QLABEL	ADaM Indicator Variables
COMPLT	Completers Population Flag	COMPLFL / COMPLFN
FULLSET	Full Analysis Set Flag	FASFL / FASFN
ITT	Intent to Treat Population Flag	ITTFL / ITTFN
PPROT	Per Protocol Set Flag	PPROTFL / PPROTFN
SAFETY	Safety Population Flag	SAFFL / SAFFN

It is possible that the ADaM subject-level population flags might not match their conceptual counterparts in the SDTM. For example, the ITT SDTM qualifier may not match the ADaM ITTFL indicator variable for a given subject. These population indicators may not match because of operational issues. It is entirely possible that a company could inherit a SDTM database that for various reasons cannot be changed. It is not incumbent on those creating analysis datasets to go back and “fix” the SDTM population qualifiers and there may be good reason not to do so. The ADaM team agrees that it would be best if the SDTM subject-level population qualifiers are in harmony with the ADaM population indicator variables, but it is important to recognize that there may be situations where they differ. ADaM also supports parameter-level and record-level population flags, which do not exist in SDTM.

Similarly, a baseline record identified in SDTM may not be the record identified in an ADaM dataset and there are many reasons why this may occur. In SDTMIG 3.1.1, there is only one baseline record identified per --TEST,

whereas it may be necessary in an ADaM dataset to have separate baselines for combinations of --TEST and qualifiers such as --POS or --SPEC (as combined together in PARAM). For example it may be necessary to have a baseline for blood glucose and a different one for urine glucose. Additionally, there are ADaM parameters that are highly derived and do not have simple counterparts in a findings domain. Also, it may be necessary to have separate baselines for different periods within the study, for example to support analyses of change from screening baseline, double-blind treatment baseline, and open label extension baseline (see Section 4.1, Rule 6). When there is record-level population flagging, it may be necessary to have different baselines for two different analysis populations. Lastly, it may be desired to conduct analyses for different definitions of baseline. The ADaM baseline flag ABLFL, coupled with the BASE and BASETYPE columns, plus population flags, can handle all of these practical scenarios.

For analysis purposes, the authoritative values of population and baseline flags are found in the analysis datasets. ADaM flags should be described in ADaM metadata. It is not a requirement that the ADaM metadata explain any differences between ADaM and SDTM flags..

3.8 OTHER VARIABLES

3.8.1 Analysis-Enabling Variables

As stated above in Section 2.2.2, there is a class of variables that enable one or more of the analyses that the dataset was designed to support. Often, these enabling variables would include the indicator variables and analysis descriptor variables described above, which are often needed to make the analysis dataset one statistical procedure away from analysis results. Enabling variables may also include stratification and subgrouping variables, model covariates, censoring flags, and any other variables required to be present in order to perform an analysis.

If SDTM character variables are converted to numeric variables in ADaM, then they should be named as they are in the SDTM with an "N" suffix added. For example, a numeric version of the DM SEX variable would be SEXN in an ADaM dataset, and a numeric version of RACE would be named RACEN. Keep in mind the 8 character restriction on variable names and truncate the SDTM variable name as needed before appending the N. Note that this applies only to numeric variables that have a one-to-one mapping with the SDTM character variable.

If any combining of the SDTM character categories is done, the name of the derived ADaM character grouping variable should end in GRP, and the name of its numeric equivalent should end in GRPN. For example, if a character analysis variable is created to contain values of Caucasian and Non-Caucasian from the SDTM RACE variable that has 5 categories, then it should be named RACEGRP, and its numeric equivalent should be named RACEGRPN. Truncation of the original variable name may be necessary when appending suffix fragments GRP, or GRPN

3.8.2 Supportive Variables

Variables to support traceability should be included whenever practical. The variables from SDTM that serve as primary candidates for traceability are --SEQ, VISIT and VISITNUM. Although it is not a requirement that all three variables be retained, it is prudent to include as many as relevant.

In the event that the value of AVAL or AVALC is taken from a supplemental qualifier in SDTM, the two-letter domain prefix of --SEQ in the ADaM dataset is the related domain abbreviation (the value of RDOMAIN in SUPP-- or SUPPQUAL), and the value of --SEQ is the sequence number of the relevant related domain record.

Table 3.8.2.1 defines additional variables useful in certain situations to facilitate traceability. Section 4.3 contains an example.

Table 3.8.2.1

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
SRCDOM	Source Domain	Char		Perm	The 2-character identifier of the SDTM domain from which the value of AVAL or AVALC was taken.
SRCVAR	Source	Char		Perm	The name of the column (in the SDTM domain identified by SRCDOM) from

Variable Name	Variable Label	Type	Controlled Terminology	Core	Definition
	Variable				which the value of AVAL or AVALC was taken.
SRCSEQ	Source Sequence Number	Num		Perm	The sequence number SEQ of the row (in the SDTM domain identified by SRCDOM) from which the value of AVAL or AVALC was taken.

Supportive variables may also include event dates, censor dates, reason for censoring, normal ranges, and any other variables that facilitate transparency and clarity of derivations and analysis for statistical reviewers.

3.9 ADSL VARIABLES

In the ADaM Version 2.1 document, it is noted that one of the requirements of ADaM is that at a minimum the subject-level analysis dataset ADSL must always be submitted. In brief, the purpose of this ADSL analysis dataset is to provide in one location variables that describe the important attributes of a subject. The structure of ADSL is one record per subject, regardless of the type of clinical trial design. Whereas the ADaM V2.1 document discusses the likely content of ADSL, this section lists the types of variables that are required to be in every ADSL, along with standard variable names. In defining this minimum set of required variables, it is anticipated that software tools will be developed to utilize them.

Numeric dates and times should be formatted, such as with standard SAS date formats, so as to be human readable with no loss of precision.

3.9.1 Required ADSL Variables

Table 3.9.1.1 Required ADSL Variables

Variable Name	Variable Label	Type	Core	Comment
Study Identifiers				
STUDYID	Study Identifier	Char	Req	Input for these variables would most likely be the SDTM DM domain
USUBJID	Unique Subject Identifier	Char	Req	
SITEID	Study Site Identifier	Char	Req	
Subject Demographics				
AGE	Age	Num	Req	
SEX	Sex	Char	Req	
RACE	Race	Char	Req	
Population Indicator(s)				
FASFL	Full Analysis Set Population Flag	Char	Cond	A character indicator variable is required for every population that is defined in the statistical analysis plan. A minimum of one population flag variable is required for every clinical trial. Additional population flags may be added.
SAFFL	Safety Population Flag	Char	Cond	
ITTFL	Intent-To-Treat Population Flag	Char	Cond	
PPROTFL	Per-Protocol Population Flag	Char	Cond	
COMPLFL	Completers Population Flag	Char	Cond	
Treatment Variables. See Table 3.9.2 for illustration of treatment variables for various clinical trial designs.				

ARM	Description of Planned Arm	Char	Req	DM.ARM
TRTxP	Planned Treatment for Period x	Char	Req	The planned treatment for the subject for period x. "x" represents an integer. In a one-period randomized trial, TRT1P would be the treatment to which the subject was randomized. TRTxP may be derived from the SDTM DM variable ARM. At least one variable TRTxP is required.
TRTxA	Actual Treatment for Period x	Char	Cond	TRTxA is required if a situation occurred in the conduct of the trial where a subject received a treatment other than what was planned. Permissible in trials where all subjects received the planned treatment.
TRTSEQP	Planned Sequence of Treatments	Char	Cond	Required when there is a sequence of treatments that are analyzed, for example in a cross over design. TRTSEQP is not necessarily equal to ARM, for example if ARM contains elements that are not relevant to analysis of treatments. Whenever applicable, TRTSEQP is required even if identical to ARM.
TRTSEQA	Actual Sequence of Treatments	Char	Cond	TRTSEQA is required if a situation occurred in the conduct of the trial where a subject received a sequence of treatments other than what was planned. Permissible in trials where all subjects received the planned sequence of treatments.
Trial Dates				
RANDDT	Date of Randomization	Num	Cond	Required in randomized trials
TRTSTDT	Date of First Exposure to Treatment	Num	Req	
TRTENDT	Date of Last Exposure to Treatment	Num	Req	
TRTxSTDT	Date of First Exposure in Period x	Num	Cond	Required in trial designs where multiple treatments are given, such as a crossover design.
TRTxENDT	Date of Last Exposure in Period x	Num	Cond	

3.9.2 Other ADSL Variables

Numeric versions of treatment variables may be useful for logical sorting in analysis and reporting. Names of numeric versions of the treatment variables are created by appending N to the variable name of the corresponding character treatment variable. Values of the numeric versions of the treatment variables must map one-to-one to the values in the corresponding character variables.

Subject-level pooled treatment variables similar to those referred to in Table 3.2,1, but corresponding to TRTxP and

524 TRTxA, would be named TRTxPGy and TRTxAGy, where y is an integer representing the pooling scheme.

525 Treatment variables that are not necessarily subject-level and that are presented in Table 3.2.1 for use in the basic
526 structure may not be present in ADSL unless they are constant within subjects.

527 Readers are reminded that most ADSL datasets will likely contain additional permissible variables containing any
528 subject-level information that is important for the analysis, such as RFSTDTC and RFENDTC from SDTM, numeric
529 equivalents of treatment variables and population flags, stratification variables, demographic variables and subject
530 characteristics, categorical variables for use in subgrouping, duration of treatment exposure, treatment compliance
531 percentage, date of end of study, key visit dates, time at risk, indicator flags for survivor status, death, or other
532 important protocol specific events, or any other fact about the subject that is relevant to analysis or review.

533 The ADaM variable-naming conventions described previously in Section 3 are also applicable to ADSL variables.

Table 3.9.2. Abbreviated illustration of ADSL treatment variables for common trial designs.

Randomized Parallel Design

USUBJID	ARM	TRT1P	TRT1A	TRTSTDT	TRTENDT	Comments
1001	Drug X 5 mg	Drug X 5 mg	Drug X 5 mg	23OCT2007	17DEC2007	
1002	Placebo	Placebo	Placebo	19JUL2006	20SEP2007	
1003	Drug X 5 mg	Drug X 5 mg	Placebo	01NOV2007	20NOV2007	This subject was randomized to active treatment yet received Placebo instead

Two Period Cross-Over Design

USUBJID	ARM, TRTSEQP	TRT1P	TRT2P	TRTSEQA	TRT1A	TRT2A	TRT1STDT	TRT1ENDT	TRT2STDT	TRT2ENDT	Comments
1001	Placebo – Drug X	Placebo	Drug X	Placebo – Drug X	Placebo	Drug X	15FEB2006	03MAY2006	10MAY2006	15AUG2006	
1002	Placebo – Drug X	Placebo	Drug X	Placebo – Placebo	Placebo	Placebo	01MAR2006	12JUN2006	20JUN2006	23SEP2006	These subjects were each exposed to Placebo for both trial periods
1003	Drug X – Placebo	Drug X	Placebo	Placebo – Placebo	Placebo	Placebo	03FEB2006	25APR2006	01MAY2006	04AUG2006	

Three Period Cross-Over Design

USUBJID	ARM, TRTSEQP	TRT1P	TRT2P	TRT3P	TRT1STDT	TRT1ENDT	TRT2STDT	TRT2ENDT	TRT3STDT	TRT3ENDT	Comments
1001	Placebo – Drug X – Drug Y	Placebo	Drug X	Drug Y	15FEB2006	03MAY2006	10MAY2006	15AUG2006	23AUG2006	14NOV2006	In this trial, all subjects received the planned treatment at each period so the TRTxA variables are not needed
1002	Drug Y – Placebo – Drug X	Drug Y	Placebo	Drug X	01MAR2006	12JUN2006	20JUN2006	23SEP2006	01OCT2006	05DEC2006	
1003	Drug X – Drug Y – Placebo	Drug X	Drug Y	Placebo	03FEB2006	25APR2006	01MAY2006	04AUG2006	12AUG2006	15OCT2006	

Open Label Extension of a Parallel Design Study

USUBJID	ARM, TRTSEQP	TRT1P	TRT2P	TRTSTDT	TRTENDT	TRT2STDT	TRT2ENDT	Comments
1001	Drug X 5 mg - Drug X 5 mg	Drug X 5 mg	Drug X 5 mg	14AUG2007	20SEP2007	21SEP2007	15MAR2008	For open label studies, the variable TRT1P is used for the treatment to which the subject was randomized in the double blinded trial. TRT2P is used for the open label treatment.
1002	Placebo - Drug X 5 mg	Placebo	Drug X 5 mg	05JUL2007	15AUG2007	17AUG2007	04FEB2008	

542

543 **Dose Escalating Trials**

544 Dose escalating trials can be complex with respect to how exposure to each dose is analyzed. In some trials, subjects are randomized to a given dose of a treatment and then the
545 dose is escalated or de-escalated as needed. In other trials, subjects are randomized to a treatment and the dose that they first receive is not part of the randomization scheme but
546 is dependent on subject-level variables, such as weight. The maximum number of times a subject is allowed to escalate or de-escalate may be specified in the protocol. If a
547 subject is exposed to multiple doses during the course of a study, the analysis plan must clearly describe under which dose a subject's response is summarized. The approach
548 may differ for safety versus efficacy analyses. The duration of exposure experienced for each dose also may be considered. For other analysis datasets, it is assumed that
549 treatment/dose used for analysis is present on each record. For ADSL, it is difficult to present required treatment variables that will be appropriate for all dose escalating trials
550 but at a minimum, ADSL should contain the treatment/dose which is used for the subject disposition and demographic tables.

