Paper SI03 - PHUSE EU 2025
The Chicken, the Egg, and the ADaM:
Cracking the PRE-ADSL vs ADSL Sequence

Carla Santillan, Fortrea, Brussels, Belgium
Dubravka Georgijević, Fortrea, Leiden, the Netherlands

ABSTRACT
ADaMIG v1.2 introduced the concept of Pre-ADSL, an ADaM OTHER dataset intended for use when certain ADSL variables are better derived in other ADaM datasets. Our collaborator adopted this concept and formalized it as ADCORE in their ADaM standard.
We share our experience from a study where ADCORE was the first created ADaM domain and included all standard ADSL variables except those based on efficacy endpoints, which were derived using downstream efficacy ADaMs. We reflect on past studies where ADCORE was not used but, in hindsight, we feel it would have improved workflow and traceability, and we present a case where its use was limited - specifically, when ADSL analysis set flags needed early availability, but derivations depended on complex relative dose intensity calculations. This presentation aims to provide a balanced perspective on integrating ADCORE into ADaM workflows and when its use may - or may not - be beneficial.

INTRODUCTION
Our collaborator introduced the concept of ADCORE (Pre-ADSL) alongside a detailed implementation of complex calculations for asthma and chronic obstructive pulmonary disease (COPD) studies. These calculations primarily involve composite endpoints consisting of multiple event types. Before applying this approach in a respiratory study, we reviewed available CDISC (Clinical Data Interchange Standards Consortium) documentation related to pre-ADSL (Subject-Level Analysis Dataset) concepts.
In this paper, we share the requirements for one study, how we implemented the pre-ADSL concept, and how we documented it. We also reflect on two additional studies: one where this concept could have been applied to improve workflow and traceability, and another where a slightly different approach to pre-ADSL could have been beneficial. Based on our experience across these three examples, we present the advantages and limitations of using pre-ADSL.
This paper is organized into the following sections:
- Documentation available at CDISC
- Pre-ADSL collaborator implementation
- Implementation example
- Potential Situations Benefiting from This Approach
- Documentation best practices
- Comparison of Pre-ADSL approaches

Documentation Available at CDISC
The concept of a pre-ADSL dataset is briefly mentioned in ADaMIG v1.2[1] and ADaMIG v1.3[2], specifically in Section 3.2: *"There may be situations where highly derived variables are to be included in ADSL, yet the derivation of these variables may better be performed in another ADaM (Analysis Data Model) dataset."*

A more detailed explanation is provided in ADaM Examples of Traceability v1.0[3], Section 2.9, which presents ADSL as an intermediate dataset. A comprehensive example is discussed in the Rheumatoid Therapeutic Area User Guide v1.0[4], Section 5.1.1 *"Additional Highly Derived Baseline Values."* This example includes a flowchart illustrating how intermediate datasets were used and offers guidance on how to document them in the ADaM specifications and the Analysis Data Reviewer's Guide Completion Guidelines (ADRG) [6].

To ensure compliance with the ADaM v1.3 [2] standard, the following principles should be applied when developing ADaM datasets and associated metadata:
- **Clarity and Transparency**: Datasets must clearly and unambiguously communicate their content and origin, supporting the statistical analyses conducted in the clinical study.
- **Source and Derived Data**: ADaM datasets typically include both source and derived variables. Source data should be referenced in Define.xml (metadata of a study's datasets) as *Predecessor*, while derived variables must include a clear algorithm describing the derivation logic, including references to source SDTM (Study Data Tabulation Model) domains (e.g., EX, QS, FA) and any applied methods (e.g., imputation, aggregation).
- **Traceability**: The lineage of each variable must be evident, showing the relationship between derived

elements and their immediate predecessors. This traceability ensures that reviewers can follow the derivation path and understand the rationale behind each transformation.

By adhering to these principles, ADaM datasets maintain regulatory compliance and support reproducibility and transparency in clinical data analysis.

