PharmaSUG 2026 - Paper DS-111

Ready for Next Level SDTMs and ADaMs Compliance with End-to-End Processing?

Sunil Gupta, Senior CDISC Consultant, SAS & R Trainer, Gupta Programming
Tomás Sabat Stofsel, CEO and Co-Founder of Verisian

ABSTRACT

From specifications to results, smarter tools enable better SDTM and ADaM compliance for higher quality submissions. This is achieved by anticipating common issues and checklists during the SDTM and ADaM mapping process. While many organizations have SOPs for QC, smarter organizations incorporate continuous monitoring of program development.

First, SDTM and ADaM specifications and SAPs are imported for metadata processing and compliance checking. This metadata is essential for cross-referencing the logic of all derived SDTM and ADaM variables. While SDTM and ADaM structures and rules must pass the litmus test, the single story message from raw data must be transparent and consistent with SDTMs and ADaMs.

Finally, SDTM and ADaM required variables must be created and all raw data must be mapped to SDTMs. With this end-to-end system, onboarding new members to the programming team becomes easier because the documentation and all related programs can be quickly explained.

THE ROLE AND EVOLUTION OF PROGRAMMING IN CLINICAL TRIALS FOR VALIDATION

Deliverables in clinical trials are created from a data management process that consists of subsetting, conditions, algorithms, summaries, and joins. Each component and step in this process is critical and needs to be performed correctly and completely. Quality control (QC) steps ensure a structured checklist and logical order execution to prevent sequence failures.

Because clinical trials are so expensive, sponsors have developed methods to eliminate risks associated with producing incorrect results. Thirty years ago, the only practical method to achieve this was to double program and compare the results. In today's world, with advancements in best practices and technology, the relevance of manually reviewing variable derivations is increasingly questioned.

Without the aid of advanced technology, organizations continue to manually select and review programs that call other programs to better understand exactly how and which source variables are used in complex variable derivations. By having systems that can read, understand and link related code snippets, programmer's time is better leveraged.

CURRENT SOLUTIONS TO DELIVER SDTMS AND ADAMS

Today's process of transforming raw data into SDTM and ADaM formats involves structured steps that include applying standard names, units, and control terminology, creating required variables, renaming variables, converting dates to ISO 8601 character format, removing all user-defined formats, and establishing paired and hierarchical variables. Additional steps involve ordering variables, sorting records, converting data into a vertical structure, creating supplemental domains when needed, assigning unscheduled visits, and, finally, conducting checks for SDTM and ADaM inconsistencies or data issues.

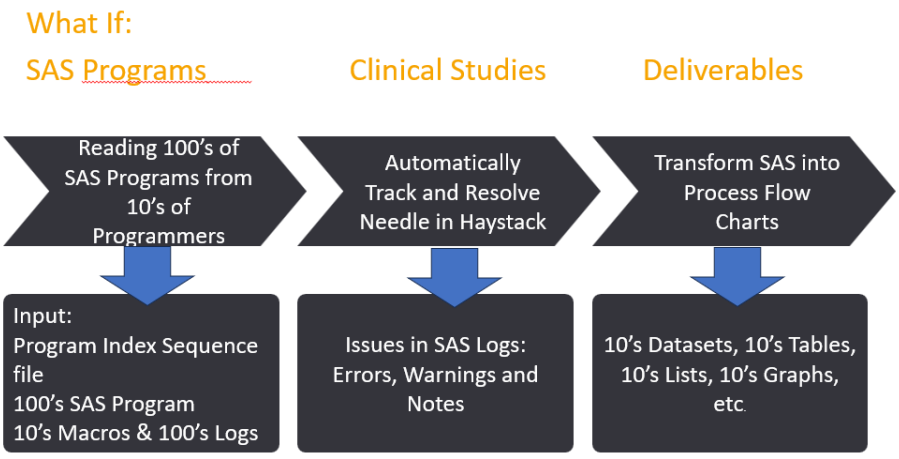
For variable derivations, multiple raw variables may be combined into a single SDTM variable or, conversely, a single raw variable may be split into separate SDTM variables. To ensure consistent control terminology transformation from raw to SDTM, a format catalog is often created and applied using the PUT() function, rather than relying on multiple IF-THEN statements, which can become cumbersome and prone to error.

Most SAS macro systems and tools provide predefined templates that automate some of this process.

