PharmaSUG 2026 - Paper PN-245

ADaM Pet Peeves Part 2: More Things Programmers Do That Make Us Crazy

Nancy Brucken and Paul Slagle; IQVIA;
Nate Freimark, The Griesser Group;
Sandra Minjoe, ICON PLC;
Tatiana Sotingco, Johnson & Johnson;
Richann Watson, DataRich Consulting;
Alyssa Wittle, Atorus Research Inc.

ABSTRACT

The authors have been actively involved in the Clinical Data Interchange Standards Consortium (CDISC) Analysis Data Model (ADaM) team for many years, and they include past and future CDISC ADaM Team Leads, CDISC ADaM Sub-team leads, and authorized CDISC ADaM trainers. Because of our extensive ADaM expertise, we each end up reviewing a lot of ADaM submissions before they are sent to regulatory agencies. Following the companion paper presented at PharmaSUG in 2025, this paper highlights additional common issues that we've seen ADaM developers make. For each topic, we explain the issue and provide a better and/or more conformant approach.

INTRODUCTION

This paper includes multiple topics, and addresses situations where we regularly see content that is either incorrect or just not optimal:

- Keeping the end in mind
- Design of exposure analysis dataset(s)
- Misuse of category and criterion variables
- When to use traceability variables
- SDTM data in ADaM
- When to use Class ADAM OTHER
- Use of senior reviewers
- ADRG Content: Describing Data Issues and Enhancing Review

For each of these issues, we not only explain the issue, but also provide a better and/or more conformant approach.

Note: The examples shown in this paper present some of the key variables and records that would be included in the dataset and are not intended to contain every possible variable.

KEEPING THE END IN MIND

In clinical data standards, ADaM datasets serve as the bridge between raw data collection and final statistical reporting. To ensure a seamless submission, ADaM design must not be treated as a vacuum-sealed step. This paper explores a bidirectional approach: ensuring upstream inputs are scrutinized through an analytical lens and designing ADaM structures that anticipate the specific needs of downstream deliverables. Considering these approaches will streamline the study and eliminate rework when issues are discovered.

UPSTREAM INPUTS: REVIEWING WITH AN ANALYSIS PERSPECTIVE

ADaM excellence begins long before the first line of any programming code is written. The following components must be reviewed with the final analysis goals in mind to prevent "data debt" during the programming phase.

- **Protocol & Statistical Analysis Plan (SAP):** The Protocol and SAP are the blueprints. Reviewing these early ensures that planned endpoints are actually supported by the data being collected.
- **Case Report Forms (CRFs) & Completion Guidelines:** CRFs must capture the raw data necessary for complex ADaM derivations and not collect data that would be instead programmatically determined in ADaM. An example of data that would be programmatically determined in ADaM is whether a concomitant medication is a medication of special interest. If completion guidelines are vague, data entry variability can lead to inconsistent analysis variables.
- **Data Transfer Agreements (DTAs):** For external data (Labs, Biomarkers), DTAs must contain the data needed for analysis and ensure that vendor data aligns with SDTM requirements to facilitate smooth ADaM mapping.
- **SDTM Datasets:** SDTM datasets should be reviewed for "ADaM readiness." Unnecessary hurdles for ADaM traceability can be caused by missing SDTM content, inconsistent mapping in SDTM, and, where applicable, not using CDISC-defined controlled terminology.

DOWNSTREAM DELIVERABLES: DESIGNING FOR THE CONSUMER

A successful ADaM designer must look forward to how the datasets will be used by regulators, programmers, and medical writers. By adopting a "start with the end in mind" philosophy, ADaM designers transform from passive data mappers into strategic architects. This approach not only ensures regulatory compliance but significantly reduces the time from database lock to final clinical study report.

Submission Documents & Metadata

- define.xml Integration: Are your ADaM specifications detailed enough to auto-populate the define.xml? Mapping metadata directly from specs reduces manual error and ensures compliance.
- Analysis Data Reviewer’s Guide (ADRG): Documentation should happen in real-time. By recording complex derivation logic or data issues during the programming phase, the ADRG becomes a robust narrative of the study’s data integrity.

Structural Alignment with the Table, Figure, and Listing (TFL) Table of Contents (TOC):

- Cross-check the TOC early. Ensure every table and figure has a clear data source within the ADaM library. Listings can be sourced from an ADaM dataset, if it exists, but there is no mandate to create an ADaM dataset solely to support a listing. For example, in many studies medical history is only reported in a listing, so there would be no need to create an ADaM dataset for that content.

TFL Generation

- Does the ADaM design facilitate direct reporting, or will the TFL programmer need to perform extensive data manipulation? Incorporating analysis-ready flags, summary parameters, and pre-calculated categories (beyond AVAL) is essential. ADaM should pre-calculate the endpoints used, the record selection (such as with flags), and the pooling required for analysis. These variables should not be derived in table or figure programs.

Compound-Level Consistency

- When working on a single compound, ADaM design should be standardized across studies. This foresight facilitates a smoother transition to Integrated Summaries of Safety and Efficacy (ISS/ISE) by ensuring ADaM structures, variables and definitions are pre-harmonized. It also helps regulatory reviewers as they move between studies within the same submission.