To further support conformance with ADaM standards, we refer to section 2.3.1 of the ADaM Implementation Guide v1.3 [2], which outlines key principles related to the ADSL dataset. The following excerpt serves as a foundation for our approach:

The ADaM Subject-Level Analysis Dataset (ADSL)

It should be noted that although the ADSL contains subject-level variables that are also important in other datasets, there is no requirement that every ADSL variable be present in other analysis datasets. However, at a minimum, any ADSL variable needed to enable analysis (e.g., statistical model covariates, population flags, subgrouping variables) should appear in the analysis dataset. Other ADSL variables may also be included for traceability or other reasons. A variable that is present in both ADSL and any other ADaM dataset must have the same values, type, and label. Note that the FDA sdTCG [5] (Food and Drug Administration Study Data Technical Conformance Guide) requests that "core" subject-level variables be present in all analysis datasets. For all regulatory submissions, refer to relevant regulatory agency requirements and discuss with the reviewing agency.

Pre-ADSL Collaborator Implementation

The strategy outlined below was defined by the collaborator to support the creation of ADaM datasets that are both easy to understand and traceable to their data origin. Complex derivations that were difficult to trace have been broken down into simpler components with clearly defined predecessors.

The collaborator introduced a pre-ADSL domain named ADCORE, which is restricted to using only SDTM datasets as source material. ADCORE includes all ADSL variables and core variables, except those that require other ADaM datasets for derivation (e.g., ADY, APHASE, ANXXFL). This pre-ADSL domain is then used across other ADaM domains to perform necessary derivations. Finally, all ADCORE variables, along with additional ADSL variables derived in other domains, are consolidated into the final ADSL dataset. See Figure 1.

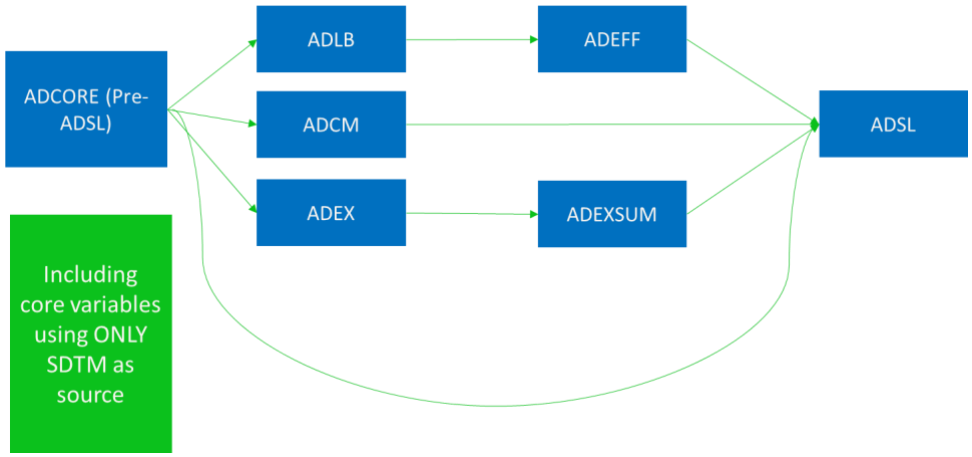

Figure 1 - Pre-ADSL collaborator implementation

Implementation Example

Example 1 - complex calculation composite endpoint consisting of multiple event types

In this example, the derivations required for composite endpoints, consisting of multiple event types for the study, were sufficiently addressed by the collaborator's proposal. This approach leveraged core variables at the initial stage, incorporated intermediate ADaM datasets, and prioritized the execution of domains based on analytical relevance.

Our objective was to derive complex composite endpoints, where the individual input parameters could themselves be other primary or secondary endpoints. Each intermediate dataset was used to simplify the final derivation. Several Basic Data Structure (BDS) domains were developed to include binary (Yes/No) parameters and other identifying variables.

These intermediate components were then combined in a final derivation within the last BDS domain, which defined whether a patient met the composite endpoint, based on the occurrence of any predetermined component events. Ultimately, the ADSL dataset captured variables associated with these derived concepts, ensuring traceability and transparency throughout the derivation process.

