

Implementation Considerations for PARAM/PARAMCD Using ADaM BDS

Karl Miller, inVentiv Health Clinical, Lincoln, Nebraska
J.J. Hantsch, inVentiv Health Clinical, Chicago, Illinois

ABSTRACT

One of the key variables to the Analysis Data Model (ADaM) Basic Data Structure (BDS) data set is the PARAM variable which describes the analysis parameter and is directly useable in Clinical Study Report (CSR) displays and/or analysis table headings. Current guidance consists of stuffing all essential information, including the full test name, specimen type, unit of measurement, etc. into PARAM and generating a system to squeeze this information into the eight characters of PARAMCD. Such PARAM variables frequently become too unwieldy for column headings in the CSR and analysis tables, and require an arcane process to generate a uniquely matched PARAMCD. Consequently, when too much data is stuffed into only eight characters an incomprehensible value is the result.

Further, data set inquiries are hampered by an overly complex PARAM variable value. Along with impacting the PARAMCD variable, such overstuffed PARAM variables carry potential issues into the programming of the analysis itself. We examine the current implementation as well as a functional alternative, non-standard ADaM implementation for generation of the PARAM/PARAMCD variables. Our alternative method reduces PARAM to the full test name and the specified units, uses the PARCATy variables for any qualifying information, and keeps PARAMCD as a simple abbreviation of the test name reserving the eighth or last character for encoding which of the alternative units is reported. Although our method does not conform to the current ADaM Guidance, it maximizes the functionality and transparency of the programming and analysis.

INTRODUCTION

This paper will discuss the implementation of ADaM BDS structure, focusing on the use of the variables PARAM and PARAMCD. In some situations, following the strict interpretation of the BDS structure can lead to a gap between the data set and the functional use for analysis. Our alternative method, although non-compliant, can make the implementation easier and more functional in its use for programming and analysis.

The authors of the paper provide the pros and cons of using the alternative method and when to consider implementation. It is assumed that the reader of this paper is familiar with the Clinical Data Interchange Standards Consortium (CDISC) standard models: Study Data Tabulation Model (SDTM) and ADaM, which are available at www.cdisc.org.

CONSIDERATION FOR PARAM UTILIZATION IN BDS

ADaM, just like SDTM, attempts to keep variable naming consistent across different data sets; a variable with the same function in multiple data sets will be consistently named. Designated to describe the analysis parameter and provide information about the value being analyzed, or the values derivation, the variable PARAM is key in the ADaM BDS data standard. However, unlike variables in SDTM, the PARAM variable must be designed with two additional purposes in mind, (1) ease of programming usage and (2) applicability to output usage.

Because the variety of analyses and required subsets of gathered data cannot be fully anticipated, PARAM must be defined in a flexible manner to allow future programmers to subset the data sets efficiently. Also, because table output is an immediate function of ADaM data sets, the variable's content should be suitable for either column headings and/or the restricted confines of output tables. All of which are to be considered in creating the "analysis ready" data sets that will allow a statistical analysis with minimal additional processing, either through sorting and/or selection of the appropriate records from the analysis data set.

In many of the ADaM BDS data sets, which could be considered direct descendants of the SDTM findings domain, the 'topic' or controlling variables are the parameter name (PARAM) as well as the eight character short form of the parameter name, PARAMCD. This variable coordinates or 'slices' the data set into a set of records where each variable has a non-varying function. Thus in ADVS, the ADaM descendant of SDTM.VS, when PARAMCD equals 'SYSBP' the PARAM variable should be 'Systolic Blood Pressure (mmHg)' and the AVAL variable will contain a reported systolic blood pressure value.

Example 1:

SDTM.VSTESTCD	SDTM.VSTEST	SDTM.VSSTRESU	SDTM.VSPOS
SYSBP	Systolic Blood Pressure	mmHg	Sitting
SYSBP	Systolic Blood Pressure	mmHg	Supine
DIABP	Diastolic Blood Pressure	mmHg	Sitting
DIABP	Diastolic Blood Pressure	mmHg	Standing

Table 1. Vital Signs (VS): Variables that could be used in the analysis of Blood Pressure.

ADVS.PARAMCD	ADVS.PARAM
SYSBPSIT	Systolic Blood Pressure, Sitting (mmHg)
SYSBPSUP	Systolic Blood Pressure, Supine (mmHg)
DIABPSIT	Diastolic Blood Pressure, Sitting (mmHg)
DIABPSTD	Diastolic Blood Pressure, Standing (mmHg)

Table 2. Vital Signs (ADVS): Parameter information used in analysis, including position (i.e. VSPOS).