Design of Exposure Analysis Datasets

The pressure to deliver "analysis-ready" datasets often leads programming teams to map SDTM EX and EC domains directly into a summarized Basic Data Structure (BDS) format. While this shortcut appears efficient, it often compromises the CDISC traceability chain by collapsing "as-collected" dose adjustments (EC) and "as-treated" administration records (EX) without a record-level audit trail.

For simple, single-dose studies, sourcing summary variables directly from the Study-Level Analysis Dataset (ADSL) is often sufficient. For more complex dosing schedules, to maintain regulatory rigor while ensuring programming efficiency, developers should consider adopting a dual-dataset approach as described in the ADaM Oncology Examples v1.0: using ADEXP to harmonize occurrence-level exposure and ADEXSUM to deliver summarized parameters.

THE ADEX CONNECTOR

In this model, ADEXP serves as the vital connector between SDTM and ADaM. By using a structure other than BDS, ADEXP combines EX and EC while maintaining high-resolution traceability through paired variables (e.g., EXSEQ/ECSEQ and EXDOSE/ECDOSE). For example, one record in ADEXP can represent the administration of (study) drug at an unchanged dose level during the time window defined by ASTDT and AENDT. Performing complex, record-level derivations, such as dose intensity, at this stage simplifies the subsequent summarization. Furthermore, creating numeric ADaM dates here ensures that calculations are direct and verifiable. Content from the Drug Accountability (DA) dataset might also be pulled in at this step.

THE ADEXSUM SUMMARIZER (BDS)

Dataset ADEXSUM is designed for use in producing exposure summary results, such as duration of therapy and total dose. It can be created by summarizing content across multiple ADEXP records into BDS parameters. This structure provides another option for integrating drug accountability data.

WORKING WITH THE STANDARD, NOT AGAINST IT

The ADaM IG does not explicitly mandate this two-step process, but the spirit of the standard favors it in complex dosing schedules. Sophisticated ADaM development requires knowing when a single structure is being "forced." Trying to compel an event-level non-BDS dataset to behave like a BDS—or forcing interventions into BDS parameters —adds unnecessary complexity. If an ADaM dataset feels forced, it is usually a sign to take a step back and re-evaluate.

WHEN LESS IS MORE

As previously stated, a dual-dataset approach is not always necessary. For simple, single-dose studies, sourcing summary variables directly from ADSL is often sufficient. However, if you find yourself comparing intermediate datasets during QC to resolve discrepancies, that is a clear signal that the intermediate data deserves its own ADaM dataset. Splitting the data at its logical dividing point—the transition from event-level to summary—is the best way to ensure quality, transparency, and ease of review. This separation results in significantly cleaner, more modular programming.

Misuse of Category and Criterion Variables

The ADaMIG pre-defines variables used for categorizing values (AVALCAT***y***, BASECAT***y***, CHGCAT***y***, and PCHGCAT***y***), for flagging records that meet specific criteria (CRIT***y*** / CRIT***y***FL and MCRIT***y*** / MCRIT***y***ML), and for identifying records that are needed for analyses (ANL***zz***FL). Table 1 provides a brief description of each of these variables.

CATEGORY/ CRITERIA VARIABLE	DESCRIPTION	BASED ON:
AVALCAT***y***	Categorizes the analysis value into mutually exclusive groups	AVAL
BASECAT***y***	Categorizes the baseline value into mutually exclusive groups	BASE
(P)CHGCAT***y***	Categorizes the analysis value (percent) change from baseline into mutually exclusive groups	(P)CHG
(M)CRIT***y***	Flags a record indicating whether it met a specific criterion. Criterion can be binary (CRIT***y***/CRIT***y***FL) or can have multiple response (MCRIT***y***/MCRIT***y***ML).	Any variable on the **same** row
ANL***zz***FL	Flag used to select a record for analysis when the existing variables are not sufficient for selecting the record based on specific criteria.	Any variable and/or **any** row

Table 1: Category and Criterion Variables Description

The proper use of these variables tends to lead to confusion. Below are recommendations of when to use each type of variable.

- If the categories are mutually exclusive and are based solely on AVAL, BASE, CHG or PCHG, then use the corresponding CAT***y*** variable (AVALCAT***y***, BASECAT***y***, CHGCAT***y***, PCHGCAT***y***). In other words, if your categorization relies on another variable such as ANRLO or ANRHI, then the CAT***y*** variables are not the correct option. The categorization is only based on the variable that corresponds to the root of the CAT***y*** variable.
- CRIT***y*** should be used when assessing a binary criterion for a parameter if the value is derived based on variables other than AVAL, BASE, CHG, or PCHG alone, thereby precluding the use of CAT***y***. CRIT***y*** permits evaluation of criteria using any variable present on the current record; however, it does not support deriving criteria based on values from other records.
- MCRIT***y*** is similar to CRITy, with the distinction that MCRITy supports multi-level (non-binary) mutually exclusive response categories.
- If assessing criteria based on other rows to select a specific record because the existing variables are not adequate to uniquely select a record for analysis, then use ANL***zz***FL**.**
- For other scenarios it would probably be best to create a new parameter that contains the criteria and use AVAL/AVALC to capture the result and in some cases creating a new dataset could be beneficial. This is not the focus of this paper, but you can read more at “Proper Parenting: A Guide in Using ADaM Flag/Criterion Variables and When to Create a Child Dataset” (Watson, Miller, & Slagle, 2015).

The following examples help to demonstrate when to use AVALCAT***y***, CRIT***y***(FL), MCRIT***y***(ML), or ANL***zz***FL.