These templates can save programming time and improve standardization across studies. However, they come with inherent limitations in terms of traceability and validation. Since SAS macros and programs often call multiple subprograms, achieving a complete end-to-end view of variable derivation can be challenging.

SHARPING THE SAW: SDTMS AND ADAMS

One of the biggest challenges for sponsors is that SDTMs and ADaMs deliverables may involve multiple international teams. Similar to a moving target, programming teams may change in experience level and their ability to document and trouble-shoot. If you expand this issue across tens of studies and programmers with hundreds of programs, then sponsors have to reply on the quality of the programs to be error free and SOP compliant!



With increased regulatory requirements requires an improvement from manual to automatic program traceability and advanced tools for compliance ready SDTMs and ADaMs. This is important since one clinical study can have many different methods of development which could all take hours to understand. Both development and qc of SDTMs and ADaMs can benefit from a higher standard and control of development methods. This means to fine-tune all metadata sources and minimum manual efforts.

Although double programming has long been the gold standard for validation, advancements in technology and regulatory guidance have shown that it is unnecessary. In fact, even though the FDA requires sponsors to SDTMs, ADaMs, and TLGs, they do not mandate how this validation should be performed.

In other words, double programming is not a regulatory requirement.

Instead, the FDA encourages a risk-based approach, allowing organizations to tailor their validation strategies based on the complexity of the study, the level of risk, and available technology.

By using, for example, validated SDTM and ADaM macros, organizations have decreased the risk of producing incorrect SDTMs and ADaMs; however, mistakes are still made. Some of these practices emphasize leveraging metadata from all available sources to ensure consistency and accuracy. For example, SDTM IG metadata provides predefined standards for variable order, names, lengths, types, labels, and sorting rules for each domain, reducing the potential for inconsistencies.

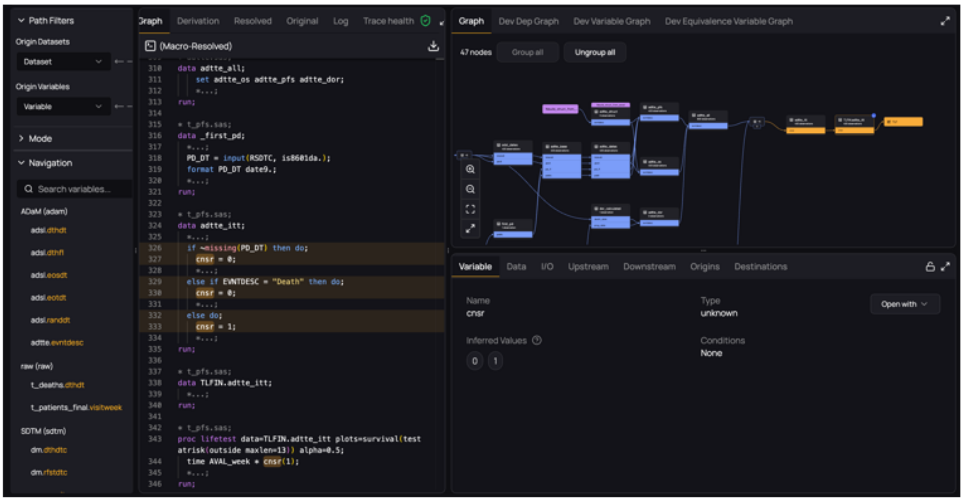
Despite these advancements, many organizations continue to document variable specifications in English sentences or pseudo-code, introducing potential inconsistencies. To ensure alignment, metadata vitals — such as frequency counts, unique and missing values of categorical variables, and means and ranges of continuous variables — must be checked across raw data and SDTMs. These vitals should be consistent across all data categories, including demographics, treatments, vitals, medical history, exposure, labs,

adverse events (AEs), and CMs.

By shifting away from rigid double programming requirements and embracing new approaches of validation, organizations can enhance efficiency while maintaining — or even improving — data accuracy and regulatory compliance.

SAS TOOLS AND TRACEABILITY-POWERED AI SYSTEM ALTERNATIVES TO DOUBLE PROGRAMMING

***What if** there is complete transparency between code, raw, SDTMs, ADaMs and TLGs with drill down features to confirm variable derivations?*



For the first time, organizations can improve their process with targeted validation powered by code traceability-driven AI systems. These systems provide a new level of transparency, empowering SAS programmers in their development and QC processes. Code is no longer just a tool for data transformation — it now serves as an input to metadata, improving organization, categorization, and analysis, and ultimately giving programmers a deeper understanding of the data. Additionally, AI enables more advanced processing and querying of large data sets, leading to faster and more reliable insights.

