

APAC 2026

SAS to R: Opportunities and Challenges in SDTM and ADaM Transformation

Jalendhar Etedi, Padmashree Jahagirdar
Atorus Research



Why SAS to R?

Advantages:

- Open-source, no licensing cost
- Strong data manipulation capabilities (dplyr, tidyverse)
- Metadata-driven programming (metacore, {xportr})
- Pharmaverse ecosystem supports SDTM & ADaM

Challenges:

- SAS macro heavy workflows → R functional style
- Need for validation frameworks
- Sponsor / regulatory confidence building



R - Core Packages

Envsetup: Utility package for standardizing project folder structures and initializing consistent R environments.

Dplyr: Grammar of data manipulation providing fast, readable functions for filtering, joining, summarizing, and transforming datasets.

Metacore: Core metadata framework for storing, structuring, and retrieving CDISC study specifications (SDTM and ADaM) to drive standardized, metadata-driven clinical programming workflows.

Metatools: Tools for applying metadata consistently - variable attributes, lengths, and SUPPQUAL processing in SDTM workflows.

Tidyr: Provides functions to reshape data (e.g., pivoting) to ensure tidy, analysis-ready structures.



R - Core Packages

Admiral: Validated pharmaverse package offering standardized, reusable ADaM derivation functions.

Lubridate: Simplifies parsing, manipulating, and calculating dates/times in a consistent and human-friendly way.

Xportr: Ensures CDISC compliance by applying metadata, variable order, types, labels, and exporting FDA-ready XPT files.

Readxl: Reads Excel files (.xlsx/.xls) into R without requiring external dependencies.

Diffdf: Compares two data frames and produces clear, audit-ready reports of differences - essential for QC.



SDTM in R: What Changes?

SAS Workflow

- DATA step + PROC SQL
- Macros to support repetitive derivations
- Explicit metadata assignments

R Workflow

- tidyverse pipelines
- {metacore} for metadata structure
- {xportr} to apply labels, types, CT
- {lubridate} for date manipulations



SDTM: Key Derivations

DM Example: Deriving RFSTDTC & RFENDTC

R Code

```
dm <- ex %>%
```

```
  group_by(USUBJID) %>%
```

```
  summarise(RFSTDTC = min(EXSTDTC, na.rm = TRUE), RFENDTC = max(EXENDTC, na.rm = TRUE) )
```

EX

USUBJID	EXSTDTC	EXENDTC
SUB001	2025-01-05	2025-01-05
SUB001	NA	2025-01-10
SUB001	2025-01-15	NA

DM

USUBJID	RFSTDTC	RFENDTC
SUB001	2025-01-05	2025-01-10

Advantages

- Grouping/summarization is clearer
- Simplified and readable transformation workflow



SDTM: Key Derivations

AE Domain Derivations

Typical Derivations:

- **AESTDY / AEENDY** – Study day
- **TRTEMFL** – Treatment-emergent flag

Source Dependencies:

- DM for RFSTDTC/RFXSTDTC



SDTM: Key Derivations

AE Derivation Example: TRTEMFL

Definition: AE occurs on or after first study treatment.

R Code

```
ae <- ae %>%
```

```
left_join(dm %>% select(USUBJID, RFXSTDTC)) %>%
```

```
mutate(TRTEMFL = if_else(AESTDTC >= RFXSTDTC, "Y", "N"))
```

USUBJID	AESTDTC	AEDECOD	RFXSTDTC	TRTEMFL
SUB001	2022-01-20	Headache	2022-01-04	Y
SUB001	2021-12-30	Nausea	2022-01-04	N

Advantages: R (with dplyr) expresses the Clear, vectorized date comparison in a single pipeline



SDTM: Key Derivations

LB Domain Derivations

Typical Derivations:

- **LBDY** – Study day
- Baseline flag (LBBOXFL)

Source Dependencies:

- DM for RFSTDTC/RFXSTDTC
- Multiple LB records per visit



SDTM: Key Derivations

LB Example: Study day

Definition: Relative day from the reference start date.

R Code

```
lb <- lb_raw %>%  
  derive_vars_dy(  
    reference_date = RFSTDT, source_vars = exprs(LBDT) )
```

USUBJID	LBTESTCD	RFSTDT	LBDT	LBDY
SUB001	ALB	2023-05-31	2023-04-18	-43
SUB001	ALB	2023-05-31	2023-06-01	2
SUB001	ALB	2023-05-31	2023-06-05	6

Advantages

- With `admiral::derive_vars_dy()`, R reliably calculates study day values (e.g., --DY) relative to a reference date through a standardized, validated, and reusable approach.



SDTM: Key Derivations

SUPPQUAL Handling:

R

```
suppdm <- bind_rows(  
  build_qnam(dm, "RACEOR1", "Original Race 1", "", "", "CRF"),  
  build_qnam(dm, "RACEOR2", "Original Race 2", "", "", "CRF"))
```

USUBJID	RDOMAIN	IDVAR	IDVARVAL	QNAM	QLABEL	QVAL	QORIG	QEVAL
SUB001	DM			RACEOR1	Original Race 1	EUROPEAN	CRF	
SUB001	DM			RACEOR2	Original Race 2	WHITE	CRF	

Advantages

- With `metatools::build_qnam()`, R simplifies SUPPQUAL creation by automating variable pivoting, record linkage, and QNAM/QLABEL assignment through a reusable function.