Example Using AVALCATy

For illustration purposes, let’s assume that the parameter AST is to be categorized into 3 mutually exclusive categories: < 75, 75 - < 150, and >= 150. Because these are mutually exclusive groups and because the categories are solely based on the value of AVAL, then AVALCAT***y*** is the proper variable to use in this case as seen in Sample Data 1.

Row	USUBJID	PARAMCD	AVISIT	ADT	AVAL	AVALCAT1
1	ABC-001-001	AST	SCREENING	2025-11-24	32	< 75
2	ABC-001-001	AST	BASELINE	2025-12-02	36	< 75
3	ABC-001-001	AST	WEEK 2	2025-12-17	58	< 75
4	ABC-001-001	AST	WEEK 4	2025-12-29	75	75 - < 150
5	ABC-001-001	AST	WEEK 6	2026-01-14	97	75 - < 150
6	ABC-001-001	AST	WEEK 8	2026-01-28	157	>= 150
7	ABC-001-001	AST	WEEK 8	2026-01-29	134	75 - < 150

*Sample Data 1: Illustration of the use of AVALCAT**y** For Groupings That are Mutually Exclusive*

Examples Using CRITy

But what if the need is to determine the number of records are >= 75, as well as the number of records that are >= 150. Although the groupings are based only on AVAL, these are not mutually exclusive groupings because a value of 157 can be both >= 75 and >= 150. Since these are not mutually exclusive, a suggested solution would be to use multiple CRIT***y***(FL) variables (Sample Data 2).

Row	USUBJID	PARAMCD	AVISIT	ADT	AVAL	CRIT1	CRIT1FL	CRIT2	CRIT2FL
1	ABC-001-001	AST	SCREENING	2025-11-24	32				
2	ABC-001-001	AST	BASELINE	2025-12-02	36				
3	ABC-001-001	AST	WEEK 2	2025-12-17	58				
4	ABC-001-001	AST	WEEK 4	2025-12-29	75	>= 75	Y		
5	ABC-001-001	AST	WEEK 6	2026-01-14	97	>= 75	Y		
6	ABC-001-001	AST	WEEK 8	2026-01-28	157	>= 75	Y	>= 150	Y
7	ABC-001-001	AST	WEEK 8	2026-01-29	134	>= 75	Y		

*Sample Data 2: Illustration of the Use of CRIT**y**(FL) for Groupings That are Not Mutually Exclusive*

Another illustration of the use of CRIT***y***(FL) is when the criteria are based on at least one other variable in the same row. For example, Sample Data 3 shows that the criterion is based on AVAL and ANRHI. In this situation CRIT***y***(FL) would be the ideal variables to use.

Row	USUBJID	PARAMCD	AVISIT	ADT	AVAL	ANRHI	CRIT3	CRIT3FL
1	ABC-001-001	AST	SCREENING	2025-11-24	32	48		
2	ABC-001-001	AST	BASELINE	2025-12-02	36	48		
3	ABC-001-001	AST	WEEK 2	2025-12-17	58	48		
4	ABC-001-001	AST	WEEK 4	2025-12-29	75	48		
5	ABC-001-001	AST	WEEK 6	2026-01-14	97	48		
6	ABC-001-001	AST	WEEK 8	2026-01-28	157	48	>=3xULN	Y
7	ABC-001-001	AST	WEEK 8	2026-01-29	134	48		

*Sample Data 3: Illustration of the Use of CRIT**y**(FL) When Criteria is Based on Other Variables*

Example using MCRITy

What if the need is to determine records that are above upper limit of normal (ULN) but at varying degrees, such as ULN - < 1.5xULN, 1.5 - <3xULN, 3 - <5xULN, and >=5xULN. Although the criteria are mutually exclusive, they are based on using AVAL as well as ANRHI. Therefore, MCRIT***y***(ML) should be used as shown in Sample Data 4.

Row	USUBJID	PARAMCD	AVISIT	ADT	AVAL	ANRHI	MCRIT1	MCRIT1ML
1	ABC-001-001	AST	SCREENING	2025-11-24	32	48	Above ULN	
2	ABC-001-001	AST	BASELINE	2025-12-02	36	48	Above ULN	
3	ABC-001-001	AST	WEEK 2	2025-12-17	58	48	Above ULN	ULN - < 1.5xULN
4	ABC-001-001	AST	WEEK 4	2025-12-29	75	48	Above ULN	1.5 - <3xULN
5	ABC-001-001	AST	WEEK 6	2026-01-14	97	48	Above ULN	1.5 - <3xULN
6	ABC-001-001	AST	WEEK 8	2026-01-28	157	48	Above ULN	3 - <5xULN
7	ABC-001-001	AST	WEEK 8	2026-01-29	134	48	Above ULN	1.5 - <3xULN

*Sample Data 4: Illustration of the Use of MCRIT**y**(ML) When There are Multiple Criteria Levels*

*Example Using ANL**zz**FL*

In some cases, it is necessary to look across multiple rows to identify the best record to use for a specific analysis. If looking across multiple rows, then ANL***zz***FL should be used to identify the best record. In Sample Data 5, there are two records for Week 8 and only one should be used for analyses. In this example, it is decided that only the baseline record and post-baseline visits with the maximum value would be used; therefore, ANL01FL is set to 'Y' for rows 2 – 6 only.