Domains names were changed to generic names for this example. See below a flow chart Figure 2 showing interdependencies of datasets:

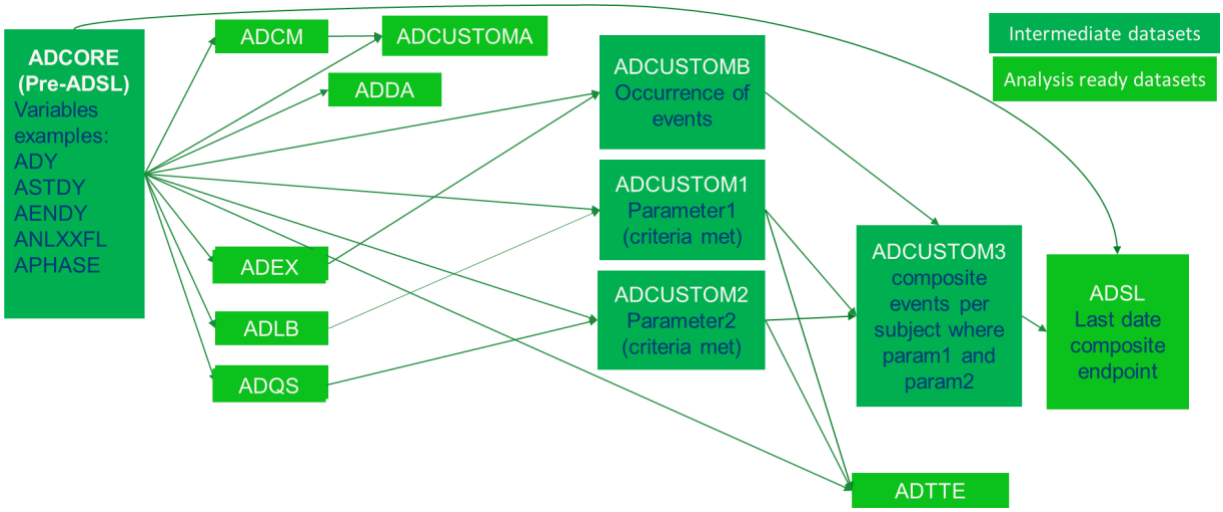

Figure 2 - Final derivation in ADSL after other domains

The flowchart presented in Figure 2 supports the identification of intermediate datasets, which are comprehensively documented in the Define.xml, and the Analysis Data Reviewer's Guide (ADRG).

Integration of Core Variables across ADaM Domains: To ensure consistency across ADaM datasets, core variables from the ADCORE dataset are embedded into all relevant ADaM domains (certainly in the analysis ready domains). This is achieved by introducing an additional column, '*ADCORECore*', within the ADaM specifications to flag these variables. A macro facilitates the automatic inclusion of flagged core variables into any domain, streamlining the programming process. When updates to the core variable list are required, only minimal modifications are necessary within the ADaM programs. See Figure 3.

Ord	Dataset	Variable	Label	Data Type	Origin	Predecessor	Derivation rule	Core	ADCORECore
1	ADCORE	STUDYID	Study Identifier	Char	Predecessor	DM.STUDYID		Req	
2	ADCORE	USUBJID	Unique Subject Identifier	Char	Predecessor	DM.USUBJID		Req	
3	ADCORE	SUBJID	Subject Identifier for the Study	Char	Predecessor	DM.SUBJID		Req	Y
4	ADCORE	SITEID	Study Site Identifier	Char	Predecessor	DM.SITEID		Req	Y
5	ADCORE	ARM	Description of Planned Arm	Char	Predecessor	DM.ARM		Req	Y
6	ADCORE	ARMCD	Planned Arm Code	Char	Predecessor	DM.ARMCD		Perm	Y
7	ADCORE	ACTARM	Description of Actual Arm	Char	Predecessor	DM.ACTARM		Perm	Y
8	ADCORE	ACTARMCD	Actual Arm Code	Char	Predecessor	DM.ACTARMCD		Perm	Y

Figure 3 – core variables set up

Execution Priority of ADaM Dataset Programs: The execution priority of ADaM dataset programs is defined within the *deliverables* file, as illustrated in Figure 4. Specifically, the 'PROGRAM_BATCH_ORDER' column in this file determines the sequence in which the datasets are processed. The *Run All ADaM* program references this order to ensure that each dataset is executed in alignment with the predefined priority, facilitating a structured and reproducible workflow.

.

