PharmaSUG 2026 - Paper TT-403

From SAP to CSR: A Metadata-Driven TFL Workflow

Bhavin Busa, Clymb Clinical

ABSTRACT

Clinical trial reporting workflows are often fragmented across disconnected tools and manual handoffs, resulting in inefficiencies, limited traceability, and challenges in maintaining consistency from analysis planning through Clinical Study Report (CSR) delivery. While standards such as CDISC ADaM and the recently published Analysis Results Standard (ARS) provide a strong foundation, many organizations still struggle to operationalize these standards across the full analysis results lifecycle.

This paper presents a metadata-driven framework that connects key stages of the analysis results pipeline: from Statistical Analysis Plan (SAP) specification through TFL design, automated result generation, centralized review, and downstream CSR integration. The proposed workflow demonstrates how structured metadata derived from the SAP and protocol can be used to prospectively define TFL shells, drive generation of Analysis Results Datasets (ARDs), automate production of tables, figures, and listings, and support collaborative review and version control in a centralized environment.

By treating analysis results metadata as a single source of truth, this approach improves traceability between planned analyses, executed programs, and reported outputs, while enabling automation, reproducibility, and reuse. Importantly, it establishes a robust foundation for AI-enabled clinical reporting by providing structured, machine-readable inputs aligned with ARS and ARD. The framework reduces manual programming effort, shortens review cycles, and supports scalable, future-ready statistical reporting workflows.

1. Introduction

Clinical Study Reports (CSRs) are critical deliverables in the clinical development lifecycle, summarizing study design, statistical analyses, and results for regulatory submission. Despite advancements in CDISC standards such as SDTM and ADaM, the workflow connecting analysis planning to CSR delivery remains fragmented, manual, and dependent on interpretation.

In traditional workflows, the Statistical Analysis Plan (SAP) is authored as a narrative document outlining the intended analyses. However, this document is not structured in a way that can be directly consumed by downstream systems. As a result, statistical programmers must interpret the SAP and translate it into TFL shells, derive datasets, and implement statistical analyses. Outputs are generated as static files, reviewed through disconnected processes, and manually incorporated into CSR documents.

This process introduces inefficiencies at multiple stages. Analysis definitions are reinterpreted repeatedly, leading to inconsistencies and increased quality control effort. Traceability between planned analyses and reported outputs is often incomplete, making it difficult to validate alignment during audits or regulatory reviews.

A fundamental limitation of traditional workflows is their reliance on unstructured content. Such workflows are not easily consumable by automated systems or artificial intelligence (AI). In contrast, structured metadata such as CDISC Unified Study Definitions Model (USDM), Analysis Concepts (AC), Derivation Concepts (DC), Analysis Results Standard (ARS), and Analysis Results Data (ARD) provides a machine-readable foundation that enables automation, validation, and AI-driven insights.

This paper presents a metadata-driven framework that transforms the workflow from SAP to CSR into a connected, structured, and AI-ready pipeline. By leveraging standardized metadata across the lifecycle, organizations can reduce manual effort, improve consistency, and enable next-generation automation in clinical reporting.

2. Current Challenges in Clinical Reporting Workflows

Clinical reporting workflows today are characterized by fragmentation across tools, manual interpretation of analysis plans, and inefficient review processes.

SAPs are typically authored in document formats such as Word or PDF, while TFL shells are created independently using spreadsheets or document templates. Programming is performed in SAS or R without direct linkage to SAP structure, and review processes rely on static outputs circulated via email or document tracking systems. This lack of integration leads to duplication of effort and misalignment across teams.

The manual translation of SAP content into programming specifications introduces variability. Different programmers may interpret analysis definitions differently, resulting in inconsistencies in output. This increases the burden on quality control processes and requires additional review cycles to reconcile discrepancies.

Traceability across the lifecycle from SAP to TFLs, programs, outputs, and CSR are often incomplete or maintained retrospectively. Manual traceability matrices are difficult to maintain and do not scale well across complex studies.