551

4 Implementation Issues and Solutions

This section illustrates the application of the required ADaM basic structure to common analysis situations that are independent of therapeutic area and type of statistical analysis. Examples are provided to illustrate the general applicability of the principles.

For space reasons, each example necessarily omits many required and permissible ADaM columns, and shows only the columns needed to facilitate understanding of the points being addressed.

4.1 CREATION OF DERIVED COLUMNS VERSUS CREATION OF DERIVED ROWS

For the purposes of Section 4, it is useful to think of the creation of an analysis dataset as occurring in two steps. The first step consists in creating a set of rows and columns more or less directly derived from or loaded from input datasets into their appropriate places. The second step consists of further derivation of additional rows and columns based on this precursor set of analysis dataset records. It is this second step that is addressed in Section 4.1.

To be specific, derived data are defined in Section 4.1 to be data that are created based on data already present in the analysis dataset, as opposed to data that are (1) copied or derived directly from external inputs, such as SDTM; or (2) copied from other analysis datasets. This section only addresses the creation of columns and rows to accommodate internally-derived data.

This section discusses the ADaM rules that govern when such internal derivation of data should result in creation of columns, and when it should result in creation of rows.

4.1.1 Rules for the Creation of Rows and Columns

Rule 1. A parameter-invariant function of AVAL and BASE on the same row that does not invalidate the description in PARAM should be added as a new column.

A new parameter (set of rows) should not be created if the information to be analyzed (the response variable) is either AVAL itself, or any parameter-invariant function of AVAL and BASE on the same row that does not invalidate the description of AVAL in PARAM. "Parameter-invariant" means that the functional form does not change from parameter to parameter and that the meaning of the function is the same on each row of the dataset. Such parameter-invariant functions of AVAL and BASE on the same row may also involve a parameter-specific or subject-specific constant (also to appear on the same row). However, if the function itself varies according to the parameter, then it may not be added as a column. Common examples of allowed columns are change from baseline (CHG), percent change from baseline (PCHG), ratio to baseline (R2BASE), and ratio to upper limit of normal (R2ULN). R2ULN is an example of an allowed function that does not involve BASE. Any number of such columns may be added. Any qualifying function is allowed. However, if the function is listed in Section 3, then the ADaM standard column name should be used. A column may not be created if its purpose is to contain a collection of parameter-specific functions.

Table 4.1.1.1. Illustration of rule 1: Creation of a column containing a same-row parameter-invariant function of AVAL and BASE

PARAM	AVISIT	ABLFL	AVAL	BASE	CHG
Weight (kg)	Screening		99	100	**-1**
Weight (kg)	Run-In		101	100	**1**
Weight (kg)	Baseline	Y	100	100	**0**
Weight (kg)	Week 24		94	100	**-6**
Weight (kg)	Week 48		92	100	**-8**
Weight (kg)	Week 52		95	100	**-5**

The function, CHG = AVAL - BASE, does not vary according to what parameter is being considered, and means the same throughout the dataset, regardless of the value of PARAM. Also, CHG is a function of AVAL and BASE on the same row, and does not invalidate the description of AVAL in PARAM. Therefore CHG is an allowable column.

The baseline flag column ABLFL identifies the row that was used to populate the BASE column.

Rule 2. A transformation of AVAL that necessitates a new description in PARAM should be added as a new parameter, and AVAL should contain the transformed value.

If the intention is to redefine AVAL, BASE, CHG, etc. in terms of a transform of AVAL, then a new parameter must be added, in which PARAM describes the transform. **The creation of a new parameter results by definition in the creation of a new set of rows.**

For example, in a change from baseline analysis of the logarithm of weight, AVAL should contain the log of weight, BASE should contain the baseline value of the log of weight, and CHG should contain the difference between the two. PARAM should contain a description of the transformed data contained in AVAL, e.g. 'Log10 (Weight (kg))'. In this way the ADaM standard accommodates an analysis of transformed data in the standard columns without creating a multiplicity of new special-purpose columns.

Table 4.1.1.2. Illustration of rule 2: Creation of a new parameter to handle a transformation

PARAM	AVISIT	AVISITN	VISITNUM	ABLFL	AVAL	BASE	CHG
Weight (kg)	Screening	-4	1		99	100	-1
Weight (kg)	Run-In	-2	2		101	100	1
Weight (kg)	Baseline	0	3	Y	100	100	0
Weight (kg)	Week 24	24	4		94	100	-6
Weight (kg)	Week 48	48	5		92	100	-8
Weight (kg)	Week 52	52	6		95	100	-5
Log10(Weight (kg))	**Screening**	**-4**	**1**		**1.9956**	**2**	**-0.0044**
Log10(Weight (kg))	**Run-In**	**-2**	**2**		**2.0043**	**2**	**0.0043**
Log10(Weight (kg))	**Baseline**	**0**	**3**	**Y**	**2**	**2**	**0**
Log10(Weight (kg))	**Week 24**	**24**	**4**		**1.9731**	**2**	**-0.0269**
Log10(Weight (kg))	**Week 48**	**48**	**5**		**1.9638**	**2**	**-0.0362**
Log10(Weight (kg))	**Week 52**	**52**	**6**		**1.9777**	**2**	**-0.0223**

In this example we see that the sponsor has chosen values of AVISITN that correspond to week number and which serve well for sorting and for plotting. VISITNUM is the SDTM visit number.

Rule 3. A function of multiple rows within the same parameter for the purpose of creating an analysis timepoint should be added as a new row for the same parameter.

For example, suppose that the analysis endpoint value is defined as the average of last two available postbaseline values. In this case, a new row should be added, with a corresponding description in AVISIT, and the DTYPE (derivation type) column should contain a description on that row such as "AVERAGE" to indicate both that the row was derived, and also the derivation method. The metadata associated with AVISIT=Endpoint should adequately describe which visits are used in the definition of the average. Note that even though the set of records for the log transformation of weight are derived, DTYPE is not populated for every row. DTYPE should be used to indicate rows that are derived within a given value of PARAM and is not to be used as an indication of whether the record exists in SDTM. Also note that the value of VISITNUM is retained on the derived records for the log transformation yet the value of VSSEQ is not retained. Since the --SEQ variables are required in SDTM and denote a specific record, it would be confusing to retain the value of VSSEQ on the derived records, since this would imply that these values of AVAL would be found in SDTM. VISITNUM, however, is populated because this variable aids traceability yet does not imply that these records are found in SDTM.

Table 4.1.1.3. Illustration of rule 3: Creation of a new row to handle a derived analysis timepoint

PARAM	AVISIT	AVISITN	VISITNUM	VSSEQ	ABLFL	AVAL	BASE	CHG	DTYPE
Weight (kg)	Screening	-4	1	1164		99	100	-1	
Weight (kg)	Run-In	-2	2	1165		101	100	1	
Weight (kg)	Baseline	0	3	1166	Y	100	100	0	
Weight (kg)	Week 24	24	4	1167		94	100	-6	
Weight (kg)	Week 48	48	5	1168		92	100	-8	
Weight (kg)	Week 52	52	6	1169		95	100	-5	
Weight (kg)	**Endpoint**	**9999**				**93.5**	**100**	**-6.5**	**AVERAGE**
Log10(Weight (kg))	Screening	-4	1			1.9956	2	-0.0044	
Log10(Weight (kg))	Run-In	-2	2			2.0043	2	0.0043	
Log10(Weight (kg))	Baseline	0	3		Y	2	2	0	
Log10(Weight (kg))	Week 24	24	4			1.9731	2	-0.0269	
Log10(Weight (kg))	Week 48	48	5			1.9638	2	-0.0362	
Log10(Weight (kg))	Week 52	52	6			1.9777	2	-0.0223	
Log10(Weight (kg))	**Endpoint**	**9999**				**1.9708**	**2**	**-0.0292**	**AVERAGE**

An extension of rule 3 is necessary in the case where there is value-level (record-level) population flagging. For example, assume the Statistical Analysis Plan states that if the subject is off drug for seven days prior to a visit, the measurement collected at that visit is not included in the per-protocol analysis. Then for some subjects, the last two available values may be different for Intent-to-Treat and for Per-Protocol analyses, so that the calculated endpoint averages would be different. For such subjects, two distinct derived endpoint rows would be needed, the appropriate row for each analysis indicated by the record-level population flags ITTRFL and PPROTRFL.

Table 4.1.1.4. Illustration of rule 3: Creation of new rows to handle a derived analysis timepoint when there is value-level population flagging

PARAM	AVISIT	AVISITN	VISITNUM	VSSEQ	ABLFL	AVAL	BASE	CHG	DTYPE	ITTRFL	PPROTRFL
Weight (kg)	Screening	-4	1	1164		99	100	-1		Y	Y
Weight (kg)	Run-In	-2	2	1165		101	100	1		Y	Y
Weight (kg)	Baseline	0	3	1166	Y	100	100	0		Y	Y
Weight (kg)	Week 24	24	4	1167		94	100	-6		Y	Y
Weight (kg)	Week 48	48	5	1168		92	100	-8		Y	Y
Weight (kg)	Week 52	52	6	1169		95	100	-5		Y	N
Weight (kg)	**Endpoint**	**9999**				**93.5**	**100**	**-6.5**	**AVERAGE**	**Y**	**N**
Weight (kg)	**Endpoint**	**9999**				**93**	**100**	**-7**	**AVERAGE**	**N**	**Y**
Log10(Weight (kg))	Screening	-4	1			1.9956	2	-0.0044		Y	Y
Log10(Weight (kg))	Run-In	-2	2			2.0043	2	0.0043		Y	Y
Log10(Weight (kg))	Baseline	0	3		Y	2	2	0		Y	Y
Log10(Weight (kg))	Week 24	24	4			1.9731	2	-0.0269		Y	Y
Log10(Weight (kg))	Week 48	48	5			1.9638	2	-0.0362		Y	Y
Log10(Weight (kg))	Week 52	52	6			1.9777	2	-0.0223		Y	N
Log10(Weight (kg))	**Endpoint**	**9999**				**1.9708**	**2**	**-0.0292**	**AVERAGE**	**Y**	**N**
Log10(Weight (kg))	**Endpoint**	**9999**				**1.9685**	**2**	**-0.0315**	**AVERAGE**	**N**	**Y**

Note that in this example, the analyzed endpoint value varies according to the population. For example, for PARAM=Weight (kg), the last two available ITT values are 92 and 95, whose average is 93.5; whereas the last two Per-Protocol values are 94 and 92, whose average is 93. That is why two derived Endpoint rows are required.

Table 4.1.1.5. Illustration of rule 3: Creation of new rows to handle imputation of missing values by Last Observation Carried Forward and Worst Observation Carried Forward

PARAM	AVISIT	AVISITN	VISITNUM	VSSEQ	ABLFL	AVAL	BASE	CHG	DTYPE	ADY	AWTARGET	AWTDIFF	ANLFL
Systolic BP (mm Hg)	Screening	-4	1	3821		120	114	6		-30	-28	2	Y
Systolic BP (mm Hg)	Run-In	-1	2	3822		116	114	2		-16	-14	2	Y
Systolic BP (mm Hg)	Week 0	0	3	3823	Y	114	114	0		-2	1	2	Y
Systolic BP (mm Hg)	Week 2	2	4	3824		118	114	4		13	14	1	Y
Systolic BP (mm Hg)	Week 2	2	4.1	3825		126	114	12		17	14	3	
Systolic BP (mm Hg)	Week 4	4	5	3826		122	114	8		23	28	5	Y
Systolic BP (mm Hg)	**Week 8**	**8**	**5**	**3826**		**122**	**114**	**8**	**LOCF**				**Y**
Systolic BP (mm Hg)	**Week 8**	**8**	4.1	**3825**		**126**	**114**	**12**	**WOCF**				**Y**
Systolic BP (mm Hg)	Week 12	12	7	3827		134	114	20		83	84	1	Y

In this example, missing post-baseline values are imputed by last observation carried forward, and also by worst observation carried forward.

For LOCF analysis, the missing Week 8 (VISITNUM 6) result is imputed by carrying forward the most recent prior available value, which is the VISITNUM 5 value. That the Week 8 value is imputed is indicated by LOCF in the derivation type (DTYPE) column.

For WOCF analysis, even though the unscheduled VISITNUM 4.1 value was not chosen to represent the Week 2 analysis timepoint, it is used to impute the missing Week 8 timepoint because it was the worst postbaseline result up to that point.

Traceability is enhanced by the addition of the SDTM VISITNUM and --SEQ columns. The combination of USUBJID and VSSEQ provides a link to the exact input record in the SDTM VS domain. On the derived LOCF and WOCF rows, VISITNUM and VSSEQ provide clarity about where the value came from.

There are several other concepts presented in this example. Analysis relative day (ADY) in this protocol is defined relative to date of first dose. In many but not all protocols, ADY would equal the value of the SDTM --DY variable. The data presented here illustrates that this particular subject did not take drug until two days after randomization, so the value of ADY is -2 at the randomization visit, Visit 3 (VISITNUM 3). As is the case for SDTM study day, there is no day 0 for ADY.