Row	USUBJID	PARAMCD	AVISIT	ADT	AVAL	ABLFL	ANL01FL
1	ABC-001-001	AST	SCREENING	2025-11-24	32		
2	ABC-001-001	AST	BASELINE	2025-12-02	36	Y	Y
3	ABC-001-001	AST	WEEK 2	2025-12-17	58		Y
4	ABC-001-001	AST	WEEK 4	2025-12-29	75		Y
5	ABC-001-001	AST	WEEK 6	2026-01-14	97		Y
6	ABC-001-001	AST	WEEK 8	2026-01-28	157		Y
7	ABC-001-001	AST	WEEK 8	2026-01-29	134		

*Sample Data 5: Illustration of the Use of ANL**zz**FL to Select a Record*

To help navigate the use of these variables, refer to the navigation tree found in the APPENDIX – NAVIGATING CATEGORY AND CRITERIA VARIABLES.

WHEN TO USE TRACEABILITY VARIABLES

The ADaMIG defines two levels of traceability:

- Metadata traceability describes the relationship between an analysis variable and its source dataset(s) and variable(s), and it is required for ADaM compliance.
- Datapoint traceability goes a step further, pointing directly to the specific predecessor record(s), and should be implemented when possible.

Metadata traceability is provided via the ADaM specifications, and ultimately, by the define.xml file. Each analysis dataset created for a study is described by metadata, which includes information such as the ADaM class, dataset label, structure, and unique keys. Each variable in an analysis dataset is also described by metadata, including attributes such as the variable label, its origin and possible values, and details on how the variable was created.

Datapoint traceability is commonly provided through the inclusion of --SEQ variables from source SDTM datasets, or by populating the ADaM SRCDOM, SRCVAR and SRCSEQ variables to point back to the record and variable used in deriving an analysis record or variable. While these variables are permissible, and not required, their use is encouraged wherever feasible, as they provide a direct route back to the source of the analysis data. If metadata traceability is the equivalent of holding a road map in your hands, datapoint traceability is analogous to having a GPS providing turn-by-turn directions.

Example 1: Using --SEQ variables

Here's an example of using datapoint traceability in an analysis dataset of ECG results where all of the records come from an SDTM EG dataset. For this study, triplicate readings were taken at the Baseline visit, and the average of those measurements was used as the baseline value. Only a single reading was taken at subsequent visits. Since there is only one source dataset, EGSEQ from dataset EG in Sample Data 6 is all that is needed to provide datapoint traceability for dataset ADEG in Sample Data 7.

USUBJID	EGSEQ	EGTESTCD	EGSTRESN	VISIT
A100-01	1	PRAG	150	BASELINE
A100-01	2	PRAG	152	BASELINE
A100-01	3	PRAG	155	BASELINE
A100-01	4	PRAG	149	WEEK 4
A100-01	5	PRAG	153	WEEK 8

Sample Data 6: Selected variables from dataset EG

USUBJID	EGSEQ	PARAMCD	AVAL	BASE	ABLFL	AVISIT	DTYPE	ANL01FL
A100-01	1	PRAG	150	152.33		BASELINE		
A100-01	2	PRAG	152	152.33		BASELINE		
A100-01	3	PRAG	155	152.33		BASELINE		
A100-01		PRAG	152.33	152.33	Y	BASELINE	AVERAGE	Y
A100-01	4	PRAG	149	152.33		WEEK 4		Y
A100-01	5	PRAG	153	152.33		WEEK 8		Y

Sample Data 7: Selected variables from dataset ADEG

Note that in Sample Data 7 the record containing the derived average value at the Baseline visit does not have EGSEQ populated. Since its analysis value is derived from multiple records, no single EGSEQ value applies. In addition, DTYPE has been set to “AVERAGE” to indicate how the value of AVAL has been derived for that record.

Example 2: Using Source (SRC) Variables*

Here’s a slightly more complicated example showing how datapoint traceability can be implemented when an analysis dataset is built from multiple source datasets. For this study, time to first headache was calculated based on the start date of the first reported headache in the SDTM AE dataset. If a subject did

not experience any headaches, they were censored at their end of study date in ADSL. Since there are multiple source datasets, AE in Sample Data 8 and ADSL in Sample Data 9, the combination of SRCDOM/SRCVAR/SRCSEQ variables can be used to provide datapoint traceability within dataset ADTTE in Sample Data 10.

USUBJID	AESEQ	AEDECOD	AESTDTC	AESTDY
B100-02	1	HEADACHE	2025-12-02	25
B100-02	2	FEVER	2025-12-04	27
B100-02	3	HEADACHE	2025-12-17	40
B100-03	1	APPENDICITIS	2026-01-05	12

Sample Data 8: Selected variables from dataset AE

USUBJID	EOSDT	EOSDY
B100-02	30DEC2025	53
B100-03	25JAN2026	32

Sample Data 9: Selected variables from dataset ADSL

USUBJID	PARAMCD	ADT	AVAL	SRCDOM	SRCVAR	SRCSEQ	CNSR
B100-02	TTHEAD1	02DEC2025	25	AE	AESTDY	1	0
B100-03	TTHEAD1	25JAN2026	32	ADSL	EOSDY		1

Sample Data 10: Selected variables from dataset ADTTE

Here, the first subject had more than one headache, so AVAL is the value of AESTDY for the first occurrence (value of 25). This is from the AE row where AESEQ = 1.