This illustrates the two issues mentioned above, sub-setting and use for output. The PARAM variable must be kept adequately clear to allow a subset of ‘... all systolic blood pressures administered while standing only...’ if the analysis includes the measurement position (i.e. Standing, Sitting, etc.) or to function as a column heading in an output table. These competing priorities require a balanced approach.

STRICT INTERPRETATION OF PARAM/PARAMCD

The current ADaM Implementation Guide (v2.1) stipulates that the qualifying information which in SDTM was gathered in ancillary variables (unit of measurement, position or location reference, testing methodology and testing specimen, all if applicable) be gathered exclusively in the PARAM variable. The controlling PARAMCD variable which changes value in synchronization with the PARAM variable then must have a wide variety of unique codes to match this complexity. Due to SAS V5 variable naming conventions and in order that the data set records might be transposed and appropriate variable names provided, the PARAMCD variable is restricted to only eight characters long. Thus the current guidance mandates a certain level of complexity for the PARAMCD variable.

Although for our example Systolic Blood Pressure (SYSBP) is unlikely to be confused. It is reported the world over in “millimeters of mercury” or as “mmHg”, and adding three letters should suffice to identify the subject’s measurement position (SUP for supine, STD for standing, SIT for sitting, etc.) as indicated in Table 2. However, when for many laboratory tests, additional information is necessary for appropriate analysis, such as the subject’s fasting status, then the assignment of a test code becomes an issue. Laboratory results reported have the tendency to encounter test codes (LBTESTCD) that are at or near eight characters alone; therefore it becomes an issue to add the additional space for the test unit, specimen type, or any other information needed for analysis. Many finding results with multiple modalities, samples or reporting units may require more than the eight character limit.

Working with the ADaM BDS, if a test code (--TESTCD) from the SDTM data is at least six characters long, either an alpha-numeric coding or a truly shoe-horn abbreviation system must be applied to squeeze any additional information into the last two (or less) characters of PARAMCD. Other approaches are possible, but keeping the PARAMCD meaningful should be a top priority; it is inefficient when a code is either inconsistent or must be looked up every time.

Example 2:

LBTESTCD	LBTEST	LBSTRESU	LBSPEC	LBFAST
GLUC	Glucose	mmol/L	BLOOD	Y
HGB	Hemoglobin	g/dL	BLOOD	Y
BILI	Total Bilirubin	mg/L	SERUM	Y
GLUC	Glucose	mg/dL	URINE	
HGB	Hemoglobin	g/L	BLOOD	
BILI	Total Bilirubin	µmol/L	URINE	

Table 3. SDTM Laboratory Test Results (LB): Potential information used for analysis to be stored in ADaM.

PARAMCD	PARAM
GLUCBDFS	Glucose, Blood, Fasting, (mmol/L)
HGBBDFC	Hemoglobin, Blood, Fasting, (g/dL)
BILSRFC	Total Bilirubin, Serum, Fasting, (mg/L)
GLUCURC	Glucose, Urine, (mg/dL)
HGBBDS	Hemoglobin, Blood, (g/L)
BILIURS	Total Bilirubin, Urine, (µmol/L)

Table 4. ADaM Laboratory Analysis (ADLB): Strict adherence to storing applicable information would require meaningful abbreviation for the unit (LBSTRESU), specimen (LBSPEC) and fasting status (LBFAST) to be placed in PARAMCD.

The current ADaM Implementation Guide requires all of this information for analysis to be included in the PARAM variable, which can be up to two hundred characters in length. Then, a one-to-one match to the PARAMCD variable is also required which can be difficult to see the traceability as PARAMCD can be significantly different. With the inclusion of all the necessary information used in the analysis, it is easy for PARAM to become unwieldy for its direct use in column headers for the table output. Analysis tables, such as laboratory summary tables, are of limited space to keep the number of necessary table outputs relatively small. Thus, it creates a need to keep PARAM suitable for presentation.

Example 3:

Treatment	Time Point	Statistic	Glucose, Blood, Fasting, (mmol/L)
Treatment XYZ	Day X	N	XX
		SD	XX.XXX
		Median	XX.XX
		Minimum	XX.X
		Maximum	XX.X

Output 1. Output displaying the direct value of the PARAM variable used as a column in a standard analysis table. Using the full value of PARAM can create long column headers that decrease the number of tests presented per page as well as can vastly increase the size and pages of the output file.

Test Name (units)	Specimen Type	Fasting Status	Statistic	Treatment XYZ
Glucose (mmol/L)	Blood	Fasting	N	XX
			SD	XX.XXX
			Median	XX.XX
			Minimum	XX.X
			Maximum	XX.X

Output 2. Output displaying the value of the PARAM variable used in an analysis table. Sub-strings of the value in PARAM are needed to create the output according to the approved table shell.