Assume that in this protocol, if there are multiple data points within an analysis time window, the value that is observed closest to a pre-specified target planned relative day is the value that is chosen to represent the analysis timepoint. For this study then, AWTARGET = VISITDY (Planned Study Day) from SDTM, and ADY=DY. AWTDIFF is the absolute value of ADY - AWTARGET, adjusted for the fact that there is no day 0. For AVISIT=Week 2, there were two values observed, at study days 13 and 17. Day 13 is closer to the target, day 14. So the day 13 record is chosen, as denoted by the analyzed record flag ANLFL = Y.

AVISIT by itself functions as a description of an analysis time window. AVISIT, DTYPE, and ANLFL are all needed to identify the records to be used in a given analysis.

Table 4.1.1.6. Illustration of rule 3: Creation of endpoint rows to facilitate analysis of a crossover design

USUBJID	PARAMCD	AVISIT	AVISITN	VISITNUM	DTYPE	ANLFL	TRT1PN	TRT2PN	TRTPN	PERIOD	TRTSEQPN	AVAL	ABLFL	BASE	CHG
0987_3984	ALT	Screening	-4	1		Y	2	1			2	16		17	-1
0987_3984	ALT	Week -2	-2	2		Y	2	1			2	16		17	-1
0987_3984	ALT	Week 0	0	3		Y	2	1			2	18		17	1
0987_3984	ALT	Baseline	-8888		**AVERAGE**	Y	2	1			2	17	Y	17	0
0987_3984	ALT	Week 4	4	4		Y	2	1	2	1	2	14		17	-3
0987_3984	ALT	Week 8	8	4.1			2	1	1	2	2	10		17	-7
0987_3984	ALT	Week 8	8	5		Y	2	1	1	2	2	12		17	-5
0987_3984	**ALT**	**Endpoint**	**9999**		**ENDPOINT**	**Y**	**2**	**1**	**2**	**1**	**2**	**14**		**17**	**-3**
0987_3984	**ALT**	**Endpoint**	**9999**		**ENDPOINT**	**Y**	**2**	**1**	**1**	**2**	**2**	**12**		**17**	**-5**
0987_4252	ALT	Screening	-4	1		Y	1	2			1	12		11	1
0987_4252	ALT	Week 0	-2	3		Y	1	2			1	11		11	0
0987_4252	ALT	Baseline	-8888		**AVERAGE**	Y	1	2			1	11	Y	11	0
0987_4252	ALT	Week 4	4	4		Y	1	2	1	1	1	14		11	3
0987_4252	ALT	Week 8	8	5		Y	1	2	2	2	1	15		11	4
0987_4252	**ALT**	**Endpoint**	**9999**		**ENDPOINT**	**Y**	**1**	**2**	**1**	**1**	**1**	**14**		**11**	**3**
0987_4252	**ALT**	**Endpoint**	**9999**		**ENDPOINT**	**Y**	**1**	**2**	**2**	**2**	**1**	**15**		**11**	**4**

This is a 2-period crossover study. The planned visits are 1 (Screening and beginning of placebo run-in period), 2 (Week -2, halfway through placebo run-in period), 3 (Week 0, end of placebo run-in and randomization), 4 (Week 4, the end of the first treatment period), and 5 (Week 8, the end of the second treatment period). Baseline is defined in the Statistical Analysis Plan as the average of the Week -2 (VISIT 2) and Week 0 (VISIT 3) measurements. USUBJID 0987_4252 has no VISIT 2 measurement, so the average is just the Week 0 (VISIT 3) measurement. Within any postbaseline week window, the last observation is used to characterize that week. For example, for USUBJID 0987_3984, the VISIT 5 value is used to characterize Week 8. The visit 4.1 value was observed during the second treatment period.

PERIOD is the crossover period number. TRT1PN and TRT2PN, planned treatment numbers for crossover periods 1 and 2, are from ADSL, as is TRTSEQPN, planned ordering of crossover treatments. TRTP is the analyzed planned treatment for the given period. The two endpoint records are derived only for the subjects who have data for both periods.

The conventions used in AVISITN are sponsor-defined. In this example, the sponsor has decided that AVISITN will contain -8888 for the derived baseline records, 9999 for the derived endpoint records, and week number otherwise.

Rule 4. A function of multiple rows within a parameter that invalidates the description in PARAM should be added as a new parameter.

For example, in a clinical trial of an HIV vaccine, blood samples are drawn at each visit, and CD4 cell count is measured. To assess efficacy, it is important to look at the cumulative effect over time on CD4 cell count during followup after administration.

Let AVAL(t) equal the value of CD4 cell count at postbaseline visit t, and let VISITDY(t) be the planned study day of visit t.

CD4AUC (cumulative daily CD4 count over followup) is calculated at any given postbaseline visit as follows:

- CD4AUC at baseline visit is set to 0.
- CD4AUC(t) = CD4AUC(t-1) + [0.5 * AVAL(t-1) + 0.5 * AVAL(t)] * [VISITDY(t) - VISITDY(t-1)].

CD4AUC is not a simple same-row function of BASE and AVAL. It is calculated based on data from multiple observations (rows) of CD4 data, so it should be added as a new parameter rather than as a new column. CD4AUC is not defined pre-baseline, which is why there is no Week -1 for this parameter.

CD4AUCMB (cumulative average change from baseline in daily CD4 count over followup) is calculated as

- CD4AUCMB(t) = CD4AUC(t) / [VISITDY(t) - 1] - baseline value of CD4 cell count.

CD4AUCMB is a function of both CD4AUC and the baseline value of CD4, so it also must be its own parameter (see rule 5 below). CD4AUCMB is not defined for pre-baseline and baseline records and therefore these records are not represented within this value of PARAM.

661 **Table 4.1.1.7. Illustration of rule 4: Creation of a new parameter to handle a function of more than one row of a parameter**

PARAM	PARAMCD	AVISIT	VISITDY	ABLFL	AVAL	BASE
CD4 (cells/mm3)	CD4	Week -1	-7		75	76
CD4 (cells/mm3)	CD4	Week 0	1	Y	76	76
CD4 (cells/mm3)	CD4	Week 2	15		128	76
CD4 (cells/mm3)	CD4	Week 4	29		125	76
CD4 (cells/mm3)	CD4	Week 8	57		191	76
CD4 (cells/mm3)	CD4	Week 12	85		167	76
CD4 (cells/mm3)	CD4	Week 16	113		136	76
CD4 Cumulative AUC	**CD4AUC**	**Week 0**	**1**	**Y**	**0**	**0**
CD4 Cumulative AUC	**CD4AUC**	**Week 2**	**15**		**1428**	**0**
CD4 Cumulative AUC	**CD4AUC**	**Week 4**	**29**		**3199**	**0**
CD4 Cumulative AUC	**CD4AUC**	**Week 8**	**57**		**7623**	**0**
CD4 Cumulative AUC	**CD4AUC**	**Week 12**	**85**		**12635**	**0**
CD4 Cumulative AUC	**CD4AUC**	**Week 16**	**113**		**16877**	**0**
CD4 Cumulative AUCMB	**CD4AUCMB**	**Week 2**	**15**		**26**	**.**
CD4 Cumulative AUCMB	**CD4AUCMB**	**Week 4**	**29**		**38.25**	**.**
CD4 Cumulative AUCMB	**CD4AUCMB**	**Week 8**	**57**		**60.125**	**.**
CD4 Cumulative AUCMB	**CD4AUCMB**	**Week 12**	**85**		**74.4167**	**.**
CD4 Cumulative AUCMB	**CD4AUCMB**	**Week 16**	**113**		**74.6875**	**.**

662

Rule 5. A function of more than one parameter should be added as a new parameter.

For example, a questionnaire total domain score is calculated as a function of more than one observed question. The total domain score should be added as a new parameter, with its corresponding set of derived rows. For this derived parameter, the value of PARAM would be e.g. 'Total Domain Score', and the value of the total domain score would be stored in the standard AVAL column, the baseline value would be stored in the standard BASE column, change from baseline would be stored in CHG, as usual.

In another example, assume that blood samples are drawn at every visit, and laboratory test measurements of total cholesterol and high-density lipoprotein cholesterol are found in the SDTM LB domain for a given study. The protocol calls for analysis of each individual lab analyte, and also for an analysis of the ratio of total cholesterol to high-density lipoprotein cholesterol. The analysis dataset would contain parameters for each of the two measured lab tests, as well as a new set of derived rows where the description in PARAM might be 'Total Cholesterol:HDL-C ratio', and AVAL contains the calculated ratio at each timepoint.

In the example in Table 4.1.1.8, the analysis of percent change from baseline (PCHG) is of interest for all three parameters and is therefore populated on all records. However, if percent change is not analyzed for a particular value of PARAM, then it is not necessary to populate PCHG for those rows.

Note that DTYPE (Derivation Type) happens to be blank for this subject on the rows corresponding to the derived parameter (PARAMCD=CHOLH). This is because DTYPE is not used to identify parameters that are derived. The function of DTYPE is to identify records that were derived from other records within the same parameter. However, if, subsequent to the creation of the PARAMCD=CHOLH rows, a DTYPE=LOCF or AVERAGE or other record had been derived from other records within CHOLH, then DTYPE would have been populated for that derived record.

If it had been desired to indicate that the parameter Total Cholesterol:HDL-C ratio had been derived from one or more other parameters, the variable PARAMTYP (Parameter Type) could have been included and set to DERIVED on that parameter's rows.

679 **Table 4.1.1.8. Illustration of rule 5: Creation of new parameter to handle a function of more than one parameter**

PARAM	PARAMCD	AVISIT	AVISITN	VISITNUM	ABLFL	AVAL	BASE	CHG	PCHG	DTYPE
Total Cholesterol (mg/dL)	CHOL	Screening	-2	1		265	266	-1	-0.376	
Total Cholesterol (mg/dL)	CHOL	Run-In	-1	2		278	266	12	4.511	
Total Cholesterol (mg/dL)	CHOL	Week 0	0	3	Y	266	266	0	0.000	
Total Cholesterol (mg/dL)	CHOL	Week 2	2	4		259	266	-7	-2.632	
Total Cholesterol (mg/dL)	CHOL	Week 4	4	5		235	266	-31	-11.654	
Total Cholesterol (mg/dL)	CHOL	Week 8	8	6		242	266	-24	-9.023	
Total Cholesterol (mg/dL)	CHOL	Week 12	12	7		217	266	-49	-18.421	
High-Density Lipoprotein Chol (mg/dL)	HDL	Screening	-2	1		44	42	2	4.762	
High-Density Lipoprotein Chol (mg/dL)	HDL	Run-In	-1	2		40	42	-2	-4.762	
High-Density Lipoprotein Chol (mg/dL)	HDL	Week 0	0	3	Y	42	42	0	0.000	
High-Density Lipoprotein Chol (mg/dL)	HDL	Week 2	2	4		43	42	1	2.381	
High-Density Lipoprotein Chol (mg/dL)	HDL	Week 4	4	5		47	42	5	11.905	
High-Density Lipoprotein Chol (mg/dL)	HDL	Week 8	8	6		46	42	4	9.524	
High-Density Lipoprotein Chol (mg/dL)	HDL	Week 12	12	7		47	42	5	11.905	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Screening**	**-2**	**1**		**6.023**	**6.333**	**-0.311**	**-4.904**	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Run-In**	**-1**	**2**		**6.950**	**6.333**	**0.617**	**9.737**	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Week 0**	**0**	**3**	**Y**	**6.333**	**6.333**	**0.000**	**0.000**	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Week 2**	**2**	**4**		**6.023**	**6.333**	**-0.310**	**-4.896**	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Week 4**	**4**	**5**		**5.000**	**6.333**	**-1.333**	**-21.053**	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Week 8**	**8**	**6**		**5.261**	**6.333**	**-1.072**	**-16.934**	
Total Cholesterol:HDL-C ratio	**CHOLH**	**Week 12**	**12**	**7**		**4.617**	**6.333**	**-1.716**	**-27.100**	

680

Rule 6. When there is more than one definition of baseline, each additional definition of baseline requires the creation of its own set of rows.

In the case there is more than one definition of baseline, new rows must be created for each additional alternative definition of baseline. There will therefore be multiple sets of rows, where each set of rows corresponds to a particular definition of baseline. Whenever there is more than one definition of baseline, the BASETYPE column is required. BASETYPE identifies the definition of baseline that corresponds to the value of BASE in each row. There is only one BASE column, and only one column for each qualifying function of AVAL and BASE.

As an alternative to stacking together the multiple sets of rows into one dataset, it is permitted to create a separate dataset for each definition of baseline. If so, BASETYPE is still required, and the column structure and other functionality of the resulting datasets should be the same.

The example in Table 4.1.1.9 presents a dataset supporting shift analysis from three different baselines. Accordingly, it makes use of the BASETYPE variable described above, as well as the ACAT and SHIFT variables.