The second subject did not have a headache, so AVAL is the value of EOSDY (value of 32). This is from the ADSL row for the subject. Since their analysis value comes from ADSL, which does not contain a sequence number, SRCSEQ is not populated.

Note that SRCVAR contains the name of the variable that was used as the source of the analysis value on the record.

SDTM Data in ADaM

Which variables from SDTM should be copied to ADaM? Which should not?

First, any variables used in analysis need to be copied to ADaM. In BDS, there aren't many of these, but there are in the Occurrence Data Structure (OCCDS). For example, many of the MedDRA hierarchy variables are used in AE tables, so those must be copied to ADaM. You can't perform the analysis without these variables.

Next, the FDA Study Data Technical Conformance Guide section 4.1.2.2 states "all SDTM variables utilized for variable derivations in ADaM should be included in the ADaM datasets when practical." For example, when LBTEST and LBSTRESU are used to derive PARAM, when LBSTRESN is used to derive AVAL, and when LBDTC is used to derive ADT, keep these SDTM variables.

Also keep any SDTM variables that provide traceability back to the SDTM dataset. Often this is as simple as keeping the --SEQ variable, such as LBSEQ. In more complex ADaM datasets, use SRCDOM, SRCSEQ, and SRCVAR.

Consider keeping any SDTM variables that would be useful for listings. While listings don't need to be analysis-ready, why create extra work if it isn't needed? (Though not all listings necessarily need an ADaM dataset, but that's a different topic!)

However, there is no need to keep every SDTM variable in the ADaM dataset. Variables that aren't being used for analysis, to derive other variables, for traceability, or in listings don't need to be part of the ADaM dataset. After all, the reason we have traceability variables in ADaM is so that you can trace back to the input dataset to find everything else. Don't clutter up the ADaM dataset with variables that aren't useful!

WHEN TO USE CLASS ADAM OTHER

There are standard ADaM structures for ADSL, BDS, and OCCDS, plus some medical device structures, which each have their own Define-XML Class. Anything that doesn't fit into one of these defined ADaM standard structures uses instead the Class of ADAM OTHER. ADAM OTHER datasets are still ADaM datasets, and must comply with all the ADaM Fundamental Principles and naming conventions, but they are just not in a standard structure.

Some examples of datasets in the Class of ADAM OTHER include:

Case 1: A dataset used for multivariate analysis, where multiple parameters need to be on the same row in order to derive the analysis output results in a table or figure. There is a nice example of this in the ADaM Examples of Traceability v1.0 document, where a BDS dataset is transposed into a wider dataset of Class ADAM OTHER, such that the needed BDS PARAMCDs become variable names in the ADAM OTHER structure, and the content from the BDS variable AVAL for those BDS parameters becomes the value within each of those ADAM OTHER variables. It's the need to meet the "Analysis Ready" fundamental principle that drives the use of ADAM OTHER, because the original BDS dataset, with parameters on different rows, was not analysis-ready.

Case 2: A dataset that will require complicated derivations involving multiple parameters, such as Hy's Law. In this case, we need to make use of a handful of specific BDS parameters (for Hy's Law that would include values for parameters of Bilirubin, AST, and ALT, each in reference to their normal ranges), to derive other factors from them. Here it is much easier to perform, document, and review derivations when these values are on the same row. So, like Case 1, we transpose the data from an input BDS dataset, with just the few pieces we need as variables, then we can use the transposed variables to derive the other variables needed for analysis, as shown in Table 2. In this case, the horizontal structure allowed by using Class of ADAM OTHER helps us better communicate clearly and unambiguously, meeting an ADaM fundamental principle.

VARIABLE	BASED ON
AST	AVAL from PARAMCD = 'AST' in ADLB
ASTULN	ANRHI from PARAMCD = 'AST' in ADLB
ASTCAT	Derived based on AST and ASTULN
ALT	AVAL from PARAMCD = 'ALT' in ADLB
ALTULN	ANRHI from PARAMCD = 'ALT' in ADLB
ALTCAT	Derived based on ALT and ALTULN

Table 2: Selected Transposed and Derived Variables for Case 2

Case 3: An intermediate dataset, used as a step along the way to the dataset that will be used for analysis. The purpose of an intermediate dataset is to split off some of the complexity and allow for easier review and QC, where the intermediate dataset is not used for any analysis itself. A common situation for

this is a dataset that collects all the possible dates that could be used for an event or censoring, on the way to a time-to-event ADaM dataset, and examples can be found in the ADaM Examples of Traceability v1.0 document and both the Prostate Cancer and Breast Cancer Therapeutic Area User Guides. Although BDS could "sort of" work as the structure for this intermediate dataset, what would the required variables like Analysis Parameter and Analysis Value mean when you're not actually using the dataset for analysis? Here, it's the lack of an analysis need that drives the use of ADAM OTHER. The ADaM Oncology Examples v1.0 document also includes an intermediate dataset ADEXP, used for doing some of the work along the way to creating BDS dataset ADEXSUM, the exposure summary dataset used for analysis. Intermediate datasets would need to have metadata like any other ADaM dataset, so, like Case 2, using one supports the ADaM fundamental principle of providing clear and unambiguous communication.