Often the case in using the strict interpretation of PARAM, the variable will need to be manipulated, sub-stringed into separate variables to match the approved output shell which requires additional processing steps for the final output. Neither a simple substring nor scan function would generate predictable results given the inconsistencies in PARAM as shown above in Example 2, Table 4. All of which are time consuming for the programmers and the alternative to employ a searching function on a much longer PARAM variable that could return results of invalid records for analysis.

FUNCTIONAL INTERPRETATION OF PARAM/PARAMCD

In order to maintain transparency through the data set creation process and allow users to work more efficiently with less preparation time, the alternative we propose requires the inclusion of only the functional information into the PARAM variable. That is, for the vast majority the reporting values found in PARAM would simply be the test name (-TEST) and standard reporting unit (--STRESU) from the corresponding SDTM data. We feel that this maintains user traceability from the SDTM data to the ADaM analysis data set and subsequently into the analysis table, figure or listing (TFL) output. Thus, our ADVS example using SYSBP would be “Systolic Blood Pressure (mmHg)” suitable for a table column heading or as a descriptor for an analysis variable row in a table. However, there is a catch.

This method is not in compliance with the ADaM (v2.1) BDS standard. Currently, any implementation of ADaM BDS requires that any value in PARAM “should be sufficient to describe unambiguously the contents of AVAL and/or AVALC.” Simply placing the test name and unit into the PARAM value would be able to meet the criteria for some analyses, with the majority of cases being out of compliance for any analysis that requires the use of qualifying information or criteria.

Although our method is non-compliant, the unambiguous relationship for the values of AVAL is kept through the use of PARAM along with a set of variables, PARCATy. The PARCATy variables must be called into service to carry the qualifying information about specimen type, measurement position or any additional information deemed reportable to maintain the unambiguous relationship to AVAL and/or AVALC for analysis. As long as the numbering scheme for the PARCATy variables is standardized beforehand, then the values are perfectly predictable and relevant. Besides freeing the PARAM variable to be a suitable column header, employing the PARCATy variable also provides for easy access to this information for sub-setting in the future. If there is a request to only report on standing or sitting blood pressure and heart rates then it is comparatively simple to employ a sub-set on PARCATy for measurement position.

Example 4:

<PARAM>	<PARCAT2>	<PARCAT3>		
Test Name (units)	Specimen Type	Fasting Status	Statistic	Treatment XYZ
Glucose (mg/dL)	Blood	Fasting	N	XX
			SD	XX.XXX
			Median	XX.XX
			Minimum	XX.X
			Maximum	XX.X

Output 3. Output displaying the value of the PARAM variable used in analysis with the use of PARCATy variables to store the specimen type and fasting status.

Our method also simplifies the PARAMCD variable creation. The default PARAMCD values would be the test code (--TESTCD) from SDTM with the expectation that the standard unit of the test is consistent across the study compound, or even the company. Any additional units that might be collected, that are different from the established standard unit (rare occasion) would result to a meaningful system to develop and implement. Using this method still maintains the one-to-one relationship between PARAM and PARAMCD, while also providing clear traceability from SDTM to ADaM as PARAMCD would be more closely tied to the values from --TESTCD.

CONCLUSION

We have presented two possibilities for consideration: 1) strict adherence to the current ADaM IG and 2) storing useful qualifying information in the PARCATy variables. Every implementation must make tradeoffs between clarity, practicality and usefulness. The current standard does not always include the clarity of the PARAMCD variable and direct usefulness for the PARAM variable in programming for analysis outputs. Our alternative implementation employs the existing PARCATy variables to maintain a meaningful PARAMCD plus retains a useful PARAM variable and leads to a straight-forward approach in the programming of the analysis data sets through direct relationship between the outputs and the analysis data. This method also minimizes, if not eliminates, any data manipulation of PARAM in the programming of the analysis tables reducing the potential for errors. We believe our method protects clarity, and enhances practicality and usefulness of the ADaM data based on the SDTM domain.

REFERENCES

Clinical Data Interchange Standards Consortium (CDISC):

Study Data Tabulation Model (SDTM) and Implementation Guide, v.1.2, IG v.3.1.2, Available at www.cdisc.org/sdtm.

Analysis Data Model (ADaM) and Implementation Guide, v.2.1, IG v.1.0, Available at www.cdisc.org/adam.

CONTACT INFORMATION

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

Karl Miller
inVentiv Health Clinical
(402) 643-3962
karl.miller@inventivhealth.com

J.J. Hantsch
inVentiv Health Clinical
1060 W. Addison
Chicago, IL 60622
(630) 422-7375
joseph.hantsch@inventivhealth.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.