Since double programming is not a regulatory requirement, sponsors should adopt alternative validation methods. A fit-for-purpose approach to SDTM and ADaM validation should focus on two key questions: Do the results make sense? Are they consistent? While excessive data checks may seem beneficial, they often yield diminishing returns, consuming resources without significantly improving validation quality.

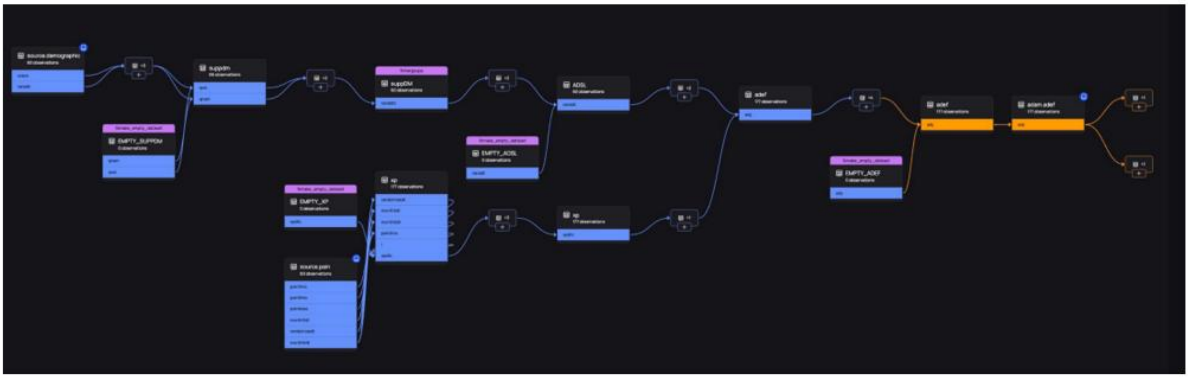
The *Good Clinical Practice Guide* outlines several acceptable validation approaches for statistical programming, including:

1. independent programming (dual programming and output comparison)
2. detailed output checks against raw data or data listings, combined with code review (essentially, system validation)
3. use of previously validated code or macros
4. retention and review of statistical software logs to confirm correct execution
5. verification of new variables and datasets derived from final data management outputs
6. review of formulae in spreadsheets, where applicable.

With traceability-powered AI systems, statistical programmers gain full transparency across raw, SDTM,

and ADaM variables, along with code, SDTM IG, ADaM IG, and variable mapping specifications. These systems allow programmers to trace each variable derivation from specification to raw data, through SDTMs, and into ADaMs, ensuring complete visibility before deploying SAS programs into production. A dashboard provides a visual representation of all related SAS programs, automatically linking them together so that every step in the data management process can be monitored in real time. This significantly reduces manual effort, eliminating the need for programmers to search through and manually piece together transformation logic across multiple SAS programs and macros.

***What if** code can be understood to 'visualize the process' by 'automatically linking program sections by data'?*



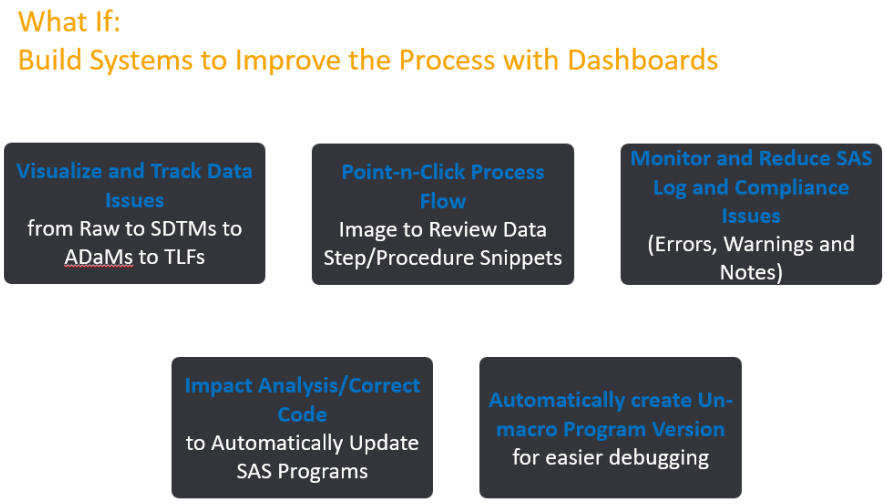
By replacing manual program review process with traceability-driven validation, organizations can ensure SDTM and ADaM compliance with confidence while freeing resources to focus on more complex tasks and additional clinical studies. However, statistical programmers must continue to adhere to high standards in program design and unit testing to maintain data integrity. Best practices include coding based on SOPs, deep knowledge of clinical data, self-QC against specifications, and comprehensive process documentation. Throughout development, programmers should actively address validation and debugging issues, including missing values, special characters, invalid syntax, and logical inconsistencies in code.