Case 4: A dataset created solely to make listing production more straightforward. For example, consider a listing of drug accountability, where you need to show pills Dispensed, Returned, and Taken on the same row of the output – this would be difficult to derive from a dataset structured like SDTM DA, where Dispensed Amount is captured on one row, Returned Amount on another row, and Amount Taken derived by subtracting. The ADaM Oncology Examples v1.0 document also includes an intermediate dataset ADEXP, generated using input datasets EX and EC, and used as not only an intermediate dataset but also for producing a listing. Although ADaM datasets are not required to be listing-ready, and you could do all this work (transposing data, subtracting values, interleaving EX and EC data) within the listing program, creating a dataset that puts all this content in a listing-ready structure might be easier to QC and facilitate data review. Like Cases 2 and 3, this also supports the ADaM fundamental principle of providing clear and unambiguous communication.

Cases 1 and 2 are ADAM OTHER datasets used directly in analysis, and they highlight how the ADaM fundamental principles, like being analysis-ready and providing clear and unambiguous communication, need to be part of the decision on which data structure to use. Since most of the time you could use an ADaM standard structure for an analysis, and you don't have highly complex derivations involving multiple parameters like needed in Hy's Law, there are usually very few datasets with the Class of ADAM OTHER that are used for analysis.

Cases 3 and 4 describe situations where the dataset is not used for analysis, meaning there is no "analysis-ready" requirement. While often the BDS structure may "sort of" work, some of the required BDS variables might not make sense when the dataset isn't used for analysis, so Class of ADAM OTHER is a reasonable choice.

USE OF SENIOR REVIEWERS

Senior reviewers are experts with deep knowledge of CDISC standards and extensive experience in clinical trials. This background allows them to know where potential challenges are likely to arise — whether in dataset structure, variable naming, derivation logic, or documentation.

One role of a senior reviewer is to provide an independent review of a study before it is used in a submission. This means approaching the work with fresh eyes, free from the assumptions and habits that can develop within a project team over time. A senior reviewer asks critical questions about the analyses performed, probing whether the methods chosen are appropriate, well-documented, and defensible to regulatory agencies.

A key part of this work is checking traceability from the Analysis datasets back to SDTM, ensuring that every derivation can be followed and justified. Just as important is the principle of sufficiency: a senior reviewer helps to assure that only what is needed is done, and not more than that. Over-engineering datasets or creating unnecessary complexity can introduce errors and slow down submissions, so keeping the work focused and proportionate is a core objective of the review.

The review process works best when senior reviewers are engaged early. Reviewing specifications before programming begins allows incorrect variable names to be caught and corrected, dataset structures to be reconsidered, and complicated derivations to be simplified or split into more manageable components. At this stage, changes are straightforward and low-risk. Once programming is underway, the same corrections become far more disruptive.

Senior reviewers also compare specifications to Tables, Listings, and Figures to identify what may be missing or misaligned. For more complex studies, they serve a mentoring role, guiding the team through nuanced ADaM decisions and helping to build the team's overall capability.

In our experience conducting senior reviews across multiple studies, several recurring themes have emerged. One of the most common is initial resistance from teams who are accustomed to doing things a particular way. The phrase "the way we have always done it" is a familiar refrain, and overcoming that inertia requires patience and clear communication about why a different approach would better serve the submission.

When recommendations are embraced, the benefits are tangible. Teams consistently report that getting the ADaM datasets right the first time requires less overall effort than correcting them later. The mentoring that occurs during the review process builds lasting knowledge within the team, improving their approach to proper ADaM construction on future studies. For complex studies in particular, early feedback has proven to make the work significantly more manageable, reducing the burden on programmers and reviewers alike.

The outcomes of incorporating senior reviewers into the study lifecycle have been consistently positive. While teams are sometimes reluctant at the outset, they typically come to welcome senior reviewers as a valued part of the process — not as members of the project team, but as an independent resource whose involvement strengthens the work.

Submissions that have undergone senior review are cleaner, with better traceability and documentation. This translates directly into fewer questions from regulatory agencies, and fewer long weekends spent responding to them. Perhaps most importantly, the engagement of senior reviewers contributes to the ongoing development of teams in their understanding and application of ADaM, helping to prepare them for newer standards as they are developed.

ADRG Content: Describing Data Issues and Enhancing Review

Section 2 of the FDA Study Data Technical Conformance Guide recommends including the Analysis Data Reviewer's Guide (ADRG) as an important part of a submission. The use of an ADRG allows for one point of reference for FDA reviewers.

The ADRG template, developed by PHUSE, contains the following sections where different types of data issues can be explained**Invalid source specified.**:

- PHUSE template Section 3.3: Subject issues that required special analysis rules are to be documented in the ADRG. For example, if there are subjects who received the wrong treatment or dose, or who were excluded from one or more analysis populations, this section is the best place to describe what actually happened to these subjects during the study, and how they were handled for analysis.
- PHUSE template Section 4: Analysis data creation and processing issues, such as split datasets, data dependencies, and intermediate datasets should be described. The purpose of this section is to describe important relationships between datasets, such as dependencies between split datasets or analysis datasets beyond ADSL, and the creation of intermediate datasets to provide traceability in complex derivations.

  For example, in an oncology study, parameters used for time-to-event analyses may be derived from multiple source datasets. In that case, an intermediate dataset can be used to gather all of the events that need to be considered when determining whether a particular event occurred or a subject was censored in order to provide a complete picture of how a subject progressed through a study. The structure and use of this intermediate dataset would be described in this section.

  Examples of how to document data dependencies in a table layout and in a flow diagram are shown in Table 3 and Figure 1, respectively.