TYPE	PROGRAM_NAME	PROGRAM_BATCH_ORDER
adam	adcore	1
adam	adsl	7
adam	adae	2
adam	adcm	2
adam	adex	2
adam	adlb	2
adam	adqs	2
adam	adcustoma	3
adam	adcustomb	3
adam	adcustomc	3
adam	adcustom1	4
adam	Adcustom2	4
adam	Adcustom3	5
adam	adtte	6

Figure 4 – use of 'deliverables' file to set up priority of ADaM Programs

Other uses of this 'deliverable' file are described in Paper SM16 - PHUSE EU 2025 [9]

Potential Situations Benefiting from this Approach

Example 2A – Handling Variables Related to Subsequent Anticancer Therapy (SCT)

In oncology trials, subsequent cancer therapies (SCTs) refer to treatments administered after the protocol-defined therapy, typically following disease progression. These therapies can have a substantial impact on overall survival (OS) analyses and must be carefully accounted for in the data.

In this example, identical derivations and imputations were required in both ADSL and ADCM (Concomitant Medications Analysis Dataset). Specifically, for records where Concomitant Medication Subcategory, CMSCAT = 'POST TREATMENT CANCER THERAPY', partial dates for SCT were calculated in ADCM. While the start date of subsequent cancer therapy (SCTDT) was needed in ADSL. However, due to the original linear data flow, where ADSL is created prior to other ADaM domains, the same derivation had to be implemented twice. This introduced the need for consistency checks between the two datasets, which was addressed with a macro to ensure alignment and reduce redundancy. See Figure 5 - example 2A

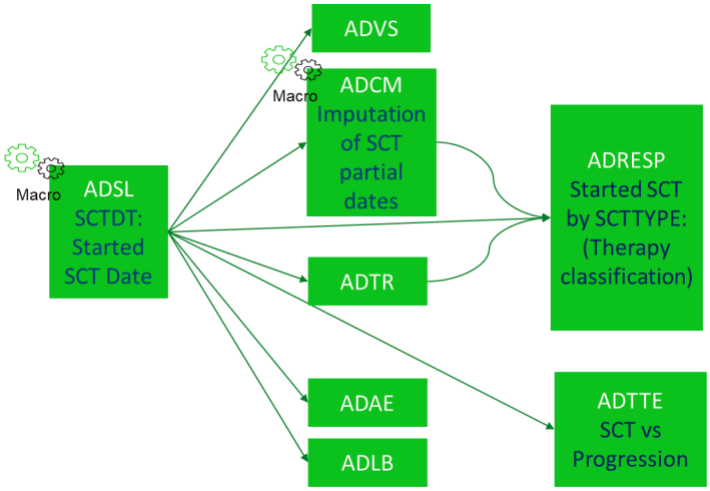

Figure 5 – example 2A - Macro applied in both ADSL and ADCM

Example 2B – Handling Variables Related to Subsequent Anticancer Therapy (SCT) Using pre-ADSL

An alternative approach to streamline partial date derivation involves leveraging the ADCORE dataset, as illustrated in Figure 6 – example 2B. Through this method, partial date derivation is performed once within the ADCM domain, and the resulting variables are subsequently reused in ADSL. This reuse enables ADSL to serve as a source for further derivations across additional ADaM domains, promoting consistency and reducing redundancy in programming efforts.

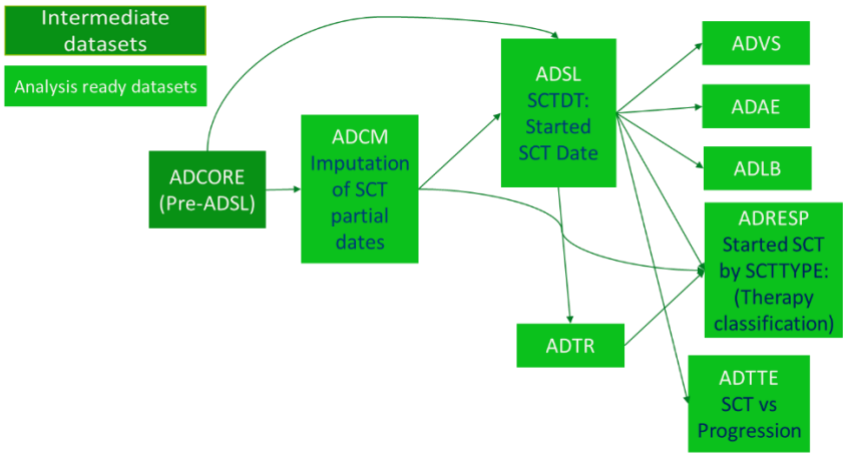

Figure 6 – example 2B - Pre-ADSL used for staging simple logic

Once ADSL SCTs variables are derived, they can be used for further analyses, such as:
Sensitivity analyses of OS, Subgroup analyses (e.g., OS by SCT status), or Adjustment in statistical models.