Review processes further compound inefficiencies. Static outputs such as RTF or PDF files are shared for review, with comments captured offline. This leads to multiple iterations, version control challenges, and delays in finalizing outputs. These challenges highlight the need for a more integrated and structured approach that connects all stages of the reporting lifecycle.

3. Metadata-driven Framework Overview

The proposed framework is built on the principle that analysis results metadata should serve as the single source of truth across the reporting lifecycle. Instead of relying on multiple disconnected representations of analysis definitions, this approach captures all relevant information in a structured, machine-readable format.

This metadata includes analysis intent, derivation logic, and display specifications. By centralizing this information, it can be reused across multiple stages of the workflow, reducing redundancy and improving consistency.

The framework connects key stages of the lifecycle, including SAP structuring, TFL shell generation, analysis execution, ARD generation, centralized review, and CSR integration. Each stage is driven by shared metadata, ensuring alignment between planning and execution.

A key advantage of this approach is that it is inherently AI-ready. Structured metadata creates a semantically rich data layer that AI systems can reliably interpret. This enables advanced capabilities such as automated validation, consistency checks, and narrative generation, which are not feasible in traditional unstructured workflows.

4. From SAP to Structured Metadata

The SAP is traditionally a narrative document designed for human interpretation. While it provides detailed descriptions of planned analyses, its unstructured format limits its usability for automation and downstream integration.

In the proposed framework, the SAP is transformed into a structured, machine-readable specification capturing endpoints, populations, parameters, timepoints, statistical methods, and display requirements. This structured representation serves as the foundation for all downstream processes.

A critical enabler of this transformation is the incorporation of CDISC Analysis Concepts (AC) and Derivation Concepts (DC). AC defines what is being analyzed, while DC defines how analysis variables and results are derived. These emerging CDISC constructs aim to standardize analysis definitions and derivations across studies [1].

By leveraging AC, organizations can define reusable analysis patterns such as change from baseline, adverse event summaries, or time-to-event analyses. This reduces ambiguity and ensures consistent interpretation of analysis intent. DC standardizes derivation logic, including baseline definitions, imputation methods, and statistical calculations, ensuring reproducibility and alignment.

Embedding AC/DC into SAP structuring enables prospective standardization, supports reuse across studies, and establishes a direct linkage between SAP definitions and ARS metadata. It also enhances AI interpretability by providing structured definitions that can be directly consumed by AI systems.

5. Automated TFL Shell Generation

Using structured metadata derived from the SAP, TFL shells can be generated automatically through template-driven approaches. Templates define layout, titles, footnotes, and formatting, while metadata provides the content.

A key innovation is the prospective generation of ARS metadata during shell creation. Each shell represents both a display and a structured definition of the corresponding analysis result. By embedding ARS concepts such as Analysis Result, Parameter, Population, and Method into shell generation, outputs are inherently aligned with standardized metadata [2, 3].

This approach aligns with implementations demonstrated through the CDISC eTFL Portal [4, 5].

Prospective ARS generation ensures consistency between planned analyses and outputs, eliminates retrospective mapping, and enables seamless integration with ARD generation and downstream automation. Prior work has demonstrated the value of ARS-driven automation [5, 6].

6. Metadata-driven Analysis and ARD Generation

Analysis Results Datasets (ARDs) represent the structured output layer of the framework, capturing analysis results along with contextual metadata aligned with CDISC ARS. Unlike traditional outputs, ARDs provide a machine-readable representation of results that can be reused across multiple downstream processes.

A key advancement in this approach is the ability to drive statistical programming directly from metadata, shifting the paradigm from manual coding to automated, metadata-driven execution. By leveraging ARS metadata and Analysis Display Metadata (ADM), programming frameworks can dynamically generate TFL programs aligned with predefined analysis definitions.

In practice, this approach is operationalized through metadata-driven frameworks implemented in both R and SAS environments. R-based frameworks such as *siera* and SAS-based frameworks such as *atlas* ingest ARS and ADM metadata to generate analysis programs in a standardized and scalable manner [7, 8].

