PharmaSUG 2026 - Paper DS-354

SUPP to NSV: Transforming Data Representation for Improved Reviewer Utility

Soumya Rajesh, IQVIA

ABSTRACT

Ever since the advent of SDTM standards, non-standard data has been mapped using the vertical structure outlined in Supplemental Qualifier (SuppQual or Supp) datasets. This has caused issues sometimes when the data needs to be merged back with parent domains and used for analysis further down the data reporting process. That's going to change majorly with the release of the new version of SDTM: Model v3.0 and IG v4.0. This paper describes the transformation from vertical to horizontal structure for mapping non-standard datasets to the newer, more efficient Non-Standard Variables (NSV) datasets in SDTM. This would enable the direct integration of non-standard data into General Observation class domains for easier viewing and metadata application, thereby streamlining submissions by reducing structural limitations and promoting consistency within SDTM standards.

INTRODUCTION

SUPPQUAL datasets have served as the long-standing mechanism for representing non-standard data in SDTM. Using QNAM (variable name), QLABEL (label), and QVAL (value), SUPP datasets store non-standard variables in a vertical structure that requires transposition or pivoting to merge with parent domains. Although this design initially offered flexibility, limitations have accumulated as data complexity and regulatory expectations have evolved.

The shift from SUPPQUAL to NSV in SDTM v3.0 and SDTMIG v4.0 is transformative. Rather than storing one row per non-standard variable, NS-- datasets store one row per parent observation, with each non-standard variable expressed as a true variable (column). This enables consistent metadata, improved readability, simpler merging, and more efficient review. It also resolves long standing pain points such as QVAL always being character, difficulty applying controlled terminology, and inconsistent value level metadata.

This paper presents a consolidated overview of how NSVs function, why they improve the SDTM data model, and how implementers can incorporate them successfully.

BACKGROUND: LIMITATIONS OF SUPPQUAL

The SUPPQUAL structure, despite serving the SDTM ecosystem for years, has had significant drawbacks:

Structural Issues

- One record per QNAM/QVAL pair results in redundancy and large datasets.
- Transposing is required to merge SUPPQUAL back to parent domains.
- Merging logic varies depending on IDVAR/IDVARVAL combinations, and even then it requires conversion of IDVARVAL (due to it being character) to merge with a numeric --SEQ.
- QVAL is always character - even when storing numeric values.

Metadata Limitations

- SUPPQUAL relies on value-level metadata in Define-XML.
- Variable-level attributes such as length, type, codelist, and origin cannot be uniquely assigned.

Reviewer Burden

- Reviewers must manually reassemble data to understand all variables associated with the parent domain.
- Data with many supplemental qualifiers becomes hard to interpret and increases review time.

Inconsistent Implementation

- SUPPQUAL allowed different join keys (IDVAR = --SEQ, --TESTCD, --GRPID, etc.), which forced programmers to create multiple merge conditions.
- Validation inconsistencies were common due to ambiguous linkages.

RATIONALE FOR TRANSITION TO NSVS

The move to NSVs addresses each of the SUPPQUAL limitations:

- **Horizontal alignment** enhances readability and interpretability.
- **Variable-level metadata** improves clarity and supports automation.
- **Consistent merge logic** based on --SEQ simplifies programming.
- Can also **reduce the number of records** needed, as the structure matches parent dataset structure (versus vertical SUPP).
- **Regulatory agencies** expressed a preference for simpler, more transparent data structures.
- **Machine-readable metadata initiatives** in SDTMIG v4.0 require a clearer metadata model that SUPPQUAL cannot support.

As CDISC modernizes the SDTM standard, NSVs provide a foundation for greater interoperability across tools, improved visualization, and enhanced regulatory clarity.

SUPP→NSV TRANSFORMATION

The transition from Supplemental Qualifier datasets to Non-Standard Variable datasets represents a significant structural overhaul in SDTM history. The SUPP→NSV transformation replaces the original vertical, QNAM/QVAL-based representation with a cleaner, horizontal structure that aligns directly with the parent domain. Where SUPPQUAL captured one non-standard variable per record, NS-- datasets capture all applicable non-standard variables for a parent record in a single observation, resulting in simpler merges, easier review, and more robust metadata control.