This example 2 with two approaches A and B illustrates a straightforward derivation that can be efficiently managed by both approaches. Both maintain full traceability while adhering to CDISC and FDA (*Food and Drug Administration)* requirements for the ADSL dataset. Therefore, it also raises an important consideration in ADaM programming: identifying the most efficient and transparent method that preserves traceability throughout the derivation process.

Example 3 – Handling Relative Dose Intensity (RDI) and Related Variables

The *CDISC ADaM Oncology Examples v1.0* [3] document provides comprehensive guidance on applying ADaM standards in oncology clinical trials, including examples of Relative Dose Intensity (RDI) derivations using pre-ADSL logic.

RDI is a critical metric in oncology studies, representing the ratio of the actual dose received to the planned dose over a defined period. It serves as an indicator of treatment adherence and tolerability.

In this modular oncology study, each module required three distinct RDI calculations. As the number of modules increased, so did the number of ADSL variables associate with RDI. To support these derivations, the intermediate dataset ADEXP was constructed from the SDTM domains EX (Exposure) and EC (Exposure as Collected). ADEXP (ADaM Exposure) provided the necessary exposure metrics, which were then summarized in ADEXSUM (ADaM Exposure Summary). This summary dataset included key parameters such as RDI, calculated as: RDI= 100* Actual Dose Intensity / Planned Dose Intensity

This structured approach enabled consistent and scalable derivation of RDI-related variables across multiple modules.

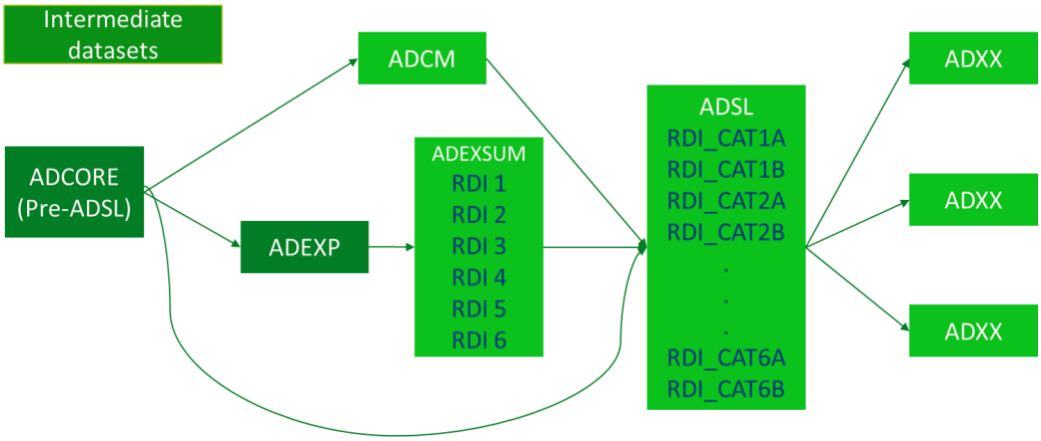

Figure 7 - Pre-ADSL used for staging complex logic, many variables

Once derived, RDI-related variables are merged into ADSL, see below some ADSL example variables:

- RDI: Numeric value of relative dose intensity.
- RDIFL: Flag indicating low RDI.
- RDI_CAT: Categorical grouping (e.g., <85%, 85–100%, >100%).
- POP01FL: DLT flag (e.g., <85%, only in Cycle1).

The variables derived for Relative Dose Intensity (RDI) support downstream analyses such as evaluating correlations between RDI and efficacy or safety outcomes, as well as conducting subgroup analyses based on dose adherence. As a result, the ADSL dataset must be available prior to the creation of other ADaM domains to ensure these analyses are feasible.