6.1 Metadata-to-Code Flow

SAP (AC/DC structured)
→ ARS + ADM metadata
→ Metadata ingestion (siera / atlas)
→ Auto-generated programs
→ Execution on ADaM datasets
→ TFL outputs + ARD

This pipeline ensures that analysis intent, derivation logic, and display specifications remain aligned throughout execution.

6.2 AI-Assisted Code Generation

Building on the metadata-driven programming paradigm, AI-assisted code generation introduces an additional layer of automation. By leveraging ARS metadata, ADM definitions, and structured analysis specifications, AI-based tools can interpret analysis intent and generate executable programs aligned with predefined standards.

For example, AI models can:

- Translate ARS metadata into SAS or R code
- Generate derivation logic based on DC definitions
- Construct output programs aligned with ADM specifications

This approach reduces the need for manual coding while improving consistency and adherence to standards. It also accelerates development timelines, particularly for complex studies with large numbers of outputs [9].

Importantly, AI-assisted code generation operates on structured metadata, ensuring that generated code is aligned with standardized definitions and reducing the risk of inconsistencies.

6.3 Role of ARDs in Downstream Processes

ARDs generated through this process serve as a central output layer that supports multiple downstream workflows. Because ARDs include both results and contextual metadata, they provide a complete and reusable representation of analysis outcomes.

Key advantages of ARDs include:

- Traceability: Direct linkage between results and underlying metadata
- Reproducibility: Consistent representation of analysis outputs
- Reusability: Ability to reuse results across review, reporting, and CSR generation
- Interoperability: Compatibility with downstream systems, including visualization and AI tools

Unlike static outputs, ARDs enable dynamic interaction with results, supporting validation, rendering, and narrative generation without requiring reprocessing of raw datasets.

7. Centralized Review, Rendering, and Collaboration

A key advancement in the metadata-driven framework is the transition from static outputs to ARD-driven rendering, where Analysis Results Data (ARD) typically represented in structured formats such as JSON serves as the primary data layer for visualization, review, and reporting.

Instead of generating static RTF or PDF files, display engines consume ARD along with Analysis Display Metadata (ADM) to dynamically render tables, figures, and listings. This introduces a clear separation between data (ARD) and presentation (display templates), ensuring that outputs are consistently generated from a single source of truth. Any updates to underlying data or metadata are immediately reflected in rendered outputs, eliminating the need for regeneration of static files.

Centralized platforms (e.g., web-based TFL Viewer) can render outputs directly from ARD, enabling a unified environment for review and collaboration. Reviewers can provide inline comments, track changes through version control, and maintain audit trails, all within the same system. This significantly reduces review cycle time and improves transparency compared to traditional document-based workflows.

A key advantage of this approach is the reuse of ARD across downstream processes, particularly CSR development. The same ARD can be used to generate:

- In-text tables within CSR narratives
- Post-text tables from rendered outputs
- Consistent linkage between tables, figures, and narrative descriptions

Because ARD contains both results and contextual metadata, it eliminates manual transcription and ensures alignment between statistical outputs and reported content.

Finally, the structured nature of ARD enables AI-driven workflows, including automated validation, consistency checks, and narrative generation. By using ARD as the central data layer, organizations can move toward a fully integrated, data-driven reporting ecosystem that supports both automation and advanced analytics.

8. CSR Integration and AI-Enabled Reporting

Structured ARD and ARS metadata enable direct integration with CSR development. AI models can consume structured results to generate narratives, validate consistency, and identify key insights.

This transforms CSR development from a manual, interpretation-heavy process into a data-driven and AI-assisted workflow, improving efficiency, consistency, and accuracy.

The combined impact of these capabilities results in measurable improvements across the clinical reporting lifecycle.

9. Benefits, Implementation, and Future Directions

This framework delivers measurable improvements in traceability, efficiency, consistency, and scalability across the clinical reporting lifecycle.

By establishing structured metadata as the single source of truth, organizations can reduce discrepancies between SAP definitions, programming logic, and outputs. Automation reduces manual effort, while standardized definitions improve consistency.

Importantly, this framework enables AI-driven clinical reporting by providing structured, machine-readable inputs that support validation, automation, and narrative generation.