Dataset	Dataset Label	Dependencies
ADRESP	Disease Response Analysis Data	ADSL, ADRS, ADTR
ADEXSUM	Exposure Summary Data	ADSL, ADEXP

Table 3: Dataset Dependencies documented in a table layout

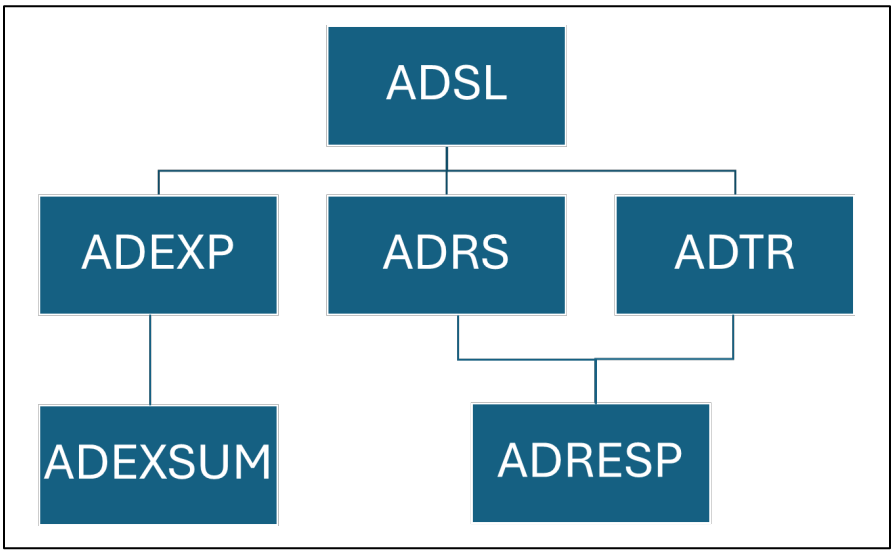


Figure 1: Dataset Dependencies documented in a flow diagram

- PHUSE template Section 6.2: ADaM has conformance rules, and issues found by automated tools that are built on these rules should be reported and explained in this section. It is understood that most studies will incur some sort of data anomalies as clinical trials are run by humans on other humans. In fact, it's an unusual study that does not experience any conformance check issues. Descriptions of the root causes of these issues should be sufficiently detailed to provide assurance that the sponsor has investigated and understood what actually happened.

  For example, one of the authors worked on a study where the subject died while receiving study drug. The subject was an organ donor, and hospital policy required that donors remain connected to everything until organ harvesting. Since they didn't harvest organs until the next day, the subject's last day on study drug was a day later than their death date. Of course, several automated checks fired, because you wouldn't normally expect a person to be dosed after death. In this case the study data was legitimate, should not be changed, but needed an explanation for a reviewer to understand it.

To enhance review, ensure the ADRG is consistent with other submitted content and includes appropriate links and references. This includes:

- PHUSE template Section 1.3: Ensure the dictionary version of MedDRA and WHODRUG match the version in the study's Clinical Study Data Reviewer's Guide (cSDRG).
- PHUSE template Section 2.1: Ensure the protocol title matches the protocol title in the cSDRG.
- PHUSE template Section 5.2:
  - Ensure the order and structure of all ADaM datasets matches the order and structure in the define.xml.

- Ensure that each ADaM dataset listed here is hyperlinked to the section in the ADRG describing the contents of that dataset.
- Ensure each dataset description designates the analyses supported by it.
- For each derived parameter, ensure the method of derivation is either described or referenced.

CONCLUSION

This paper described multiple cases where we've seen ADaM developers producing deliverables that were less than optimum. In each case, we referenced appropriate documentation, explained the situation and why it was either wrong or less than optimum, and provided a better solution. We hope that not only will this paper help with these specific issues but also provide a way of thinking that can help in other situations.

REFERENCES

CDISC ADaM documents are all available at https://www.cdisc.org/standards/foundational/adam. This paper referenced the following documents:

- Analysis Data Model Implementation Guide (ADaMIG) v1.3
- ADaM Structure for Occurrence (OCCDS) Implementation Guide v1.1
- ADaM Conformance Rules v4.0
- ADaM Examples of Traceability v1.0
- ADaM Oncology Examples Document v1.0

CDISC Define-XML is available at https://www.cdisc.org/standards/data-exchange/define-xml.

CDISC SDTM is available at https://www.cdisc.org/standards/foundational/sdtm.

CDISC Terminology is available at https://www.cdisc.org/standards/terminology/controlled-terminology.

CDISC Therapeutic Area User Guides (TAUGs) are available at https://www.cdisc.org/standards/therapeutic-areas/published-user-guides. This paper referenced the following TAUGs:

- Breast Cancer
- Prostate Cancer

The PHUSE Analysis Data Reviewer's Guide (ADRG) package is available at https://advance-phuse.atlassian.net/wiki/spaces/WEL/pages/26804660/Analysis+Data+Reviewer+s+Guide+ADRG+Package.

US FDA Study Data Technical Conformance Guide is available at https://www.fda.gov/regulatory-information/search-fda-guidance-documents/study-data-technical-conformance-guide-technical-specifications-document.