This example (Figure 7) shares similarities with Example 2 but involves a more complex derivation and a larger set of variables. The benefits of calculating all related variables within a single dataset are even more pronounced in this context. Each variable is clearly traceable to its source, and the derivation process is simplified, enhancing both transparency and consistency.

Documentation Best Practices

The use of Pre-ADSL should be documented in Define and in ADRG.

Documenting Pre-ADSL Derivations in Define.xml

Best Practices, CDISC documentation does not specify formal requirements for documenting the use of pre-ADSL in Define.xml, but it does provide best practice guidance on how to handle derivations coming from another ADAM datasets. However, in Rheumatoid Arthritis Therapeutic Area User Guide v1.0 [4] it mentions source variables from pre-ADSL as predecessor when they are used in derivations.

When variables are derived in a pre-ADSL staging dataset, this should be clearly documented in the Define.xml to ensure transparency and traceability. The following sections of Define.xml are recommended for this purpose:

Variable-Level Metadata (Derivation/Comment Field):
Indicate that the derivation was performed in a pre-ADSL dataset. As with any other variable, provide details on the derivation logic, including the source SDTM domains used (e.g., EX, QS, FA), the applied methodology (e.g., imputation, aggregation), and references to programming specifications or the Statistical Analysis Plan (SAP), if applicable.
Dataset-Level Metadata (ADSL Description):
Include an explanatory note in the ADSL dataset description, such as:
"Some variables in ADSL were derived using a pre-ADSL staging dataset that followed ADaM principles but did not include all required ADSL variables."

This documentation approach ensures that derivations performed outside the final ADSL dataset remain fully traceable and compliant with CDISC standards.

Documenting Pre-ADSL Derivations in ADRG
Best Practices, from CDISC ADSL example [7]

- The pre-ADSL dataset should follow ADaM principles, even if it doesn't meet all ADSL requirements.
- It should be documented clearly in the ADRG to ensure regulatory transparency.
- Variables derived in pre-ADSL should be traceable to their SDTM origins, and this traceability should be described in the ADRG.

When a pre-ADSL staging dataset is used to support complex derivations, it should be clearly referenced in the following sections of the ADRG to ensure transparency and traceability:

- **Section: Data Derivations and Traceability**
  This section should describe how key variables in ADSL were derived. If a pre-ADSL dataset was used to stage complex derivations—such as treatment start dates, baseline scores, or population flags—it should be explicitly mentioned. Details should include the derivation logic, source SDTM domains, and any relevant programming specifications or references to the Statistical Analysis Plan (SAP).
- **Section: Analysis Dataset Overview**
  When describing the ADSL dataset, include a note indicating that a pre-ADSL dataset contributed to its construction. Clarify the purpose of the pre-ADSL dataset, such as staging derivations, performing imputations, or applying complex logic prior to final dataset creation.
- **Section: Programming Notes or Assumptions**
  If any assumptions or imputation rules were applied within the pre-ADSL dataset, such as handling partial dates, and these are not documented elsewhere (e.g., in the SAP), they should be recorded here. This helps reviewers understand the rationale and methodology behind the derived variables and ensures consistency across documentation.

Comparison of Pre-ADSL Approaches across Previously Shared Examples:
To evaluate the practical implications of using a pre-ADSL staging dataset, we compared the presented examples, each illustrating a slightly different implementation strategy. All approaches comply with CDISC and FDA requirements, ensuring regulatory alignment. However, the use of pre-ADSL should include the automation of adding core variables in all other analysis ready domains.
The comparison focuses on the advantages and limitations of each example.

Example	Approach	Pros	Cons
1. ADSL at the end	Final derivation in ADSL after other domains	- Improved traceability through modular derivations	- Requires careful management of interdependencies - If needed, changes to final derivation must be propagated backward
2A. Same derivation using a macro twice	Macro applied in both ADSL and ADCM	- Efficient for simple derivations - Avoids circular logic - Core variables well documented in ADSL - No additional specifications needed	- Requires consistency checks across domains with duplicated logic
2B. ADSL in the middle (simple derivation)	Pre-ADSL used for staging simple logic	- Core variables documented once in pre-ADSL - Automatic inclusion of core	- Additional documentation required for pre-ADSL - Limited benefit for simple derivations?