Vertical vs. Horizontal: Structural Differences

Under the SUPPQUAL structure, non-standard variables were stored in a vertical format-one row for each variable, requiring the use of IDVAR and IDVARVAL to relate back to a parent record. Because QVAL always stored values as character strings, numeric data types were lost and had to be converted later. Merging SUPP datasets back to their parent domains requires transposing data and creating multiple join paths.

In contrast, NS-- datasets follow a horizontal design: each record contains the full set of non-standard variables applicable to the corresponding parent record (when there are NSV variable values, else records can be dropped if containing no NSV variable values across the NS record).

This simplifies merging, enhances readability, and supports variable-level metadata in Define-XML. Data types remain intact, and controlled terminology can be applied per variable rather than being constrained to value-level metadata.

Benefits of the NSV Structure

The new model offers several advantages:

- **Cleaner integration** with parent domains using standard keys.
- **Enhanced metadata handling**, including variable type, length, codelists, and origin.
- **Improved usability**, allowing reviewers to view standard and non-standard variables side by side.
- **Reduced dataset size**, as multiple NSVs share a single parent-aligned record.

- **Better consistency**, since join logic uses a single key variable rather than multiple approaches.

These benefits reflect feedback from CDISC leadership, regulatory agency representatives, and CDISC Advisory Council members, who consistently emphasized the need for a simpler, more transparent representation of non-standard data.

THE NSV METADATA SPECIFICATION TABLE

#	Variable Name	Variable Label	Type	Codelist	Allowed Controlled Terms	Format	Role	Variable Group	Root Variable C-code	Root Variable Definition	Notes	Examples	Core
1	STUDYID	Study Identifier	Char				Identifier	Study	C83082	A sequence of characters used to identify or name the study.			Req
2	RDOMAIN	Related Domain Abbreviation	Char	(DOMAIN)			Identifier	Pointer	C83391	A variable that represents the domain abbreviation for the domain in which related records are located.			Req
3	USUBJID	Unique Subject Identifier	Char				Identifier	Observation Subject	C69256	A sequence of characters used to uniquely identify a subject across all studies for all applications or submissions involving the product.	See Use of "Subject" and USUBJID and SUBJID.		Exp
4	SUBJID	Subject Identifier for the Participation	Char				Identifier	Observation Subject		A sequence of characters used to uniquely identify a subject's participation in a study.	See USUBJID and SUBJID.		Perm
5	APID	Associated Persons Identifier	Char				Identifier	Observation Subject		A sequence of characters used to uniquely identify a single associated person, a group of associated persons, or a pool of associated persons.			Perm
6	POOLID	Pool Identifier	Char				Identifier	Observation Subject	C117053	A sequence of characters used to uniquely identify a group of subjects that have been pooled together.			Perm
7	SPDEVID	Sponsor Device Identifier	Char				Identifier	Observation Subject	C117060	A sequence of characters used by the sponsor to uniquely identify a specific device.	See Performers and Evaluators.		Perm
8	IDVAR	Identifying Variable	Char				Identifier	Pointer		The name of a variable, used in combination with other variables, to identify related records.	See NS Assumption 4.	"--SEQ"	Exp
9	IDVARVLN	Identifying Variable Numeric Value	Num				Identifier	Pointer		The numeric value of the variable named in IDVAR used to identify related records.	See NS Assumption 4.		Exp

NSV IMPLEMENTATION RULES AND ASSUMPTIONS

CDISC defines a comprehensive set of rules governing NS-- dataset construction in SDTM v3.0 / SDTMIG v4.0. The following section integrates these rules into a coherent narrative, ensuring implementers understand how NSVs are generated, structured, linked, and validated.

1. **Population and Submission Requirements**
   NS-- datasets must include at least one non-standard variable populated with a value for each record. If an NSV contains only null values across all records, it must be omitted from the dataset entirely. Similarly, if all potential NSVs for a domain are null (i.e., no non-standard data were collected), the entire NS-- dataset should be omitted from submission.

   However, the planned NS-- dataset and its variables must still appear in the study metadata (Define XML), annotated as "no data collected." This ensures full transparency regarding planned, but unpopulated domain extensions.