690 **Table 4.1.1.9. Illustration of rule 6: Creation of new rows to handle multiple baseline definitions**

BASETYPE	EPOCH	AVISIT	VISIT	AVAL	LLN	ULN	ACAT	ABLFL	BASE	SHIFT	ANLFL
RUN-IN	RUN-IN	BASELINE (RUN-IN)	BASELINE	34.5	15.4	48.5	NORMAL	Y	34.5		Y
RUN-IN	RUN-IN	WEEK 8 (RUN-IN)	DAY 57	11.6	15.4	48.5	LOW		34.5	NORMAL to LOW	Y
RUN-IN	RUN-IN	END POINT (RUN-IN)	DAY 57	11.6	15.4	48.5	LOW		34.5	NORMAL to LOW	Y
RUN-IN	STABILIZATION	WEEK 14 (STAB.)	DAY 99	13.1	15.4	48.5	LOW		34.5	NORMAL to LOW	Y
RUN-IN	STABILIZATION	END POINT (STAB.)	DAY 99	13.1	15.4	48.5	LOW		34.5	NORMAL to LOW	Y
RUN-IN	DOUBLE BLIND	BASELINE (DB)	DAY 99	13.1	15.4	48.5	LOW		34.5	NORMAL to LOW	Y
RUN-IN	DOUBLE BLIND	WEEK 12 (DB)	DAY 184	13.7	15.4	48.5	LOW		34.5	NORMAL to LOW	Y
RUN-IN	DOUBLE BLIND	WEEK 12 (DB)	VISIT 98	19.7	15.4	48.5	NORMAL		34.5	NORMAL to NORMAL	
RUN-IN	DOUBLE BLIND	END POINT (DB)	VISIT 98	19.7	15.4	48.5	NORMAL		34.5	NORMAL to NORMAL	Y
RUN-IN	OPEN LABEL	BASE (OPEN)	VISIT 98	19.7	15.4	48.5	NORMAL		34.5	NORMAL to NORMAL	Y
RUN-IN	OPEN LABEL	WEEK 24 (OPEN)	DAY 169	28.1	15.4	48.5	NORMAL		34.5	NORMAL to NORMAL	Y
RUN-IN	OPEN LABEL	ENDPOINT (OPEN)	DAY 169	28.1	15.4	48.5	NORMAL		34.5	NORMAL to NORMAL	Y
DOUBLE-BLIND	DOUBLE BLIND	BASELINE (DB)	DAY 99	13.1	15.4	48.5	LOW	Y	13.1		Y
DOUBLE-BLIND	DOUBLE BLIND	WEEK 12 (DB)	DAY 184	13.7	15.4	48.5	LOW		13.1	LOW to LOW	Y
DOUBLE-BLIND	DOUBLE BLIND	WEEK 12 (DB)	VISIT 98	19.7	15.4	48.5	NORMAL		13.1	LOW to NORMAL	
DOUBLE-BLIND	DOUBLE BLIND	END POINT (DB)	VISIT 98	19.7	15.4	48.5	NORMAL		13.1	LOW to NORMAL	Y
DOUBLE-BLIND	OPEN LABEL	BASE (OPEN)	VISIT 98	19.7	15.4	48.5	NORMAL		13.1	LOW to NORMAL	Y
DOUBLE-BLIND	OPEN LABEL	WEEK 24 (OPEN)	DAY 169	28.1	15.4	48.5	NORMAL		13.1	LOW to NORMAL	Y
DOUBLE-BLIND	OPEN LABEL	END POINT (OPEN)	DAY 169	28.1	15.4	48.5	NORMAL		13.1	LOW to NORMAL	Y
OPEN LABEL	OPEN LABEL	BASE (OPEN)	VISIT 98	19.7	15.4	48.5	NORMAL	Y	19.7		Y
OPEN LABEL	OPEN LABEL	WEEK 24 (OPEN)	DAY 169	28.1	15.4	48.5	NORMAL		19.7	NORMAL to NORMAL	Y
OPEN LABEL	OPEN LABEL	END POINT (OPEN)	DAY 169	28.1	15.4	48.5	NORMAL		19.7	NORMAL to NORMAL	Y

691

Rule 7. Analysis of a parameter in different units than the SDTM standardized units requires the creation of a new parameter.

Note that in SDTM findings domains such as LB, QS, EG, etc., the STRESN column is the only numeric result column, and is also the only standardized numeric result column. The --ORRES column contains a character representation of the collected result, in the collected units specified in the --ORRESU column. The --ORRES column is not standardized. So for example, if data are typically collected in conventional units, SDTM cannot accommodate standardized data in both conventional units and Standard International units. In SDTM, for any given --TEST, a sponsor can standardize in one system of units but not two. If one wishes to be able to analyze standardized results in both conventional units and in S.I. units, a transform in an analysis dataset is needed. In each such case, a new parameter must be created in order to accommodate standardized data in the other system of units.

The description in the PARAM column must contain the units, as well as any other information such as location and specimen type that is needed to ensure that PARAM uniquely describes what is in AVAL, and differentiates between parameters as needed. PARAM cannot be the same for different units.

Sponsors may choose to add an informative column useful for grouping unit systems such as UNITSYS = < 'S.I.' | 'CONVENTIONAL'>, but the actual units are still required to be embedded in PARAM.

Rule 8. Evaluation of a criterion is handled either by creation of columns or by creation of a new parameter.

In the context of evaluating criteria, Section 4.6 discusses the creation of the CRIT, CRITFL, and CRITFN columns, which may be parameter-dependent functions of AVAL and BASE on the same row. Therefore, these allowed columns are an exception to rule 1 above, which requires parameter-invariance. In other cases, a criterion may be evaluated by the creation of a new column or a new parameter (new rows). See Section 4.6.

4.2 INCLUSION OF ALL OBSERVED AND DERIVED RECORDS FOR A PARAMETER VERSUS THE SUBSET OF RECORDS USED FOR ANALYSIS

For a given parameter, should the analysis dataset contain all records or just the subset of records that are used in the analysis?

To illustrate the issue being presented, assume that the total scores for Questionnaire A (administered at Visits 1, 2, and 3) are in the QS domain as illustrated below. Any missing total scores will be imputed by carrying the last post-baseline (post-Visit 1) total score forward. The total score for visit 3 will be analyzed.

In the SDTM QS domain data shown below, subject 0001 has data for visits 1, 2, and 3; subject 0002 will not be included in the analysis, as there are no post-baseline data for the subject; subject 0003 has data for visits 1 and 2, but is missing data for visit 3.

Table 4.2.1

Data as found in SDTM QS dataset						
DOMAIN	**USUBJID**	**VISITNUM**	**QSSEQ**	**QSCAT**	**QSTESTCD**	**QSSTRESN**
QS	0001	1	101	QUES-A	TOTSCORE	7
QS	0001	2	201	QUES-A	TOTSCORE	12
QS	0001	3	555	QUES-A	TOTSCORE	14
QS	0002	1	91	QUES-A	TOTSCORE	4
QS	0003	1	156	QUES-A	TOTSCORE	2
QS	0003	2	300	QUES-A	TOTSCORE	6

Should the analysis dataset contain data for subject 0002 even though the subject is not included in the analysis? Should the analysis dataset contain totals for visits 1 and 2 even though the data being analyzed are from visit 3?

For the purposes of discussion, there are two hypothetical options considered in this section:

1) Include in the analysis dataset only the records that are actually used in the analysis of this analysis parameter (i.e., Visit 3 records that were either observed or derived by LOCF). The main advantage of this approach would be to simplify the analysis, as no selection clause would need to be used to identify the appropriate records for inclusion in the analysis. The primary disadvantage would be the loss of traceability.

2) Include all observed and derived records for the analysis parameter in the analysis dataset. (Considering the example given above, the analysis dataset would contain observed total score rows for Subject 0001 Visits 1, 2, and 3, Subject 0002 Visit 1, and Subject 0003 Visits 1, 2. In addition it would contain a derived Visit 3 record for subject 0003. The inclusion of all analysis parameter records in the analysis dataset, including those not used in the analysis, would require that some method for identifying the records used in the specified analysis be provided. Clearly this approach would increase the size of the dataset, as well as entail a risk that users will not incorporate the appropriate selection criteria and thereby generate incorrect analysis results. An advantage would be that the inclusion of all records makes it easier to verify that the selection and derived time-point processing was done correctly, thus providing useful traceability. In addition, data would also be available to enable other analyses.

4.2.1 Recommended Solution and Examples

The ADaM-recommended solution is to include all observed and derived records for a given analysis parameter.

Rationale:

Regulatory reviewers prefer that the path followed in creating and/or selecting analysis records be clearly delineated and traceable all the way back to the originating records in the SDTM domain, if possible and within reason. Simply including the algorithm in the metadata is not usually sufficient, as any complicated data manipulations may

not be clearly identified (e.g., how missing pieces of the input data were handled). Retaining in one dataset all of the observed and derived records for the analysis parameter will provide the clearest traceability in the most flexible manner within the standard ADaM basic structure. The resulting dataset also provides the most flexibility for the regulatory reviewers in testing the robustness of an analysis (e.g. using a different imputation method).

Example 1

In the example discussed above, the analysis dataset would contain the following records for the total score parameter:

Table 4.2.1.1

PARAM	USUBJID	VISITNUM	AVISITN	AVISIT	AVAL	DTYPE
TOTSCORE	0001	1	1	Visit 1	7	
TOTSCORE	0001	2	2	Visit 2	12	
TOTSCORE	0001	3	3	Visit 3	14	
TOTSCORE	0002	1	1	Visit 1	4	
TOTSCORE	0003	1	1	Visit 1	2	
TOTSCORE	0003	2	2	Visit 2	6	
TOTSCORE	**0003**	**2**	**3**	**Visit 3**	**6**	**LOCF**

For the analysis discussed above, the data to be analyzed are selected by specifying that AVISITN = 3 (or AVISIT=Visit 3).

It should be noted that this solution does not require the inclusion of all rows from the input dataset. For example, if the input dataset contains data for several different questionnaires, the extraneous data (e.g. for questionnaires other than the one being addressed) do not have to be included in the analysis dataset.

Example 2

In the following example, the Q01 assessment is scheduled to be performed at visits 1, 3, 5, and 7, and results are to be summarized at those visits. Subject 1099 has data for the assessment at visits 1, 2, and 7. (Note that though the assessment was not scheduled to be performed at Visit 2, the data show the assessment was performed at that time.) Subject 2001 is not in the Full Analysis Set population. The SDTM dataset that is the basis for the analysis dataset has the following rows:

Table 4.2.1.2

DATA AS FOUND IN SDTM QS DATASET						
QSTESTCD	**USUBJID**	**QSSEQ**	**VISITNUM**	**VISIT**	**QSSTRESN**	**QSDTC**
Q01	1099	111	1	BASELINE	25	2005-04-04
Q01	1099	121	2	VISIT 2	24	2005-05-02
Q01	1099	132	7	VISIT 7	15	2005-08-22
Q01	2001	150	1	BASELINE	27	2005-02-05

The analysis dataset will contain records corresponding to those found in SDTM as well as records created by LOCF for the missing visit assessments, together with the flags and other columns needed to identify the records to be included in a given analysis:

772

773 **Table 4.2.1.3**

DATA AS FOUND IN ANALYSIS DATASET									
USUBJID	**VISITNUM**	**VISIT**	**AVISITN**	**AVISIT**	**AVAL**	**DTYPE**	**ANLFL**	**FASFL**	**QSSEQ**
1099	1	BASELINE	1	BASELINE	25		Y	Y	111
1099	2	VISIT 2	2	VISIT 2	24		Y	Y	121
1099	2	VISIT 2	3	VISIT 3	24	LOCF	Y	Y	121
1099	2	VISIT 2	5	VISIT 5	24	LOCF	Y	Y	121
1099	7	VISIT 7	7	VISIT 7	15		Y	Y	132
2001	1	BASELINE	1	BASELINE	27		Y	**N**	150

774

775 Selection criteria applicable to this example include:

776 • DTYPE null identifies the data as found in the SDTM dataset.

777 • DTYPE not null identifies records added for analysis purposes, while DTYPE='LOCF' specifies the method
778 used to derive the record.

779 • FASFL='Y' identifies the subjects who are members of the Full Analysis Set population.

780 • ANLFL='Y' identifies the records chosen to represent each AVISIT. There were no duplicate observations and
781 therefore in this example, all records are chosen to represent their respective analysis timepoints.

782 • “ANLFL='Y' and FASFL=’Y’ and AVISITN=5” identifies the records used in a FAS analysis of Visit 5 data.

4.3 INCLUSION OF INPUT DATA THAT ARE NOT ANALYZED BUT THAT SUPPORT A DERIVATION IN THE ANALYSIS DATASET

Section 4.2 states that for a given analysis parameter, all observed and derived rows of that parameter should be included in the dataset, not just the rows that are used in the analysis.

Section 4.2 is a special case of a more general question addressed in Section 4.3:

When an analysis dataset is created, should the analysis dataset also contain the input data used in the derivation of the analysis data? For example:

- Should input data rows and columns be included if they help to support traceability of the derivation of analyzed rows and columns?
- If raw or derived precursor parameters are not analyzed themselves but are used to derive an analyzed parameter, should the precursor parameters be included in the dataset?

The above input data rows and columns could come from one SDTM dataset or multiple datasets as necessary to derive the analysis data captured in the analysis variable or parameter.

For the purposes of discussion, there are three hypothetical options considered in this section:

1) Describe the derivation algorithms in metadata and include no input data or identification of the input data in the analysis dataset. The advantage to this approach would be simplification of the analysis dataset. However, due to the simple structure, there would be a loss of traceability between the data collected in the study (i.e., SDTM dataset) and the data analyzed (i.e., analysis dataset). Unless the derivation algorithms described in the metadata are straightforward, verification of the analysis data computation could be very challenging or even impossible.