See the following conference papers for more information on some of the topics covered in this paper:

- ADaM Grouping: Groups, Categories, and Criteria. Which Way Should I Go? Shostak, J. PharmaSUG 2017 – Paper DS17.
- ADaM Pet Peeves: Things Programmers Do That Make Us Crazy. Brucken, N. and Minjoe, S. https://pharmasug.org/proceedings/2025/DS/PharmaSUG-2025-DS-105.pdf.
- Avoiding Sinkholes: Common Mistakes During ADaM Data Set Implementation. Watson, R. and Miller, K. https://pharmasug.org/proceedings/2018/DS/PharmaSUG-2018-DS05.pdf.

- Proper Parenting: A Guide in Using ADaM Flag/Criterion Variables and When to Create a Child Dataset. Watson, R., Miller, K., and Slagle, P. https://pharmasug.org/proceedings/2015/DS/PharmaSUG-2015-DS08.pdf

ACKNOWLEDGMENTS

We would like to thank the active members of the CDISC ADaM team who have provided insight into many of the issues presented here.

CONTACT INFORMATION

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

Nancy Brucken: Nancy.Brucken@iqvia.com

Nate Freimark: NFreimark@thegriessergroup.com

Sandra Minjoe: Sandra.Minjoe@iconplc.com

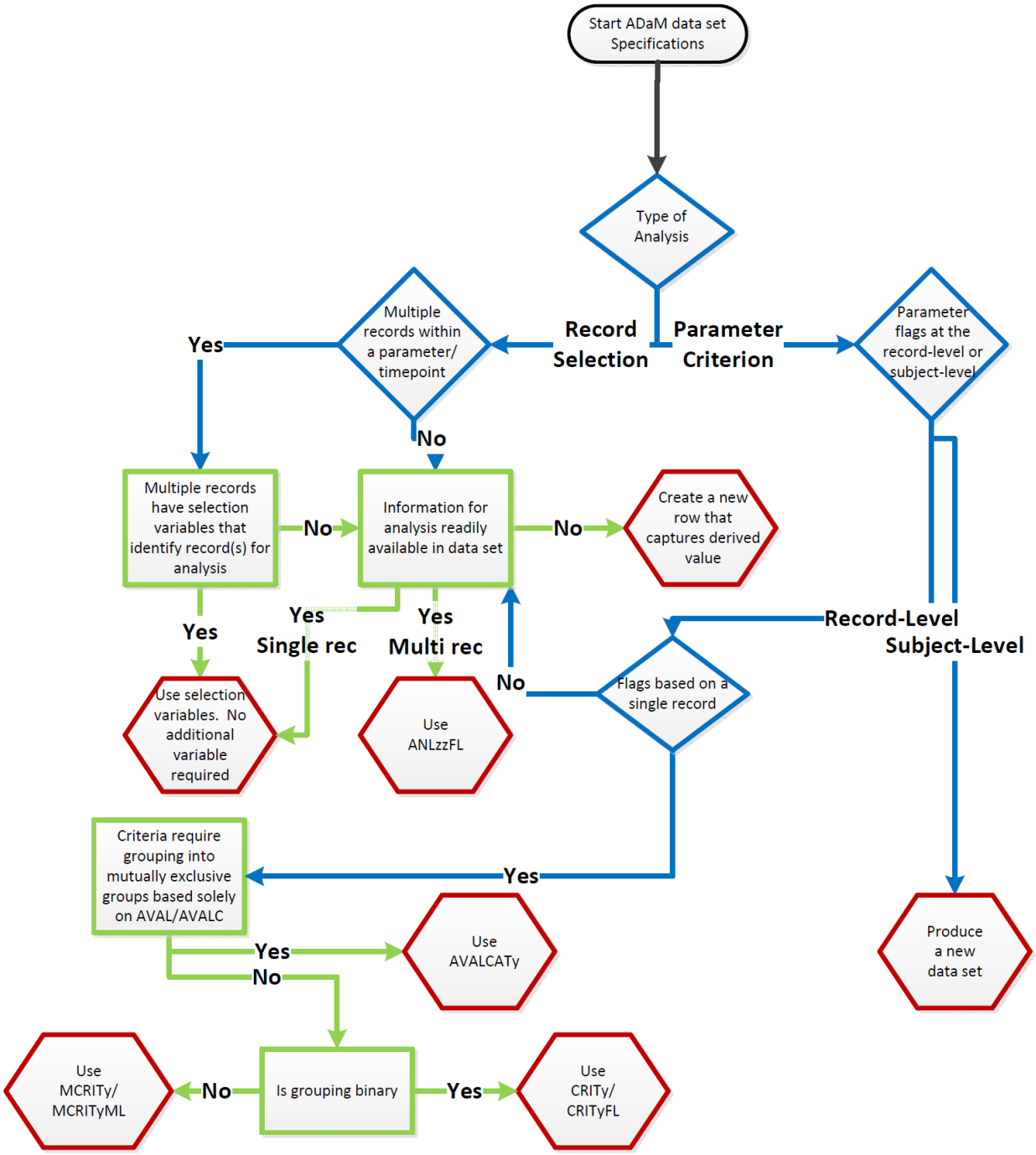
Paul Slagle: Paul.Slagle@iqvia.com

Tatiana Sotingco: TSotingc@its.jnj.com

Richann Watson: Richann.Watson@datarichconsulting.com

Alyssa Wittle: Alyssa.Wittle@atorusresearch.com

APPENDIX – NAVIGATING CATEGORY AND CRITERIA VARIABLES



(Watson, Miller, & Slagle, 2015)