2. **Record-Level Linkage to Parent Domains**

   Each NS-- record must relate back to a single, unique parent domain record. CDISC defines the complete key for this relationship as:

   - **STUDYID**
   - **RDOMAIN**
   - **USUBJID** (or **POOLID**, **SPDEVID**, **APID**, depending on context)
   - **IDVAR**
   - **IDVARVLN**

   This linkage ensures a one-to-one correspondence between the NS-- record and the parent record. Unlike SUPPQUAL datasets, where IDVAR could represent different identifying fields across variables, NS-- datasets enforce a consistent, predictable structure.

3. Identifier Requirements by Parent Record Type

Depending on the type of entity represented by the parent domain record, the appropriate identifier must be included in the NS-- record:

a. Study subject observations

If the parent record describes a study subject, **USUBJID** must be populated.

b. Pooled subject observations

If observations pertain to a pool of subjects (POOLID), then **POOLID** must be included and populated.

c. Device observations

If the parent record concerns a device, then **SPDEVID** must be included.

d. Associated person observations

For associated person data, **APID** must be included, and the NSAP-- dataset must be used.

This ensures flexibility in representing non-standard variables for different observational entities, while maintaining consistent linking logic and metadata clarity.

4. IDVAR Requirements

For most domains, CDISC mandates that **IDVAR must equal --SEQ**, since --SEQ is the universal sequence variable used to uniquely identify each parent observation. This standardizes the join logic and eliminates long-standing confusion present in SUPPQUAL datasets, where implementers could choose between --SEQ, --GRPID, --CAT, or other identifiers.

Exception - NSDM

The Demographics domain does not use --SEQ, therefore in **NSDM**, both *IDVAR* and *IDVARVLN* must be null. The parent record is uniquely identified solely by:

- STUDYID
- RDOMAIN
- USUBJID

This exception reflects the structural nature of DM, which contains only one record per subject.

5. One-to-One Correspondence Between Domain and NS-- Dataset

Every SDTM dataset in the general observation classes (AE, LB, VS, etc.) may have at most one NS-- dataset associated with it. This means:

- AE → NSAE
- LB → NSLB
- VS → NSVS

When a domain is split (e.g., FAMH for family history), the NS datset mirrors that split (NSFAMH).
For associated persons, NSV datasets follow the specification for the Associated Persons IG.

This rule eliminates the ambiguity of SUPPQUAL, where one domain could have multiple supplemental datasets representing inconsistent sets of variables.

6. Handling Text Longer than 200 Characters

NSVs follow the same rules that apply to general observation-class variables regarding text strings exceeding 200 characters. Per SDTMIG Section 4.3.6, sponsors must:

- Represent long text using permitted splitting conventions.

- Ensure variable names and labels follow NSV naming rules.
- Avoid forcing excessively long content into a single NSV field if it violates dataset format constraints.

  This keeps NSV handling consistent with SDTM-wide rules for long text.

7. Multiple Values for a Non-Result Qualifier

Certain qualifiers may legitimately contain multiple values (e.g., multiple reasons, multiple conditions, multiple contributing factors). When this occurs, Rule 7 requires the use of the **multiple-values rule**, which determines:

- Whether to create multiple NSVs
- Whether to split values into separate fields
- Whether to create multiple records when allowed

This rule ensures that multi-valued non-result qualifiers are consistently represented across sponsors and studies.

8. Naming Conventions for NSVs

All Non-Standard Variables must adhere to the naming conventions outlined in SDTMIG Section 4.2.3. This rule ensures that NSVs follow the same structural, syntactic, and semantic constraints as standard SDTM variables. NSV names must:

- Begin with a valid domain prefix.
- Follow the allowable character sets and length limits.
- Maintain alignment between variable names and labels.
- Avoid ambiguity with standard SDTM variables.

These naming conventions support machine-readability and prevent naming collisions that could cause downstream interpretation issues.

9. Controlled Terminology

CDISC provides controlled terminology for commonly used NSVs, cataloged in Appendix C1. Where applicable, implementers should use these approved names and corresponding labels to ensure consistency across studies. This reduces variability in metadata, improves traceability, and aligns with regulatory expectations for harmonized terminology.