2) Include the input data as columns in the analysis dataset. Pointer columns would be added to indicate where the input data came from – domain, variable name, and sequence number. This option would allow all pertinent input data to be retained on the relevant analyzed record, which might help simplify verification of the calculation of the analysis parameter. However, this approach would clearly increase the number of columns in the analysis dataset and would require naming the variables in a clear and concise manner. The approach also assumes that the only data to be retained are the original input values.

3) Include the input data as rows in the analysis dataset, adding columns to indicate where the input data came from – domain, variable name, and sequence number. This option would increase both the size of the dataset and the complexity of selecting the appropriate records for analysis, but it would also provide input data in an immediately accessible manner. In addition, intermediate values could be retained if appropriate flags were used to distinguish them.

4.3.1 Recommended Solution and Examples

Analysis datasets are developed to facilitate intended analyses. As they are derived from SDTM datasets most of the time, it is logical for reviewers to expect some level of traceability between SDTM dataset(s) and analysis dataset(s).

The ADaM-recommended solution to achieve the expected traceability is to describe the derivation algorithms in the metadata and, if desirable for traceability reasons and if practically feasible, to include supportive rows and columns as appropriate (combination of options 1, 2 and 3).

In general, it is strongly recommended to include as much supporting data as is needed for traceability. However, there are situations in which it may not be practical to do so. For example, if an analyzed parameter is a summary derived from a very large number of raw e-diary input records, it may be neither useful nor practical to include all of the raw e-diary records in the analysis dataset.

The remainder of this section addresses cases where the analysis datasets contain not only the analysis data but also input data that are necessary to provide clearer traceability of the algorithms used to derive the analysis data. In addition to the actual values used in the analysis, the dataset may include records not used in the analysis, records

containing input data, and records containing intermediate values computed during the derivation of the analysis data. Flags or other variables will be used to distinguish the various data types as well as to provide a traceable path from the input data to the value used in the analysis. The analysis results metadata will specify how the appropriate records are identified (either by an analyzed record flag or a specific selection clause). The identification of records used in an analysis is addressed in Sections 4.4 and 4.5.

Assuming the input data are not already present on the analysis parameter record (e.g. as covariate or supportive variable), the input data will be retained as rows in the analysis dataset. The analysis value column (AVAL and/or AVALC) on the input data row will contain the input value. Not all columns from the input dataset are carried into the analysis dataset; instead additional variables will be included indicating where the input data came from – domain, variable name, and sequence number. This approach will allow the inclusion of input data from multiple domains. If the input data are already included in columns on the analysis parameter record (e.g., as covariates or supportive information), there is no need to include additional rows for those input data. The decision on keeping the input data as rows or column will therefore be dictated by the types of input data and whether they are used for other purposes in the analysis dataset.

Rationale:

Retaining in one dataset all data used in the determination of the analysis parameter will provide the clearest traceability in the most flexible manner within the standard ADaM basic structure. This large dataset also provides the most flexibility for the regulatory reviewers in testing the robustness of an analysis.

If it is determined that this large dataset is too cumbersome, the sponsor might choose to provide two datasets, one that contains all records and another that is a subset of the first, containing only the records used in the specified analysis. Though this approach provides the needed traceability as well as providing a dataset that can be used in an analysis without specifying a selection clause, the total file size is even larger. More importantly, the developer will need to ensure consistency is maintained between the two datasets and validation will need to be done for both datasets. There is also potential confusion about which dataset supported an analysis.

Example 1

An analysis dataset is created to support time-to-event analysis of hypertension event. The analysis parameter is the study day of a hypertension event, defined to be the earliest study day among those of the following events: hospital admission, diastolic blood pressure exceeded 90, and systolic blood pressure exceeded 140.

Table 4.3.1.1

DATA AS FOUND IN SDTM VS DATASET						
USUBJID	**VISITNUM**	**VSSEQ**	**VSDTC**	**VSDY**	**VSTESTCD**	**VSSTRESN**
2010	1	22	2004-08-05	1	SYSBP	115
2010	1	23	2004-08-05	1	DIABP	75
2010	2	101	2004-08-12	8	SYSBP	120
2010	2	102	2004-08-12	8	DIABP	90
2010	3	207	2004-08-19	15	SYSBP	135
2010	3	208	2004-08-19	15	DIABP	92

Table 4.3.1.2

DATA AS FOUND IN SDTM DS DATASET					
USUBJID	**DSSEQ**	**DSSTDTC**	**DSSTDY**	**DSDECOD**	**DSTERM**
2010	25	2004-08-05	1	RANDOM	Subject Randomized
2010	99	2004-08-13	9	HOSPSTRT	Subject Hospitalized
2010	140	2004-08-15	11	HOSPEND	Subject Discharged from Hospital
2010	301	2004-08-26	22	COMPLETED	Subject Completed

The analysis dataset contains the input data used to derive the analysis parameter "HYPEREVT." The sponsor has decided that information from input records not contributing to the derivation (e.g. DIABP <=90) are not included in

the analysis dataset. Also, since SYSBP does not exceed 140 for this subject there is no SBP information in the analysis dataset for the subject.

Here are two types of analysis dataset structure one can consider.

The first possibility is illustrated in Table 4.3.1.3. In this possibility, one would include all of the sub-event records as parameters (i.e., rows) and create the input domain, input variable, and input sequence columns (SRC* columns) to identify where the input records came from. AVAL for new PARAMCD HOSPTRT is the earliest relative day of hospitalization. AVAL for new PARAMCD DBP is the earliest relative day when diastolic blood pressure exceeded 90. AVAL for new PARAMCD SBP is the earliest relative day that systolic blood pressure exceeded 140. There was no such SBP event for this patient, so there is no row for SBP for this patient. Note also that the PARAMCDs DBP and SBP are not DIABP and SYSBP, which are values of VSTEST. This was intentional, to avoid conflicts with PARAMCDs DIABP and SYSBP where AVAL would contain all of the diastolic and systolic blood pressure readings. PARAM is not shown but would clearly distinguish between parameters for blood pressure measurements from SDTM and parameters such as those being discussed, which are derived times to event. PARAMCD HYPEREVT is derived as the earliest event of the three, HOSPTRT, DBP, and SBP (the minimum AVAL of those three being the earliest relative day of the three types of events). The analysis will focus on HYPEREVT, but HOSPTRT, DBP and SBP, and indeed possibly DIABP and SYSBP, are included to support traceability, and also to enable future analysis of the sub-events should it be desired.

The main advantage of this structure is that it can handle sub-event input records from many domains in only 3 standard supportive columns. This vertical structure is preferred because it is standardized, scalable, and supports analysis of sub-events.

Table 4.3.1.3

USUBJID	PARAMCD	AVAL	SRCDOM	SRCVAR	SRCSEQ
2010	HOSPSTRT	9	DS	DSSTDY	99
2010	DBP	15	VS	VSDY	208
2010	HYPEREVT	9			

The second possibility is illustrated in Table 4.3.1.4. In this structure, all sub-events are represented in supportive column variables, and the only the compound event PARAMCD HYPEREVT is created and is available for analysis.. This structure is only practical when the sub-events are coming from only a few domains. Because all of the sub-events are shown on the same row as the compound event HYPEREVT, it is easy to verify the derivation logic.

However this approach is less scalable and standardized and thus should only be used when the reviewability of having sub-events on the same row for traceability is of paramount concern. Another drawback of this approach is that if there were a need in the future to analyze the sub-events, sub-event parameters would have to be added to have an ADaM-compliant structure supporting the analysis of sub-events. An illustration of the scalability issue is that the possibility of a tie between the earliest relative day of a diastolic blood pressure elevation and a systolic blood pressure elevation would need to be accounted for, either by metadata indicating that in the event of a tie only the diastolic blood pressure event be shown in the VSTESTCD, VSDAY and VSSEQ columns, or by the addition of three more columns, VSTSTCD2, VSDAY2, VSSEQ2 in order to support display of both events.

Table 4.3.1.4

DATA AS FOUND IN ANALYSIS DATASET								
USUBJID	PARAMCD	AVAL	DSDECOD	DSDY	DSSEQ	VSTESTCD	VSDAY	VSSEQ
2010	HYPEREVT	9	HOSPSTRT	9	99	DIABP	15	208

Example 2

The analysis parameter is glomerular filtration rate (GFR) estimated from serum creatinine using the MDRD

902 Equation (Modification of Diet in Renal Disease Study Group). The equation[1] uses plasma creatinine, BUN, and
903 albumin values from the LB domain, as well as age, race, and sex.

904 **Table 4.3.1.5**

DATA AS FOUND IN SDTM LB DATASET						
USUBJID	**VISITNUM**	**LBSEQ**	**LBTEST**	**LBTESTCD**	**LBSTRESN**	**LBSTRESU**
3000	3	98	Creatinine	CREAT	78.2	micromol/L
3000	3	115	Blood Urea Nitrogen	BUN	9.1	mmol/L
3000	3	120	Albumin	ALB	40	g/L

905

906 Additional rows are not created for the input data age, race, and sex, as they are covariates in the analysis dataset.
907 The analysis records are identified by PARAMCD=MDRD_GFR, the parameter code for PARAM = glomerular
908 filtration rate (GFR) (ml/min/1.73m**2).

909 **Table 4.3.1.6**

DATA AS FOUND IN ANALYSIS DATASET									
USUBJID	**PARAMCD**	**VISITNUM**	**AVAL**	**SRCDOM**	**SRCVAR**	**SRCSEQ**	**AGE**	**SEX**	**RACE**
3000	CREAT	3	78.2	LB	LBSTRESN	98	52	F	Black
3000	BUN	3	9.1	LB	LBSTRESN	115	52	F	Black
3000	ALB	3	40	LB	LBSTRESN	120	52	F	Black
3000	MDRD_GFR	3	76.77				52	F	Black

910

911

[1] MDRD_GFR = 170 * power(PlasmaCr, -.999) * power(Age, -.176) * Sex (1 if male, 0.762 if female) * Race (1.18 if Black, 1 otherwise) * power(BUN, -.170) * power(Albumin, .318).
Reference: Levey AS, Bosch JP, Lewis JB, et. al., A more accurate method to estimate glomerular filtration rate from serum creatinine: A new prediction equation, Ann Int Med, 1999; 130:461-470. Web-based calculator found at http://medcalc3000.com/GFREstimate.htm on 25 April, 2007.

4.4 IDENTIFICATION OF ROWS USED FOR ANALYSIS

In general, regarding statistical analysis related to analysis timepoint we have questions such as "How should rows used in a last observation carried forward (LOCF) analysis be identified?", "Should a distinct baseline row with an unique value of AVISIT always be created even if redundant with an observed value record, or should baseline just be flagged?", "Should a distinct endpoint, minimum, maximum, or average post-baseline row with a unique value of AVISIT always be created even if redundant with an observed value record, or should such analysis records just be flagged?", "Should rows used in the analysis be identified via flags or by unique values of analysis timepoint description AVISIT?". For such questions, this section discusses hypothetical solutions, presents the ADaM solution, and provides examples implementing the ADaM solution.

4.4.1 Identification of Rows Used in a Timepoint Imputation Analysis, e.g. Last/Worst Observation Carried Forward (LOCF/WOCF)

Last observation carried forward (LOCF) is one of the most commonly used timepoint-related imputation analyses. Three hypothetical approaches to "How should rows used in a last observation carried forward (LOCF) analysis be identified?" are described below. In order to cover multiple imputation analyses, both LOCF and WOCF are considered in each of the hypothetical approaches.

1) Create LOCF/WOCF rows when the LOCF/WOCF analysis timepoints are missing, and identify these imputed rows by populating the derivation type variable DTYPE with values LOCF or WOCF. All of the original rows would have null values in DTYPE. It would be very simple to select the appropriate rows for analysis by selecting DTYPE = null for Data as Observed (DAO) analysis, DTYPE = null or LOCF for LOCF analysis, and DTYPE = null or WOCF for WOCF analysis. This approach would require understanding and communication that if the DTYPE flag were not referenced correctly, the analysis would default to using all rows, including the DAO rows, plus the records derived by LOCF and WOCF. To perform a correct DAO analysis, one would need explicitly to select DTYPE = null.

2) Create a complete separate set of records for each analysis type, indicating the various analysis types by assigning unique values of the analysis timepoint description AVISIT, e.g. "Week 4", "Week 4 (LOCF)" and "Week 4 (WOCF)". This approach would make it more foolproof to perform the DAO, LOCF, and WOCF analysis in one step by referencing only AVISIT. However, because so many records would be duplicated, a very large dataset is one of the major disadvantages for this approach. In addition, this approach might be less tool-friendly, in that one might need to parse AVISIT searching for a key substring, e.g. "(LOCF)".

3) Create a flag (LOCFFL/LOCFFN) to indicate when a row is created by virtue of last observation carried forward; and similarly for WOCF. This is similar to approach 1, except that a separate flag is created for each derivation type, rather than indicating record derivation type in one column DTYPE. This approach might result in fewer records than approach 1 (for example if the WOCF record is the same as the LOCF record). In other respects, this option shares the advantages and disadvantages of approach 1.

4.4.1.1 Recommended Solution and Examples

The ADaM -recommended solution is the first approach, which is to create a LOCF or WOCF row when the LOCF or WOCF analysis timepoint is missing, and identify these imputed rows by populating the derivation type variable DTYPE with values LOCF or WOCF. This recommendation is general and is not restricted to LOCF and WOCF analysis.