Example	Approach	Pros	Cons
		variables downstream domains	
3. ADSL in the middle (complex derivation)	Pre-ADSL used for staging complex logic, many variables	- Core variables derived once and reused - Simplifies updates when logic changes - Enhances traceability	- Requires careful documentation of pre-ADSL and other intermediate datasets.

Conclusion: Balancing Clarity, Efficiency, and Traceability in ADSL Derivations

Our experience across the presented examples highlights the value of using a pre-ADSL staging dataset, particularly when managing complex derivations or when multiple ADaM domains rely on shared logic. This approach enhances traceability, reduces redundancy, and simplifies updates. However, it also introduces additional documentation requirements and demands careful attention to metadata consistency.

For simpler derivations, applying a logic directly in two domains by using a macro may be tempting, but the need to ensure consistency that is maintained across these datasets. The choice of approach should be guided by the complexity of the derivation, the need for traceability, and regulatory compliance.

Introducing an intermediate dataset, commonly referred to as pre-ADSL, can sometimes been seen as an increase of documentation effort, but it often improves clarity and traceability, especially when derivations span multiple domains. These intermediate datasets may follow the ADaM OTHER class or, in some cases, adopt a BDS structure. While not intended for analysis, they serve as valuable tools for staging derivations. Note that if any variables from these datasets are required for analysis, they must be transferred into ADSL or another analysis-ready ADaM dataset.

Ultimately, the decision to use a pre-ADSL dataset should be based on a careful assessment of derivation complexity and the benefits of centralized logic. When used appropriately, it supports a more transparent and maintainable programming strategy aligned with CDISC and FDA expectations**.**

REFERENCES

[1] CDISC: ADaM V1.2
https://www.cdisc.org/standards/foundational/adam/adam-v2-1

[2] CDISC: ADaM V1.3
https://library.cdisc.org/browser/#/mdr/adam/adamig-1-3

[3] ADaM Examples of Traceability v1.0
https://www.cdisc.org/standards/foundational/adam/adam-examples-traceability-v1-0

[4] Rheumatoid Arthritis Therapeutic Area User Guide v1.0
https://www.cdisc.org/standards/therapeutic-areas/rheumatoid-arthritis/rheumatoid-arthritis-therapeutic-area-user-guide

[5] FDA Study Data Technical Conformance Guide
https://www.fda.gov/media/88173/download

[6] PHUSE: ADRG V1.2
https://phuse.s3.eu-central-1.amazonaws.com/Deliverables/Optimizing+the+Use+of+Data+Standards/ADRG+Packages+V1.2.zip

[7] CDISC ADSL example: ADaM Subject-level Analysis - ADSL Dataset
https://www.cdisc.org/kb/examples/adam-subject-level-analysis-adsl-dataset-80283806

[8] Korak Datta, AstraZeneca, Cambridge,UK
Taniya Chatterjee, AstraZeneca, Warsaw, Poland
Paper DH04: Pre ADSL – Revolutionize your ADaM workflow
PHUSE EU CONNECT 2024 Strasbourg

[9] Hongyu Liu, Vertex Pharmaceuticals Incorporated, Boston, MA Hang Pang, Vertex Pharmaceuticals Incorporated, Boston, MA
Paper PO22: Challenges in Developing ADSL with Baseline Data
PharmaSUG 2015

[10] Elizabeth Dennis, EMB Statistical Solutions LLC;
Grace Fawcett, Syneos Health

Paper DS-63: The Winding Road to ADSL
PharmaSUG 2025

[11] Sandra Minjoe, Accenture Life Sciences, San Francisco, USA
Paper DS11: It Depends On Your Analysis Need
PharmaSUG 2015

[12] Dubravka Georgijević, Fortrea, Leiden, the Netherlands
Paper SM16: On the Right Track: Managing Platform Modular Studies with a Central Metadata Switchboard
PHUSE EU CONNECT 2025 Hamburg

CONTACT INFORMATION

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

Carla Santillan
Fortrea Belgium SRL
Avenue du Bourget 50
Brussels 1130
Belgium
carla.santillan@fortrea.com

Dubravka Georgijević
Fortrea Netherlands BV
Schipholweg, 111 2316 XC
Leiden, The Netherlands
dubravka.georgijevic@fortrea.com