If an NSV does not have a corresponding code in Appendix C1, sponsors must still name and label the variable according to NSV naming rules to maintain clarity and consistency.

10. Metadata Requirements in Define-XML

All NSVs must be fully represented in Define-XML using **variable-level metadata**, replacing the value-level metadata historically used for SUPPQUAL variables. Required metadata include:

- Variable Name
- Label
- Data Type
- Length
- Controlled Terminology or Codelist
- Origin (“Collected”, “Derived”, “Assigned”, “Protocol”, etc. and “Investigator”, “Sponsor”, “Vendor”)
- Comments or computational methods (if applicable)
- This structure enables clearer traceability, simplifies reviewer interpretation, and supports automated validation systems.

11. Allowed Roles for NSVs

Each NSV must take one of the following roles, reflecting the roles allowed for standard SDTM variables:

- Non-standard Identifier
- Non-standard Timing
- Non-standard Qualifier
- Non-standard Rule

Importantly, **NSVs cannot have the "Non-standard Topic" role**, because topics would redefine a domain's conceptual structure. NSVs are intended to extend domains-not alter their foundational meaning.

12. Data Not Appropriate for NSVs

Certain data types should **not** be represented as NSVs, even if non-standard:

a. Subject-level objective characteristics: For example: National origin, Twin type, etc. Such data belong in the **Subject Characteristics (SC)** domain.

b. Interpretations of findings: For example: ECG interpretation values that should instead be expressed as a finding with its own TESTCD.

c. Comments and free-text annotations: Comments must be stored in the **CO (Comments)** domain-not as NSVs.

d. Data not directly linked to a parent domain record: If there isn't a 1:1 link with a parent, the information must be placed in a separate general observation-class or special-purpose domain.

These exclusions ensure NSV usage remains focused, structured, and consistent across the SDTM ecosystem.

ASSUMPTIONS FOR NSV CONSTRUCTION

In addition to the formal implementation rules, CDISC provides assumptions and guiding principles to ensure NSVs are used consistently and effectively across sponsors:

Assumption 1 - One NS-- Record per Parent Record

There must never be an NS-- record that links to multiple parent records. This preserves the simplicity of the one to one mapping model.

Assumption 2 - Only Non Null NSVs Are Represented

NSVs that have no collected or derived values must not appear in the dataset. This keeps the NS-- dataset clean, concise, and reviewer friendly.

Assumption 3 - NSV Datasets Follow the Same Structural Ordering as Parent Datasets

Records within NS-- datasets must follow the same ordering rules-typically sorted by key variables-to support intuitive cross comparison.

Assumption 4 - Variables Must Conform to SDTM Typing Rules

If the NSV is numeric, it must be defined as numeric. If character, appropriate lengths must be assigned. QVAL style ambiguity is no longer permissible.

Assumption 5 - Consistency Across Split Domains

When a parent domain is split into multiple datasets (e.g., VSFAST, VSFULL), the NS-- dataset must follow the split pattern.

Assumption 6 - Metadata Must Always Reflect the Actual Content

Metadata must fully describe what is present, what is omitted, and what values are or are not collected.

These assumptions serve as foundational guidance for consistent implementation.

EXAMPLES AND PRACTICAL APPLICATION:

These examples illustrate how the new model resolves long standing challenges, improves clarity, and enhances the end to end review process.

Example 1: SUPPAE vs NSAE

ae.xpt Convert IDVARVAL from character to numeric

Row	STUDYID	DOMAIN	USUBJID	AESEQ	AETERM	AESEV	AESER	AESMIE	AESTDTC	AEENDTC
1	1996001	AE	99-401	1	UTERINE FIBROIDS	SEVERE	Y	Y	2023-01-05	2023-01-12
2	1996001	AE	99-567	1	FEVER	MILD	N		2023-09-25	2023-09-25

suppae.xpt Transpose

Row	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	QNAM	QLABEL	QVAL	QORIG	QEVAL
1	1996001	AE	99-401	AESEQ	1	AESOSP	Other Medically Important SAE	SPONTANEOUS ABORTION	CRF	
2	1996001	AE	99-401	AESEQ	1	AETRTEM	Treatment Emergent Flag	Y	Derived	
3	1996001	AE	99-567	AESEQ	1	AETRTEM	Treatment Emergent Flag	N	Derived	

nsae.xpt

Row	STUDYID	DOMAIN	USUBJID	IDVAR	IDVARVLN	AESOSP	AETRTEM
1	1996001	AE	99-401	AESEQ	1	SPONTANEOUS ABORTION	Y
2	1996001	AE	99-567	AESEQ	1		N