Example 1: Identification of rows used in a LOCF analysis

In the example below, some subjects have complete data and others have records imputed by one method (LOCF). Subjects with no missing data have the observed number of records with all DTYPE values blank. Subject 1001 has complete data. DTYPE is blank for all records indicating they are not imputed. AVISIT matches VISIT (from SDTM). Subject 1002 is missing the Week 2 assessment. Week 2 is imputed using the LOCF method. AVISIT=Week 2 but VISIT=Week 1 so one can see where the imputed value came from in the original data. Subject 1003 is missing Week 2 and 3 data. A Data as Observed (DAO) analysis can be performed by selecting only those records where DTYPE is null. For a LOCF analysis, all records (DTYPE=null or DTYPE='LOCF') should be used.

Table 4.4.1.1.1

Row	USUBJID	VISIT	AVISIT	ADY	PARAM	AVAL	DTYPE
1	1001	Baseline	Baseline	-4	SUPINE SYSBP (mm Hg)	145	
2	1001	Week 1	Week 1	3	SUPINE SYSBP (mm Hg)	130	
3	1001	Week 2	Week 2	9	SUPINE SYSBP (mm Hg)	133	
4	1001	Week 3	Week 3	20	SUPINE SYSBP (mm Hg)	125	
7	1002	Baseline	Baseline	-1	SUPINE SYSBP (mm Hg)	145	
8	1002	Week 1	Week 1	7	SUPINE SYSBP (mm Hg)	130	
9	**1002**	**Week 1**	**Week 2**	**7**	**SUPINE SYSBP (mm Hg)**	**130**	**LOCF**
10	1002	Week 3	Week 3	22	SUPINE SYSBP (mm Hg)	135	
13	1003	Baseline	Baseline	1	SUPINE SYSBP (mm Hg)	150	
14	1003	Week 1	Week 1	8	SUPINE SYSBP (mm Hg)	140	
15	**1003**	**Week 1**	**Week 2**	**8**	**SUPINE SYSBP (mm Hg)**	**140**	**LOCF**
16	**1003**	**Week 1**	**Week 3**	**8**	**SUPINE SYSBP (mm Hg)**	**140**	**LOCF**

Example 2: Identification of rows used in both LOCF and WOCF analysis

This set of records shows a situation where there is more than one imputation method used. In this case, additional records are generated for each type of imputation. A DAO analysis can be performed by selecting only those records where DTYPE is null. For LOCF analysis, all records with DTYPE=null or DTYPE='LOCF' should be used. For WOCF analysis, all records with DTYPE=null or DTYPE='WOCF' should be used.

Table 4.4.1.1.2

Row	USUBJID	VISIT	AVISIT	ADY	PARAM	AVAL	DTYPE
1	1002	Baseline	Baseline	-4	SUPINE SYSBP (mm Hg)	145	
2	1002	Week 1	Week 1	3	SUPINE SYSBP (mm Hg)	130	
3	1002	Week 2	Week 2	9	SUPINE SYSBP (mm Hg)	138	
4	1002	Week 3	Week 3	18	SUPINE SYSBP (mm Hg)	135	
5	**1002**	**Week 3**	**Week 4**	**18**	**SUPINE SYSBP (mm Hg)**	**135**	**LOCF**
6	**1002**	**Week 2**	**Week 4**	**9**	**SUPINE SYSBP (mm Hg)**	**138**	**WOCF**
7	1002	Week 5	Week 5	33	SUPINE SYSBP (mm Hg)	130	
8	1003	Baseline	Baseline	-1	SUPINE SYSBP (mm Hg)	145	
9	1003	Week 1	Week 1	7	SUPINE SYSBP (mm Hg)	140	
10	1003	Week 2	Week 2	15	SUPINE SYSBP (mm Hg)	138	
11	**1003**	**Week 2**	**Week 3**	**15**	**SUPINE SYSBP (mm Hg)**	**138**	**LOCF**
12	**1003**	**Week 2**	**Week 4**	**15**	**SUPINE SYSBP (mm Hg)**	**138**	**LOCF**
13	**1003**	**Week 2**	**Week 5**	**15**	**SUPINE SYSBP (mm Hg)**	**138**	**LOCF**
14	**1003**	**Week 1**	**Week 3**	**7**	**SUPINE SYSBP (mm Hg)**	**140**	**WOCF**
15	**1003**	**Week 1**	**Week 4**	**7**	**SUPINE SYSBP (mm Hg)**	**140**	**WOCF**
16	**1003**	**Week 1**	**Week 5**	**7**	**SUPINE SYSBP (mm Hg)**	**140**	**WOCF**

4.4.2 Identification of Baseline Rows

Many statistical analyses require the identification of a baseline value. How should a record used as a baseline be identified? The two hypothetical approaches below are considered.

1) Always provide a row with unique value of AVISIT, e.g., "Baseline", designating the baseline record used for analysis, even if redundant with another record. This approach would have the advantage that one would not need to know protocol-specific details in order to identify the baseline record, and one would not need an additional flag. This option represents the general case, since baseline might be derived as e.g. an average of other rows. The disadvantage is that this approach may entail the creation of redundant rows.

2) Create a baseline flag column to indicate the row used as baseline to populate the BASE value. This scenario results in smaller dataset, as it does not require duplication of records in the event that the baseline record is not derived.

4.4.2.1 Recommended Solution and Examples

The ADaM-recommended solution is approach 2: that a baseline record flag variable ABLFL be created and used to identify the record that is the baseline record. However, this does not prohibit one from doing both. For more complicated baseline definitions (functions of multiple rows), a derived baseline record would have to be created in any case.

Example 1: Identification of baseline rows - using screening visit to impute a baseline row

This example shows the creation of a baseline flag variable ABLFL and a row with unique value of AVISIT designating the baseline record used for analysis by creating an additional row for baseline analysis timepoint when needed. Subject 1001 had complete data. Subject 1002 missed the Baseline visit. A derived baseline record (AVISIT='Baseline') is added with DTYPE="BASELINE" to indicate that the record is imputed to be used as baseline.

Table 4.4.2.1.1

Row	USUBJID	VISIT	AVISIT	ADY	ABLFL	PARAM	AVAL	DTYPE
1	1001	Screening	Screening	-12		SUPINE SYSBP (mm Hg)	144	
2	1001	Baseline	Baseline	1	Y	SUPINE SYSBP (mm Hg)	145	
3	1001	Week 1	Week 1	6		SUPINE SYSBP (mm Hg)	130	
4	1001	Week 2	Week 2	12		SUPINE SYSBP (mm Hg)	133	
5	1002	Screening	Screening	-14		SUPINE SYSBP (mm Hg)	144	
6	**1002**	**Screening**	**Baseline**	-14	**Y**	**SUPINE SYSBP (mm Hg)**	**144**	**BASELINE**
7	1002	Week 1	Week 1	8		SUPINE SYSBP (mm Hg)	130	
8	1002	Week 2	Week 2	14		SUPINE SYSBP (mm Hg)	133	

Example 2: Identification of baseline rows - using an average of multiple visits to derive a baseline row

This example shows the creation of a baseline flag variable ABLFL, and the use of it to identify the record used for analysis. Row 3 is a derived "Baseline" record using the average of the values of row 1 and row 2. DTYPE = "AVERAGE" to indicate that row 3 is derived. The Baseline flag (ABLFL="Y") indicates that AVAL from row 3 is used to populate the BASE (Baseline) column. VISIT (from SDTM) is left blank on row 3 since AVAL on that record is not merely a copy of AVAL on another record.

Table 4.4.2.1.2

Row	USUBJID	VISIT	AVISIT	ADY	ABLFL	PARAM	AVAL	BASE	DTYPE
1	1001	Screening	Screening	-12		SUPINE SYSBP (mm Hg)	144	144.5	
2	1001	Baseline	Baseline	1		SUPINE SYSBP (mm Hg)	145	144.5	
3	1001		**Baseline**		**Y**	SUPINE SYSBP (mm Hg)	144.5	144.5	**AVERAGE**
4	1001	Week 1	Week 1	12		SUPINE SYSBP (mm Hg)	130	144.5	
5	1001	Week 2	Week 2	-14		SUPINE SYSBP (mm Hg)	133	144.5	

Example 3: Identification of baseline rows - using an average of multiple visits to derive a baseline row

This example is the same as Example 2 except that the analysis timepoint description "Average Baseline" helps differentiate the derived average baseline record from an existing observed record whose timepoint description is "Baseline." This was helpful in analysis and reporting because it was desired to summarize all scheduled visits in addition to the average baseline visit. The analysis was straightforward using the distinct descriptions of AVISIT.

Table 4.4.2.1.3

Row	USUBJID	VISIT	AVISIT	ADY	ABLFL	PARAM	AVAL	BASE	DTYPE
1	1001	Screening	Screening	-12		SUPINE SYSBP (mm Hg)	144	144.5	
2	1001	Baseline	Baseline	1		SUPINE SYSBP (mm Hg)	145	144.5	
3	1001		**Average Baseline**		**Y**	SUPINE SYSBP (mm Hg)	144.5	144.5	**AVERAGE**
4	1001	Week 1	Week 1	12		SUPINE SYSBP (mm Hg)	130	144.5	
5	1001	Week 2	Week 2	-14		SUPINE SYSBP (mm Hg)	133	144.5	

4.4.3 Identification of Post-Baseline Conceptual Timepoint Records

When analysis involves cross-timepoint derivations such as endpoint, minimum, maximum and average post-baseline, we need to consider questions such as "Should distinct rows with unique value of AVISIT always be created even if redundant with an observed value record, or should these rows just be flagged?" There are two hypothetical solutions discussed below.

1) Always create a row with unique value of AVISIT designating the record used for analysis, e.g. "Endpoint", "Post-Baseline Minimum", "Post-Baseline Maximum", "Post-Baseline Average", etc. This option would have the advantage that once the AVISIT values are understood, reviewers and computer tools could rely on these values of AVISIT. This option represents the general case since any such cross-timepoint derivation can be represented in a new row with a unique AVISIT description. The disadvantage is that the dataset would contain more rows, and conventions would have to be communicated and understood.

2) Create flag columns to indicate the row used as “Endpoint”, “Post-Baseline Minimum”, “Post-Baseline Maximum” or “Post-Baseline Average”. This option might result in a smaller dataset, because it would not require the addition of new rows if not needed. However, it requires understanding and correct use of flags.

4.4.3.1 Recommended Solution and Examples

The ADaM-recommended solution is option 1, that a new row be created with a unique value of AVISIT. The advantage of this option is that this structure is simple and it is analysis friendly. It is recognized that such new rows might be redundant with observed records for some kinds of conceptual timepoint definitions.

Example 1: Identification of Endpoint rows

This example shows the creation of a derived row with unique value of AVISIT designating the Endpoint record used for analysis. Subject 1001 discontinued at Week 2, and a derived Endpoint record (AVISIT='Endpoint') is added using the Week 2 visit. DTYPE="ENDPOINT" indicates the AVISIT="Endpoint" record is derived. Subject 1002 did not have any post-baseline visits, and therefore has no Endpoint record.

Table 4.4.3.1.1

Row	USUBJID	VISIT	AVISIT	ADY	PARAM	AVAL	DTYPE
1	1001	Screening	Screening	-12	SUPINE SYSBP (mm Hg)	144	
2	1001	Baseline	Baseline	1	SUPINE SYSBP (mm Hg)	145	
3	1001	Week 1	Week 1	6	SUPINE SYSBP (mm Hg)	130	
4	1001	Week 2	Week 2	12	SUPINE SYSBP (mm Hg)	133	
5	**1001**	**Week 2**	**Endpoint**	12	**SUPINE SYSBP (mm Hg)**	**133**	**ENDPOINT**
6	1002	Screening	Screening	-14	SUPINE SYSBP (mm Hg)	144	
7	1002	Baseline	Baseline	-1	SUPINE SYSBP (mm Hg)	144	

Example 2: Identification of Endpoint and Post-Baseline Minimum, Maximum, and Average rows

This example shows the creation of rows with unique values of AVISIT designating the Endpoint record, and the Post-Baseline Minimum, Maximum, and Average records. Subject 1001 had minimum post-baseline result at Week 1, maximum post-baseline result at Week 2, and the average post-baseline result was based on the average of Week 1 and Week 2. This subject discontinued at Week 2. A derived Endpoint record (AVISIT='Endpoint') is added using the Week 2 visit. DTYPE="ENDPOINT" indicates that the AVISIT="Endpoint" record is a derived record. Subject 1002 did not have any post-baseline visit. Therefore, the Post-Baseline Minimum, Post-Baseline Maximum, Post-Baseline Average, and Endpoint records could not be derived for that subject.