BENEFITS OF TRACEABILITY-POWERED AI SYSTEMS

While this paper has primarily focused on eliminating manual programming review for SDTMs, the same approach can be extended to ADaMs and TLGs. Since SDTMs are a regulatory requirement, they serve as an ideal starting point for automation and validation improvements. Once SDTMs are streamlined, the same structured workflow can be applied to ADaMs and TLGs, as these data sets build upon SDTMs and rely on multiple SAS programs and macros for processing.

Beyond eliminating the manual process, AI-driven enhancements provide additional efficiencies in clinical trial review, preparation, and analysis. These tools improve oversight and management of CRO deliverables, reducing unnecessary manual validation of SDTMs while maintaining compliance.

From a quality control perspective, AI-powered traceability systems enhance the review of not just the adherence to specifications but also of SAS logs, ensuring that outputs are free from critical errors, warnings, and key notes, such as uninitialized values, data conversions, invalid arguments, missing values, mathematical inconsistencies, and merge conflicts. By integrating traceability-powered AI tools, even the most risk-averse sponsors can confidently replace double programming with a structured, systematic validation approach, reducing redundancy while improving accuracy.



Below is a list of benefits of a traceability-powered AI dashboard for complete traceability between SDTM IG, variable specifications, raw data, SDTMs, and ADaMs.

1. Link all related code into a single SDTM and ADaM program.
2. Cross-reference specifications to identify gaps and ensure clarity in variable definitions.
3. Map all raw data to SDTM and ADaM variables.
4. Ensure comprehensive mapping of required SDTM domains and variables, including TA, DM, EX, DS, SRCDOM, and SRCVAR.
5. Track variable derivations from raw to SDTM and ADaM using frequency counts, means, and ranges.
6. Monitor code lists, frequency counts, and missing values to maintain data consistency.
7. Identify invalid metadata or control terminology for non-extensible SDTM and ADaM variables.
8. Track patient populations, subjects, and records across raw data, SDTMs, and ADaMs (e.g., ensuring IE ≥ DM ≥ EX > DV/AE/DS).
9. Ensure compliance with protocol requirements, such as dose, visits, cycles, phases, indications, exposure, and disposition.
10. Monitor primary and secondary efficacy measures and lab events across raw, SDTM, and ADaM data sets.
11. Track treatment-emergent AEs, hospital events, standardized medical queries (SMQs), deaths, and serious adverse events (SAEs).
12. Monitor demographic and subgroup vital metrics, including sex, age, race, populations, strata, treatments, and studies.
13. Track primary reasons for discontinuation across raw, SDTMs, and ADaMs.
14. Monitor updates to raw data throughout the study life cycle.

SUMMARY

Smarter organizations with strong leadership in the industry are already moving beyond manual program reviews. They recognize that they are not alone in this shift and that more efficient alternatives are readily available. Now is the time to challenge outdated program navigation, search and traceability practices and embrace modern technology-driven approaches.

REFERENCES

1. Achieve faster and higher-quality submission, Verisian White Paper. https://sassavvy.com/resources/SAS%20Downloads/Verisian%20Booklet.pdf
2. Does SDTM Validation Really Require Double Programming? Sunil Gupta, Tomás Sabat Stofsel https://www.lexjansen.com/pharmasug/2025/MM/PharmaSUG-2025-MM-227.pdf

CONTACTS

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

Sunil Gupta, Submission SME
Gupta Programming
213 Goldenwood Circle
Simi Valley, CA 93065
GuptaProgramming@gmail.com
SASSavvy.com, R-Guru.com

Tomás Sabat Stofsel
Verisian CEO and Co-Founder of Verisian
London, England, United Kingdom
tomas@verisian.com https://www.verisian.com