ADaM Overview

Key ADaMs Covered:

- **ADSL** – subject-level
- **ADAE** – event-level
- **ADLB(BDS)** – lab analysis datasets
- **ADTTE** – time-to-event

R Tools

- {admiral} for derivations
- {dplyr} for dataset creation
- {lubridate} for date logic



ADaM: Key Derivations

ADSL: TRTSDT / TRTEDT

R Example (admiral)

```
derive_vars_dt(  
  new_vars_prefix = "TRTS", dtc = RFXSTDTC) %>%
```

```
derive_vars_dt(  
  new_vars_prefix = "TRTE", dtc = RFXENDTC) %>%
```

USUBJID	TRTSDT	TRTEDT	RFXSTDTC	RFXENDTC
SUB001	2023-05-31	2023-05-31	2023-05-31T11:43	2023-05-31T11:53

Advantages

- Admiral abstracts common logic
- Consistency across studies



ADaM: Key Derivations

ADAE: On-Treatment Flag (AE occurs during the treatment period)

```
adae <- ae %>% derive_var_ontrtfl( dataset      = ae,  
  start_date   = ASTDT, end_date   = AENDT,  
  ref_start_date = TRTSDT, ref_end_date = TRTEDT )
```

USUBJID	ASTDT	AENDT	TRTSDT	TRTEDT	ONTRTFL
SUB001	2023-09-21	NA	2023-09-21	2023-09-21	Y
SUB001	2024-06-03	2024-06-03	2024-06-03	2024-06-03	Y

Advantages

Using `admiral::derive_var_ontrtfl()`, R provides a standardized, validated, and reusable function that automatically handles treatment dates and edge cases with minimal code.



ADaM: Key Derivations

ADAE/ADCM: Partial date imputation

Start Date Derivation

```
derive_vars_dt("AST", AESTDTC, highest_imputation = "Y", min_dates = exprs(TRTSDT))
```

End Date Derivation

```
derive_vars_dt("AEN", AEENDTC, highest_imputation = "M", date_imputation = "last")
```

USUBJID	ASTDT	AESTDTC	AEENDTC	AENDT	TRTSDT	TRTEDT	ASTDTF	AENDTF
SUB001	2023-10-14	2023-10-14	2024-05	2024-05-31	2023-09-21	2023-09-21		D
SUB001	2023-10-20	2023-10-20	2023-12	2023-12-31	2023-09-21	2023-09-21		D

Advantages

With `admiral::derive_vars_dt()`, R performs standardized date conversion and controlled imputation (day/month/year flags) in a single validated call.



ADaM: Key Derivations

ADLB: Baseline Flag (Last non-missing assessment prior to treatment start)

```
restrict_derivation(derivation = derive_var_extreme_flag, args = params( by_vars = exprs(STUDYID,
  USUBJID, PARAMCD), order = exprs(ADT, ADTM), new_var = ABLFL, mode = "last" ), filter = ((
  AVALC != "" | !is.na(AVAL)) & ((!is.na(ADTM) & ADTM < TRTSDTM) | (is.na(ADTM) & !is.na(ADT) &
  ADT <= TRTSDT))))
```

BASE

```
adlb <- derive_var_base (dataset = adlb_raw, new_var = BASE, by_vars = exprs(USUBJID, PARAMCD))
```

USUBJID	PARAMCD	ABLFL	AVISIT	BASE	AVAL
SUB001	BASO	Y	Baseline	0.05	0.05
SUB001	BASO		Visit 3 - Day 1	0.05	0.03

Advantages

With `admiral::restrict_derivation()`, R automatically and consistently identifies baseline records across parameters using a validated, reusable function, while `derive_var_base()` assigns the BASE value for each subject and parameter.



ADaM: Key Derivations

TTE: Number of days from treatment start to the event date.

```
adtte <- derive_param_tte(  
  dataset_adsl = adsl,  
  start_date = TRTSDT, event_date = DTHDT, event_conditions = DTHFL == "Y")
```

USUBJID	PARAMCD	PARAM	TRTSDT	DTHDT	AVAL	CNSR
SUB001	OS	Overall Survival	2023-01-01	2023-03-15	74	0

Advantages

Using `admiral::derive_param_tte()`, R enables a declarative, end-to-end TTE derivation (start date, event logic, censoring) in a single validated function.



Key Challenges Moving to R

Technical Challenges

- R memory usage differs from SAS
- Complex imputations need custom functions
- R natively supports creation of XPT (transport) files for regulatory submission (via validated packages like xportr), but cannot generate SAS7BDAT files, which require SAS software

Process Challenges

- Validation - unit testing culture
- Need for dual programming in transition
- Regulatory documentation adaptation



Summary

R Strengths

- Clear, expressive data manipulation
- Strong metadata-driven SDTM/ADaM tools
- Open-source, community support
- Highly reproducible pipelines

Challenges

- Need robust validation strategy
- Initial setup effort for infrastructure and reusable libraries

Conclusion

R can deliver production-grade SDTM and ADaM when supported by compliant processes and validated tools.



The Global Healthcare Data Science Community

Contact Channels

📞 UK +44 1843 609600
✉ office@phuse.global
🌐 phuse.global

Social Media

🐦 [@phusetwitta](https://twitter.com/phusetwitta)
📘 [/phusebook](https://facebook.com/phusebook)
▶ [/phusetube](https://youtube.com/phusetube)
🏢 [/company/phuse](https://linkedin.com/company/phuse)