Table 4.4.3.1.2

Row	USUBJID	VISIT	AVISIT	ADY	PARAM	AVAL	DTYPE
1	1001	Screening	Screening	-12	SUPINE SYSBP (mm Hg)	144	
2	1001	Baseline	Baseline	1	SUPINE SYSBP (mm Hg)	145	
3	1001	Week 1	Week 1	6	SUPINE SYSBP (mm Hg)	130	
4	1001	Week 2	Week 2	12	SUPINE SYSBP (mm Hg)	133	
5	**1001**	**Week 1**	**Post-Baseline Minimum**	**6**	**SUPINE SYSBP (mm Hg)**	**130**	**MINIMUM**
6	**1001**	**Week 2**	**Post-Baseline Maximum**	**12**	**SUPINE SYSBP (mm Hg)**	**133**	**MAXIMUM**
7	**1001**		**Post-Baseline Average**		**SUPINE SYSBP (mm Hg)**	**131.5**	**AVERAGE**
8	**1001**	**Week 2**	**Endpoint**	12	**SUPINE SYSBP (mm Hg)**	**133**	**ENDPOINT**
9	1002	Screening	Screening	-14	SUPINE SYSBP (mm Hg)	144	
10	1002	Baseline	Baseline	-1	SUPINE SYSBP (mm Hg)	144	

4.4.4 Identification of Rows Used for Analysis – General Case

It is important to identify the rows used in or excluded from analysis. Should rows used in the analysis be identified via flags or by unique values of analysis timepoint window description AVISIT? There are two hypothetical solutions discussed below.

1) Create unique values of timepoint window description AVISIT. For example, add an asterisk to the end of AVISIT such as “Week 2 *” if not analyzed. This option might be less confusing because the user would not need to be aware of a flag. The disadvantage is that one would need to have a convention for AVISIT values, and tools might need to parse values of AVISIT for correct results to be generated.

2) Create an analyzed record flag column such as ANLFL=Y or ANLFN=1 to indicate the rows that are used for analysis. This option would allow multiple rows within a parameter with the same value of AVISIT. However, it would require flags to be added to the dataset to be used in selecting appropriate rows for analysis. Understanding of the flags would be required for correct analysis results to be generated. In addition to ANLFL/ANLFLN, additional flags might also be required, such as record-based population flags e.g. ITTRFL/ITTRFN and PPROTRFL/PPROTRFN.

4.4.4.1 Recommended Solution and Examples

The ADaM-recommended solution is to use an analyzed record flag (ANLFL/ANLFN) to indicate which records

were analyzed. ANLFL=Y (ANLFN=1) for analyzed records and is blank (null) in unused records such as a duplicate observation that was not analyzed, or pre-specified post study timepoints not used for analysis.

Example 1: Identification of rows used for analysis – multiple visits that fall within a visit window

This example shows the analysis flag variable ANLFL used to indicate the rows that were chosen for analysis from among the multiple visits that fall within the analysis timepoint windows of "Baseline" and "Week 2". Subject 1001 had two observed Baseline and Week 2 analysis timepoints according to analysis window definitions. The one that is used in analysis is flagged with ANLFL=Y. This approach is used because all original visits (records) are included in the dataset, and those selected for analysis must be identified. For traceability reasons, it is also recommended to add the AW* columns presented in Section 3.5 if appropriate, in order to indicate more clearly how the analyzed records were selected from among the candidate records within each analysis window.

Table 4.4.4.1.1

Row	USUBJID	VISIT	AVISIT	ADY	PARAM	AVAL	DTYPE	ANLFL
1	1001	Screening	Baseline	-5	SUPINE SYSBP (mm Hg)	144		
2	1001	Baseline	Baseline	1	SUPINE SYSBP (mm Hg)	145		Y
3	1001	Week 1	Week 1	7	SUPINE SYSBP (mm Hg)	130		Y
4	1001	Week 2	**Week 2**	12	SUPINE SYSBP (mm Hg)	133		Y
5	1001	Week 3	**Week 2**	17	SUPINE SYSBP (mm Hg)	125		
6	1001	Week 4	Week 4	30	SUPINE SYSBP (mm Hg)	128		Y

Example 2: Identification of rows used for analysis – visit falls outside of a target window

In this example, the Week 3 visit for subject 1001 was outside the day window of analysis Week 3, so "Post Study" was assigned to AVISIT. This visit as well as the first baseline visit were excluded from the analysis. The "Worst Post Baseline" analysis timepoint (Row 6) was imputed by worst observation carried forward (DTYPE=WOCF). The "Endpoint" row was derived using the "Week 2" visit, since it was the last available eligible observation based on the Statistical Analysis Plan. Both of the derived rows are flagged with ANLFL=Y since they were rows used for analysis.

Table 4.4.4.1.2

Row	USUBJID	VISIT	AVISIT	ADY	VISITDY	PARAM	AVAL	DTYPE	ANLFL
1	1001	Screening	Baseline	-5	1	SUPINE SYSBP (mm Hg)	144		
2	1001	Baseline	Baseline	1	1	SUPINE SYSBP (mm Hg)	145		Y
3	1001	Week 1	Week 1	7	7	SUPINE SYSBP (mm Hg)	150		Y
4	1001	Week 2	Week 2	12	14	SUPINE SYSBP (mm Hg)	133		Y
5	1001	Week 3	**Post Study**	**40**	**21**	SUPINE SYSBP (mm Hg)	140		
6	1001	Week 1	Worst Post Baseline	7	7	SUPINE SYSBP (mm Hg)	150	WOCF	Y
7	1001	Week 2	Endpoint	12	14	SUPINE SYSBP (mm Hg)	133	ENDPOINT	Y

Example 3: Identification of rows used for analysis – a visit not flagged for main analysis is used to create imputed LOCF records

This example shows two visits that occur within a window (Week 2). The first record (on row 4) is analyzed as is (it is the record chosen to represent analysis timepoint Week 2). The second Week 2 timepoint record (on row 5) is the basis for the LOCF derivation of analysis timepoints Week 3, 4 and 5 (rows 6, 7 and 8). In the LOCF analysis, Week 2 is based on the observed data on row 4, and Weeks 3, 4, 5 are imputed using the last available observation on row 5.

Table 4.4.4.1.3

Row	USUBJID	VISIT	AVISIT	ADY	PARAM	AVAL	DTYPE	ANLFL
1	1001	Screening	Baseline	-5	SUPINE SYSBP (mm Hg)	144		
2	1001	Baseline	Baseline	1	SUPINE SYSBP (mm Hg)	145		Y
3	1001	Week 1	Week 1	7	SUPINE SYSBP (mm Hg)	130		Y
4	**1001**	**Week 2**	**Week 2**	12	**SUPINE SYSBP (mm Hg)**	**133**		**Y**
5	**1001**	**Week 3**	**Week 2**	17	**SUPINE SYSBP (mm Hg)**	**125**		
6	1001	Week 3	Week 3	17	SUPINE SYSBP (mm Hg)	125	LOCF	Y
7	1001	Week 3	Week 4	17	SUPINE SYSBP (mm Hg)	125	LOCF	Y
8	1001	Week 3	Week 5	17	SUPINE SYSBP (mm Hg)	125	LOCF	Y

Example 4: Identification of rows used for analysis by creating record-level population flags

The example below shows record-based population analysis. The population flag variables indicate whether a given record is in or out of the relevant per-population analysis. The Week 1 visit for subject 1001 is not qualified for ITT analysis because it was too close to the Baseline visit (ADY <= 3). The visits of Screening, Week 1 and Week 2 are not qualified for per-protocol analysis because the assessments were not within +/- 3 days of target (VISITDY).

Table 4.4.4.1.4

Row	USUBJID	VISIT	AVISIT	ADY	VISITDY	PARAM	AVAL	ITTRFL	PPROTFL
1	1001	Screening	Screening	-7	1	SUPINE SYSBP (mm Hg)	144	Y	N
2	1001	Baseline	Baseline	1	1	SUPINE SYSBP (mm Hg)	145	Y	Y
3	1001	Week 1	Week 1	3	7	SUPINE SYSBP (mm Hg)	130	N	N
4	1001	Week 2	Week 2	10	14	SUPINE SYSBP (mm Hg)	133	Y	N
5	1001	Week 3	Week 3	20	21	SUPINE SYSBP (mm Hg)	125	Y	Y
6	1001	Week 4	Week 4	30	28	SUPINE SYSBP (mm Hg)	128	Y	Y

4.5 IDENTIFICATION OF POPULATION-SPECIFIC ANALYZED RECORDS

It is not uncommon in the statistical analysis of clinical trials to repeat analyses based on multiple populations of interest. The population of interest can be defined either at the subject level or at the record (measurement) level. For example, when defining an analysis population, a subject may be included in one analysis population such as Intent-to-Treat but may be excluded from another analysis population such as Per-Protocol. Analysis populations may also be defined using characteristics of individual measurements. For example, a measurement that was obtained outside of a pre-specified window for a particular visit may not be included in a per-protocol visit-level population. In this section, it is assumed that the definition of a record level analysis population is dependent on the definition of the subject level population. In other words, if a subject is excluded from the subject-level Per-Protocol population, then none of their individual records would be candidates for an analysis based on the record-level Per-Protocol definition. Given these various levels of population definitions, as well as multiple types of populations defined, the same record in an analysis data set could be included or excluded for an analysis depending on characteristics of the subject as a whole and the characteristics of the individual measurement. Therefore, the issue becomes how best to indicate records that are selected for each analysis.

4.5.1 Recommended Solution and Examples

The ADaM-recommended solution to this analysis issue is to have one analysis dataset that can be used to perform all analyses using population specific indicator variables to identify records that are used for each type of analysis. The advantage of this solution is that the one analysis dataset can be used for multiple analyses and the use of flag variables obviates the need to replicate rows for each type of analysis. This promotes efficiency in the operational aspects of electronic submissions, clarity of analyses, and ease for FDA reviewers to compare selected values for each population. This solution does, however, require that clear metadata be provided for the indicator variable so that each specific analysis can be reproduced accurately. Below are several examples of the use of population specific indicator variables to identify records used for different analyses.

Example 1:

Table 4.5.1.1

USUBJID	ITTFL	PPROTFL	VISIT	AVISIT	PARAMCD	AVAL	ANLFL	ITTRFL	PPROTRFL
1001	Y	Y	Week 0	Week 0	TEST1	500	Y	Y	Y
1001	Y	Y	Week 1	Week 1	TEST1	400	Y	Y	Y
1001	Y	Y	Week 2	Week 2	TEST1	600	Y	Y	Y
1002	Y	N	Week 0	Week 0	TEST1	500	Y	Y	N
1002	Y	N	Week 2	Week 1	TEST1	48	Y	Y	N
1002	Y	N	Week 2	Week 2	TEST1	46	Y	Y	N
1003	Y	Y	Week 0	Week 0	TEST1	999	Y	Y	Y
1003	Y	Y	Week 1	Week 1	TEST1	999		Y	Y
1003	Y	Y	Retest	Week 1	TEST1	49	Y	Y	N
1003	Y	Y	Week 2	Week 2	TEST1	499	Y	Y	N

The columns ITTFL and PPROTFL are the analysis population flags that identify if a subject is an Intent-to-Treat subject or a Per-Protocol subject at the subject level. If a subject is an Intent-to-Treat subject, then the column ITTFL will have the value of 'Y'. In the above example, subjects 1001, 1002, and 1003 are all Intent-to-Treat subjects. Similarly, if a subject is a Per-Protocol subject, the column PPROTFL will have the value of 'Y'. Subjects 1001 and 1003 in the above example are Per-Protocol subjects while subject 1002 with PPROTFL=N is excluded from any Per-Protocol analysis. These indicator variables are used to identify individual subjects that belong to each subject-level population.

In contrast to the subject-level population flags, the columns ITTRFL and PPROTRFL are the analysis flags at the record level. If a record is eligible for the Intent-to-Treat analysis, the variable ITTRFL is set to 'Y'; it is set to 'N' if the record is not a candidate for this analysis. In the above example, all records under the column ITTRFL are all set to 'Y'. Similarly, if a record is a candidate for the Per-Protocol analysis, the variable PPROTRFL is set to 'Y', it is set to 'N' if the record does not fulfill the criteria for this analysis. In the example above, all three records for subject 1002 and two of four records for subject 1003 are not record-level Per-Protocol data and would not be selected for a Per-Protocol analysis when we apply the subset condition: PPROTRFL='Y'.

Depending on the purpose of a statistical analysis, even if a subject is included in the Per-Protocol population, some or all data for that subject in a particular data set may not be appropriate for a per-protocol analysis. Consider a situation where a Per-Protocol analysis excludes all data after the date of last dose of study drug. The last dose for subject 1003 in the above example is at Week 1, so the data at Retest and Week 2 will have a value of 'N' under column PPROTRFL and will be excluded from any record-level Per-Protocol data analysis.

The analyzed record flag ANLFL is “N' for one record (USUBJID=1003, VISIT=WEEK1, AVISIT=WEEK1, AVAL=999) because its value was replaced for analysis purposes by the retest result in the next record (USUBJID=1003, VISIT=Retest, AVISIT=WEEK1, AVAL=49). The analyzed record flag for the Retest record is Y.

To identify records used for an Intent-to-Treat analysis for parameter code 'TEST1' at Week 1 requires the following selection specification:

ITTFL='Y' & AVISIT='Week 1' & PARAMCD='TEST1' & ANLFL='Y' & ITTRFL='Y';

Similarly, to identify records used for a Per-Protocol analysis of values of TEST1 <=400 the selection specification becomes:

PPROTFL='Y' & AVISIT='Week 1' & PARAMCD='TEST1' & ANLFL='Y' & PPROTRFL='Y';

Since an error in the specification of the selection for either of the above conditions will yield incorrect results, it is important that the metadata be clear for each indicator variable. In addition, it is also recommended that the ADaM analysis results metadata be used so that specific analyses summarized in statistical tables in a study report be accompanied by subsetting selection statements to provide clear documentation of how the indicator variables were used to select analyzed records.