NSAE metadata

Variable	Label	Type	Codelist	Role	Origin Type	Origin Source
AESOSP	Other Medically Important SAE	Char		Non-Standard Qualifier	Collected	Investigator
AETRTEM	Treatment Emergent Flag	Char	(NY)	Non-Standard Qualifier	Derived	Sponsor

Limitations of SUPPAE:

- Multiple rows per parent record
- QVAL forces all values to be character
- No variable level metadata
- Requires transposing before merging

Benefits of NSAE:

- One record per AE observation
- AESOSP and AETRTEM are true variables
- No transposition required
- Metadata clearly defined at the variable level.

Example 2: SUPPDM vs NSDM

Row 3 in DM shows that USUBJID ABC789-010-047 designated multiple races as the values that best describe their race. "MULTIPLE" is assigned in RACE.

dm.xpt

Row	STUDYID	DOMAIN	USUBJID	SUBJID	RACE
1	ABC789	DM	ABC789-010-045	010-045	WHITE
2	ABC789	DM	ABC789-010-046	010-046	ASIAN
3	ABC789	DM	ABC789-010-047	010-047	MULTIPLE

SUPPDM represents multiple values in race vertically in different rows, when a subject selects “MULTIPLE” in DM.RACE:

suppdm.xpt

Row	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVAL	QNAM	QLABEL	QVAL	QORIG	QEVAL
1	ABC789	DM	ABC789-010-047			RACE2	Race 2	ASIAN	CRF	
2	ABC789	DM	ABC789-010-047			RACE5	Race 5	WHITE	CRF	

NSDM would represent this in one row as only one record in DM is being qualified with NSVs. The record in the NSDM dataset adds qualifying information to demographics data (RDOMAIN = "DM"). USUBJID is the key variable used to link NSDM to DM.

The value in USUBJID identifies the parent DM record to which the NSDM record applies. IDVAR and IDVARVLN are both null in NSDM. The remaining columns specify the non-standard variable names and values.

nsdm.xpt

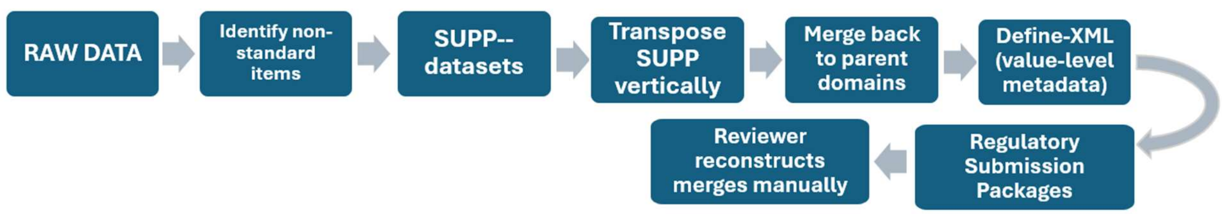
Row	STUDYID	RDOMAIN	USUBJID	IDVAR	IDVARVLN	RACE2	RACE5
1	ABC789	DM	ABC789-010-147			ASIAN	WHITE

NSDM Metadata:

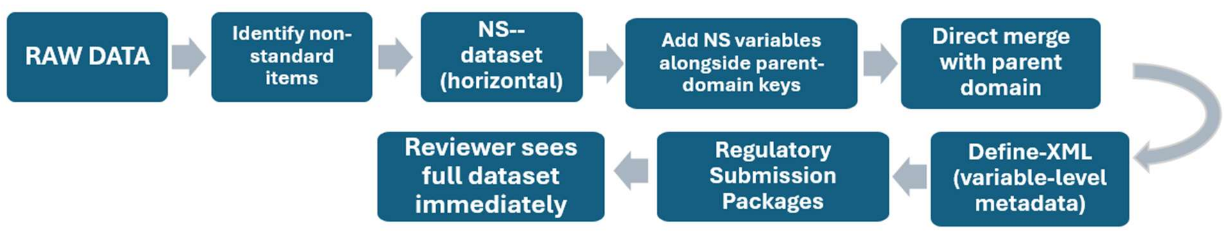
Variable	Label	Type	Codelist	Role
RACE2	Race 2	Char	(RACE)	Non-Standard Record Qualifier
RACE5	Race 5	Char	(RACE)	Non-Standard Record Qualifier