Organizations can adopt this approach incrementally, starting with shell automation and expanding to full metadata-driven workflows. Future developments include AI-driven CSR generation, automated quality control, and expansion of AC/DC libraries.

These benefits can be further illustrated through a practical comparison of traditional, metadata-driven, and AI-enabled workflows.

10. Practical Observations Across Workflows

Based on practical experience across multiple studies and early implementations of metadata-driven tools, a clear progression is observed when moving from traditional workflows to metadata-driven and AI-enabled approaches.

In traditional workflows, a significant portion of effort is spent on manual interpretation of SAP content and setup activities, leading to variability, multiple review cycles, and increased quality control effort.

With the adoption of structured metadata (AC/DC, ARS), workflows become more standardized and efficient. Automated shell generation, metadata-driven programming, and centralized review reduce manual effort and improve consistency across outputs.

Early implementations of AI-enabled workflows, leveraging structured ARS and ARD, further demonstrate potential in automating validation, improving consistency, and supporting CSR narrative generation.

The metrics below represent initial efficiency gains observed in early implementations and pilot use cases. Actual results may vary based on study complexity and organizational maturity.

Area	Traditional Workflow	Metadata-Driven Workflow	AI-Enabled Workflow
Programming Effort	High manual setup and interpretation (~30–40%)	Reduced through standardized logic and metadata (~30–50% improvement)	Further reduced with AI-assisted code generation
Shell Development	Manual, time-consuming	Automated, template-driven (~50–80% reduction)	Fully automated and reusable
Review Cycles	Multiple iterations (2–3 cycles typical)	Streamlined (often 1–2 cycles)	Faster cycles with AI-assisted validation
QC Effort	High, focused on reconciliation	Reduced due to improved consistency	Further reduced via automated checks
CSR Development	Manual interpretation of outputs	Partially streamlined using structured outputs	AI-assisted generation (~60–80% reduction for standard sections)
Consistency Across Outputs	Variable, dependent on programmer	Improved through standardization	High consistency enforced through structured metadata and AI
AI Readiness	Limited (unstructured inputs)	Foundational (structured metadata)	Fully enabled

11. Conclusion

A metadata-driven TFL workflow transforms clinical reporting by connecting SAP, analysis, and CSR processes into a unified pipeline. By leveraging AC, DC, ARS, and ARD, organizations can improve traceability, reduce manual effort, enhance consistency, and enable AI-driven automation. This approach provides a scalable, future-ready foundation for transforming clinical reporting into a data-driven, automation-enabled ecosystem.

12. References

1. Busa, Bhavin; Harris, Brian. *Analysis Concepts Definition.* CDISC U.S. Interchange Oct 2025. Presentation Link
2. CDISC ARS Model: https://cdisc-org.github.io/analysis-results-standard/
3. CDISC ARS User Guide version 1.0: https://wiki.cdisc.org/display/ARSP/Analysis+Results+Standard+User+Guide+v1.0
4. eTFL Portal: https://cdisc.org/kb/etfl
5. Busa, Bhavin; LeRoy, Bess; 'The Development of the CDISC eTFL Portal to Support the Adoption and Implementation of the Analysis Result Standard' at PHUSE EU Connect (Nov 2024). Paper Link & Presentation Link
6. Busa, Bhavin; Marshall, Richard; LeRoy, Bess; 'All You Need to Know about the New CDISC Analysis Result Standards!' PharmaSUG, May 2023.
7. Bosman, Malan. '*Advancing TFL Automation: R-Based Meta-Programming Powered by ARS*' PHUSE US Connect, March 2025.
8. Wachara, Julie. '*Let the Machine Do the Coding: A SAS-Based Framework for Automating TFL*

Generation with ARS Metadata' PHUSE US Connect, March 2025.

9. *Dedhia Navin; Busa Bhavin. '*AI Code Generator: Automating TFL Programming with ARS Metadata*'* PHUSE US Connect, March 2026.

Contact Information

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

Bhavin Busa
Principal & Co-founder,
Clymb Clinical
bhavin@clymbclinical.com