In this example, just one analyzed record flag, ANLFL, is sufficient for both analyses. Alternatively, one could have chosen to simplify the selection criteria through the use of the ANLxFL variables, one for each of the analysis types. In this case, the standard variables ANLxFL should be used and the metadata associated with these variables should clearly state which analysis is associated with each analysis flag.

Taking that approach, ANL1FL would indicate the records to include in the ITT analysis, and ANL2FL would indicate the records to include in the PP analysis. ANL1FL would be 'Y' only when ITTFL, ITTRFL, and ANLFL were all 'Y', and ANL2FL would be Y only for the records where PPROTFL, PPROTRFL, and ANLFL were all 'Y'. The above selection criteria could thus be simplified, respectively, as

ANL1FL = 'Y' & AVISIT='Week 1' & PARAMCD='TEST1'

and

ANL2FL = 'Y' & AVISIT='Week 1' & PARAMCD='TEST1'.

4.6 IDENTIFICATION OF RECORDS WHICH SATISFY A PREDEFINED CRITERION FOR ANALYSIS PURPOSES

For analysis purposes, criteria are often defined to group results based on the collected value's relationship to one or more algorithmic conditions. For example, how many subjects had a result greater than five times the upper limit of the normal range? Or how many subjects had a systolic blood pressure value > 160 mm Hg with at least a 25 point increase from the BASE value? In addition to creating subgroups of subjects, the categorization of the presence or absence of a criterion is often used in listings, tabular displays or statistical modeling (as a covariate or a response variable).

4.6.1 Recommended Solution and Examples

The ADaM-recommended solution is to use a the category criterion variable, CRIT, to identify whether a criterion is met. Related sections of the Standard ADaM Metadata (in this document) are as follows:

Section 3.6 Categorical Variables

- CRIT Category criterion

Section 3.7 Indicator Variables

- CRITFL Companion variable to CRIT
- CRITFN Companion variable to CRIT

As described in Section 3.6, if there is a need for more than one CRIT column, the columns should be named CRIT1-CRITn, where n is an integer. As described in Section 3.7, CRIT1FL-CRITnFL and CRIT1FN-CRITnFN may also be included.

CRIT is populated with a text description defining the conditions necessary to satisfy the presence of the criterion. The definition of CRIT can use any variable(s) located on the row and the definition must stay constant across all rows with the same value of PARAM. CRITFL and CRITFN are thus not parameter-invariant.

Example 1: CRIT without companion variable(s)

When a criterion is defined for a PARAM but conditions are not met on a specific row, CRIT is left blank. CRIT would also be left blank when one or more data inputs to a criterion are missing, resulting in an undefined criterion assessment.

When the three following conditions are true for a specific parameter, then CRIT is used for subsetting within the parameter.

1. Variable CRIT is present in the dataset;
2. Analysis Variable Metadata defines CRIT relative to the specific parameter;
3. Companion variables CRITFL and CRITFN are not present on the dataset, or one or both are present but have null values on all rows for the parameter.

Table 4.6.1.1

USUBJID	PARAM	AVAL	BASE	CHG	CRIT	Comment
1001	Systolic Blood Pressure (mm Hg)	163	148	15	Systolic Pressure >160	Presence of a value in CRIT indicates this subject satisfied the criterion. Since companion variables are missing, this criterion is used only for subsetting purposes when the interest is in the subgroup of subjects who fulfilled the criterion.
1002	Systolic Blood Pressure (mm Hg)	140	148	-8		Absence of a value in CRIT indicates this subject did not satisfy the criterion, or the criterion assessment is undefined due to missing inputs.

Example 2: CRIT with companion variable(s)

The companion variables are included and populated with non-null values when the criterion is used in tabular displays and/or statistical modeling for the parameter.

Table 4.6.1.2

USUBJID	PARAM	AVAL	BASE	CHG	CRIT	CRITFL	Comment
1001	Systolic Blood Pressure (mm Hg)	163	148	15	Systolic Pressure >160 and Change from Baseline in Systolic Pressure>10	Y	Presence of at least one of the companion variables and population of non-null values in the companion variable for this parameter indicates that this criterion is used for statistical modeling and/or tabular displays for this parameter.
1002	Systolic Blood Pressure (mm Hg)	140	148	-8	Systolic Pressure >160 and Change from Baseline in Systolic Pressure>10	N	Since this criterion is used for modeling or analysis, it is necessary to populate the rows which fail to satisfy the criterion
1005	Systolic Blood Pressure (mm Hg)	120			Systolic Pressure >160 and Change from Baseline in Systolic Pressure>10		The criterion has missing input(s).

Example 3: Compound criteria

If the definition of a criterion uses values located on multiple rows (different parameters or multiple rows for a single parameter), then a new row should be added with the value of PARAM being the textual description of the criterion.

Table 4.6.1.3

USUBJID	PARAM	AVAL	AVALC	BASE	CHG	CRIT	Comment
1001	Systolic Blood Pressure (mm Hg)	163		148	15	Systolic Pressure >160	This criterion is used alone for subsetting purposes
1001	Diastolic Blood Pressure (mm Hg)	96		87	5		The criterion of Diastolic Pressure > 95 is never used and therefore does not need to be defined on these rows (it is strictly a subcriterion of the compound criterion shown on the next row).
1001	Systolic Pressure >160 and Diastolic Pressure > 95		Y				This is a compound criterion that uses the values of variables that are present on rows with different values of PARAM.

For compound criterion rows, AVALC must always be populated with Y/N. If an analysis also requires a numeric indicator variable, either of the following two options may be chosen:

1) CRIT may be set to the same criterion text as PARAM, CRITFL set to the same Y/N value as AVALC, and CRITFN set to 1/0 .

2) AVAL may be set to a numeric 1/0 indicator value.

If an analysis requires only simple subsetting of the "hits" on a particular compound criterion, it is acceptable to add only the "compound criterion met" (AVALC='Y') rows to the dataset.

Note that if a compound criterion is defined, then the subcriteria do not have to exist on their own unless these subcriteria are themselves used for subsetting, display, or modeling purposes.

Compound criteria may be included in the same dataset with non-compound criteria (Example 4).

Example 4: Single-parameter (non-compound) criteria in the same dataset with a compound criterion.

Table 4.6.1.4

USUBJID	PARAM	AVAL	AVALC	BASE	CHG	CRIT	CRITFL	CRITN	CRIT1	CRIT1FL	CRIT1FN	CRIT2	CRIT2FL	CRIT2FN
Single-parameter criteria:														
1001	Systolic Blood Pressure (mm Hg)	163		148	15	Systolic Pressure >160	Y	1	Change from Baseline in Systolic Pressure>10	Y	1	Systolic Pressure >160 and Change from Baseline in Systolic Pressure>10	Y	1
1001	Diastolic Blood Pressure (mm Hg)	96		87	5	Diastolic Pressure >95	Y	1						
Compound criterion:														
1001	Systolic Pressure >160 and Diastolic Pressure > 95		Y											

Note that criterion “Diastolic Pressure >95” can coexist in the same CRIT column with “Systolic Pressure >160”. Each of these criteria is specific to its own subset of PARAM rows.

4.7 OTHER ISSUES TO CONSIDER

The issues presented in the previous sections represent analysis decisions that commonly occur when creating analysis datasets. However, the ADaM team recognizes that this is not an exhaustive list. This section provides comment on other issues that may arise.

4.7.1 Adding Records To Create a Full Complement of Analysis Timepoints For Every Subject

It is not unusual for a given subject to have missing data for a specified analysis timepoint. For example, suppose an analysis is to be performed for the data obtained at each of 4 visits and that no imputation is to be performed. For subjects who did not attend all 4 visits, it would be possible to create records in the analysis dataset for these missed assessments, with AVAL and AVALC missing (null). There are some advantages of having an analysis dataset contain the same number of observations for each subject. For example, programming is facilitated by having the same data dimensions for all subjects, and by explicitly representing missing data rather than implicitly representing it by the absence of a record. For some categorical analyses, the denominators can be obtained directly from the analysis dataset rather than from another input such as ADSL. The disadvantage of this approach is that it may require additional metadata to explain the use of these derived blank records and would require in some cases that subsetting statements be used to exclude the rows on which AVAL is missing. The ADaM team neither advocates nor discourages this practice. Readers are encouraged to consider if and how the SDTM trial design datasets could be used in conjunction with an analysis dataset to determine if a subject attended or missed a planned visit. Note that trial design datasets are only for planned visits, which may not be the same as analyzed timepoints.

4.7.2 Creating Multiple Datasets to Support Analysis of the Same Type of Data

The statistical analysis plan often specifies that an analysis will be performed using slightly different methodologies. For example, the primary efficacy analysis may be performed using two different imputation algorithms for missing values. The sponsor must decide whether to include both sets of the imputed observations in one analysis dataset or create two analysis datasets, each representing just one of imputation algorithms. The ADaM model provides variables that can be used to identify records that are used for different purposes. However, this does not imply that the sponsor should not or cannot submit multiple analysis datasets of similar content, each designed for a specific analysis.

4.7.3 Using SDTM with Additional Columns for Analysis

The purpose of analysis datasets is to provide a researcher or reviewer with the ability to recreate submitted analyses and/or explore alternate analyses. If an analysis is simple and is based largely on the observed data, then it is quite possible that appending column variables, such as treatment, population flags, analyzed flag, to a native SDTM domain may be sufficient to support simple analyses. In this case, one could argue that there is no reason to rename existing SDTM variables to the corresponding ADaM variable, such as renaming --TEST to PARAM, in order to declare this an ADaM-compliant dataset. While it is true that an 'SDTM+' dataset may be submitted as an analysis dataset, this would not be considered an ADaM-compliant dataset. As with any standard, it is important to adhere to the definable aspects of the standard in order to be compliant. This does not mean that SDTM+ dataset should not or cannot be submitted along with other ADaM-compliant datasets but sponsors should consider alerting the reviewer to this fact in the analysis dataset metadata. Ideally, as software matures, one would anticipate that a reviewer could create their own SDTM+ datasets on demand using inputs from ADSL and native SDTM domains.

APPENDIX A: REPRESENTATIONS AND WARRANTIES; LIMITATIONS OF LIABILITY, AND DISCLAIMERS

CDISC Patent Disclaimers.

It is possible that implementation of and compliance with this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any claim or of any patent rights in connection therewith. CDISC, including the CDISC Board of Directors, shall not be responsible for identifying patent claims for which a license may be required in order to implement this standard or for conducting inquiries into the legal validity or scope of those patents or patent claims that are brought to its attention.

Representations and Warranties.

Each Participant in the development of this standard shall be deemed to represent, warrant, and covenant, at the time of a Contribution by such Participant (or by its Representative), that to the best of its knowledge and ability: (a) it holds or has the right to grant all relevant licenses to any of its Contributions in all jurisdictions or territories in which it holds relevant intellectual property rights; (b) there are no limits to the Participant's ability to make the grants, acknowledgments, and agreements herein; and (c) the Contribution does not subject any Contribution, Draft Standard, Final Standard, or implementations thereof, in whole or in part, to licensing obligations with additional restrictions or requirements inconsistent with those set forth in this Policy, or that would require any such Contribution, Final Standard, or implementation, in whole or in part, to be either: (i) disclosed or distributed in source code form; (ii) licensed for the purpose of making derivative works (other than as set forth in Section 4.2 of the CDISC Intellectual Property Policy ("the Policy")); or (iii) distributed at no charge, except as set forth in Sections 3, 5.1, and 4.2 of the Policy. If a Participant has knowledge that a Contribution made by any Participant or any other party may subject any Contribution, Draft Standard, Final Standard, or implementation, in whole or in part, to one or more of the licensing obligations listed in Section 9.3, such Participant shall give prompt notice of the same to the CDISC President who shall promptly notify all Participants.

No Other Warranties/Disclaimers. ALL PARTICIPANTS ACKNOWLEDGE THAT, EXCEPT AS PROVIDED UNDER SECTION 9.3 OF THE CDISC INTELLECTUAL PROPERTY POLICY, ALL DRAFT STANDARDS AND FINAL STANDARDS, AND ALL CONTRIBUTIONS TO FINAL STANDARDS AND DRAFT STANDARDS, ARE PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, AND THE PARTICIPANTS, REPRESENTATIVES, THE CDISC PRESIDENT, THE CDISC BOARD OF DIRECTORS, AND CDISC EXPRESSLY DISCLAIM ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR OR INTENDED PURPOSE, OR ANY OTHER WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, FINAL STANDARDS OR DRAFT STANDARDS, OR CONTRIBUTION.

Limitation of Liability.

IN NO EVENT WILL CDISC OR ANY OF ITS CONSTITUENT PARTS (INCLUDING, BUT NOT LIMITED TO, THE CDISC BOARD OF DIRECTORS, THE CDISC PRESIDENT, CDISC STAFF, AND CDISC MEMBERS) BE LIABLE TO ANY OTHER PERSON OR ENTITY FOR ANY LOSS OF PROFITS, LOSS OF USE, DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THIS POLICY OR ANY RELATED AGREEMENT, WHETHER OR NOT SUCH PARTY HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Note: The CDISC Intellectual Property Policy can be found at
http://www.cdisc.org/about/bylaws_pdfs/CDISCIPPolicy-FINAL.pdf.