WORKFLOW COMPARISON

SUPPQUAL Workflow



NS-Workflow



Structural Differences:

Feature	SUPPQUAL (SUPP--)	NSV (NS--)
Structure	Vertical (one row per non-standard variable)	Horizontal (one row per parent record)
Metadata	Value-level only	Variable-level (codelists, length, type, origin)
Merge Complexity	Requires transpose + join	Direct join via STUDYID/RDOMAIN/USUBJID/--SEQ
Data Types	QVAL = character always	True numeric/character typing
Review Burden	High (multiple files, reassembly needed)	Low (directly readable)
Alignment	Deprecated in SDTM v3.0	Required in SDTM v3.0

Technical Comparison

SUPP-- Merge Logic (Old)

```
proc transpose data=suppae out=work.ae_supp;
  by usubjid idvar idvarval;
  id qnam;
  var qval;
run;
data ae_merged;
  merge ae ae_supp;
  by usubjid aeseq;
run;
```

NS-- Merge Logic (New)

```
data ae_merged;
  merge ae nsae;
  by usubjid aeseq;
run;
```

PRACTICAL CONSIDERATIONS FOR SPONSORS:

Sponsors adopting NSVs for the first time should consider:

- Updating macro libraries to support NS-- creation
- Validating all IDVAR and IDVARVLN linkages
- Ensuring Define-XML metadata is complete
- Updating reviewer-friendly outputs (e.g., SDRG notes)
- Training statisticians and reviewers on new structures

ADDITIONAL CLARIFICATION/ DISCUSSION

The transition to NSVs significantly simplifies both programming and review. Because NSVs follow the same record structure as parent domains, merging, browsing, and analysis become far more intuitive. Metadata improvements introduced in SDTMIG v4.0 amplify these benefits by enabling consistent representation, controlled terminology enforcement, and clear domain assumptions.

Furthermore, reviewers benefit from having a complete representation of variables within a single dataset structure-aligning with regulatory expectations for clarity and ease of use. CDISC identifies this shift as a major modernization effort intended to support increased automation and machine readability in future submissions.

CONCLUSION

By transitioning from a vertical QNAM/QVAL model to a horizontal, variable centric design, NSVs resolve long standing SUPPQUAL limitations, including:

- Forced character typing via QVAL
- Value level metadata constraints
- Ambiguous or inconsistent join logic
- Burdensome reviewer reconstruction

- Difficulty applying controlled terminology

The updated NS-- model also integrates with SDTMIG v4.0's **Metadata Restructuring** initiative, which reorganizes SDTM variable specifications for machine readability, clarity, and standardization.

This positions NSVs not only as a technical improvement, but as an essential component of CDISC's broader modernization roadmap.

The adoption of NS-- datasets yields immediate benefits:

- **Cleaner, more intuitive data structures**
- **Better metadata quality and interpretability**
- **Reduced programming complexity**
- **Improved reviewer efficiency**
- **Alignment with modern regulatory expectations**

REFERENCES

SDTM v3.0 and SDTMIG v4.0

SDTMIG v4.0 Home - SDTMIG v4.0 - Wiki

SDTM | CDISC

SDTMIG | CDISC

Non-Standard Variables Registry | CDISC

ACKNOWLEDGMENTS

Special thanks to my esteemed colleague and co-lead of the 'SUPP to NSV' sub-team, **David Neubauer**, Senior Manager, Clinical Data Assets, Repositories, & Standards, AbbVie.

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

Soumya Rajesh, Sr. Standards Engineer, Global Data Standards, IQVIA
Soumya.rajesh@iqvia.com
linkedin.com/in/soumya-rajesh-2b7a074