cdisc®

2026 Europe Interchange

THE FUTURE IS CONNECTED: STANDARDS AND AI POWERING DIGITAL TRANSFORMATION

SDTM Oversight: Now with Fewer "Huh?" Moments with S3C

Parag Wani, Senior Director Statistical Program, AstraZeneca
Davide Marinucci, Statistical Programmer II, AstraZeneca

Agenda

1. Our Dream
2. Workflow
3. Dashboard
4. Modular Design
5. AI Agent
6. Result
7. Next Steps

Speakers

Davide Marinucci
Statistical Programmer II
AstraZeneca

Parag Wani
Senior Director Statistical Programming
AstraZeneca

Our Dream

Vision

SDTM Consistency **C**ross-**C**hecker

A future where SDTM validation and oversight is faster, more reliable, and less resource intensive, with minimal reliance on full parallel programming.

Scope

Targeted SDTM Quality Checks

Equip Statistical Programming teams and Study Lead Programmer (SLP) with a streamlined tool for precise spot checks on SDTM datasets.

Validation Enablement

Provide a scalable support capability that enhances the SDTM validation workflow, improving consistency, traceability, and speed.

Hypothesis

Assumptions

- YAML rules capture SDTM checks transparently and reproducibly.
- R can efficiently execute standardized QC logic across studies.

Expected Outcome

- Reduced manual validation effort while preserving expert review for complex, study-specific derivations.

Workflow

Input Data for RAW→SDTM Comparison

Original SDTMs

Used as the reference for comparison and consistency checks.

Raw Datasets

Source data used for RAW→SDTM mapping verification and completeness checks.

Configure Domains & Rules

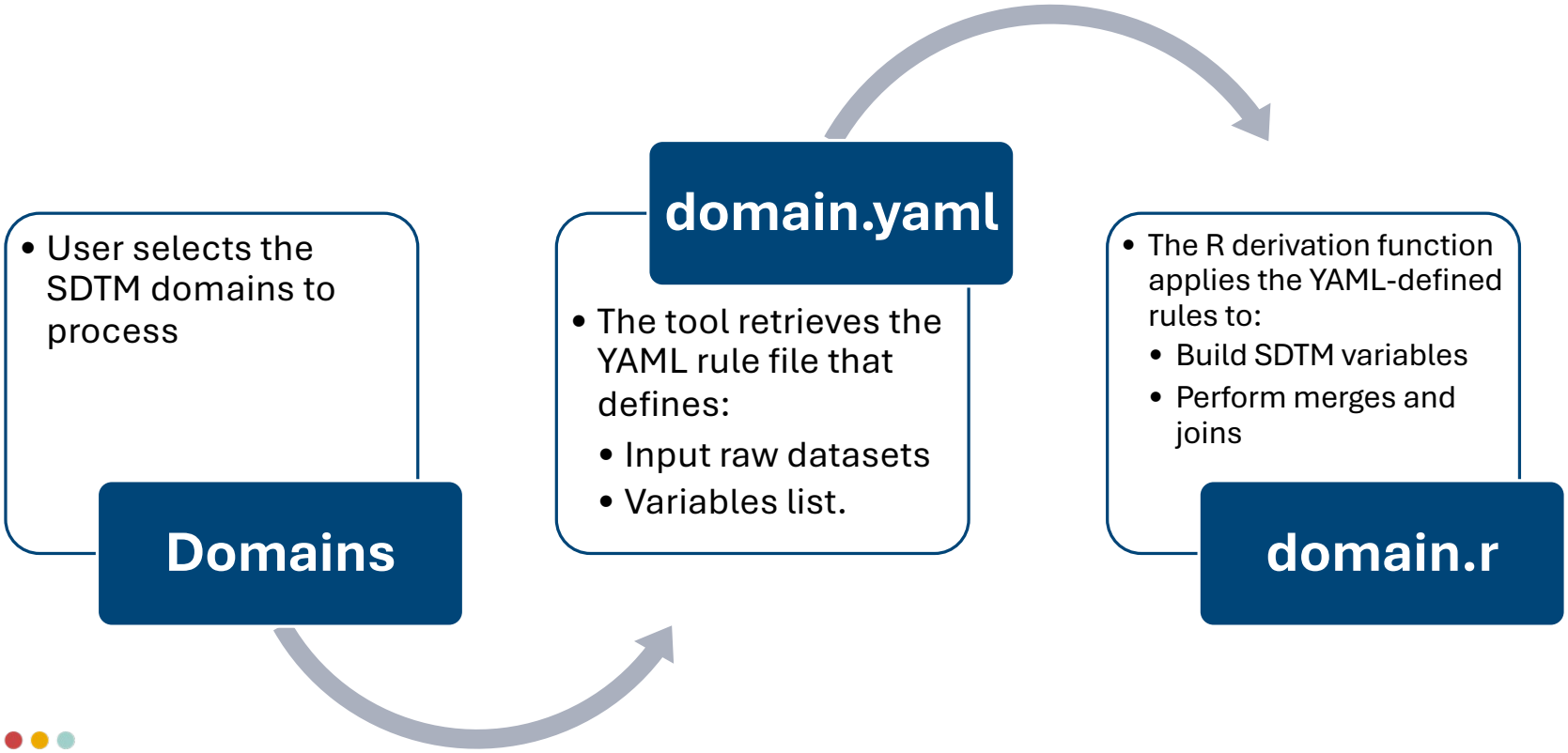
Selection

- Domains
- Variables

Customization

- Sorting Variables

From Raw to SDTM – Domains Level



From Raw to SDTM – Variables Level

S3C Generated vs Original SDTMs

Original SDTMs

S3C SDTMs

RAW→SDTM Comparison

Visualization

- High-level graphical overview of RAW → SDTM differences
- Domain- and variable-level summaries using dynamic plots

Interactive filter

- SQL-style filtering for targeted discrepancy investigation
- Quickly isolate subsets of records or variables

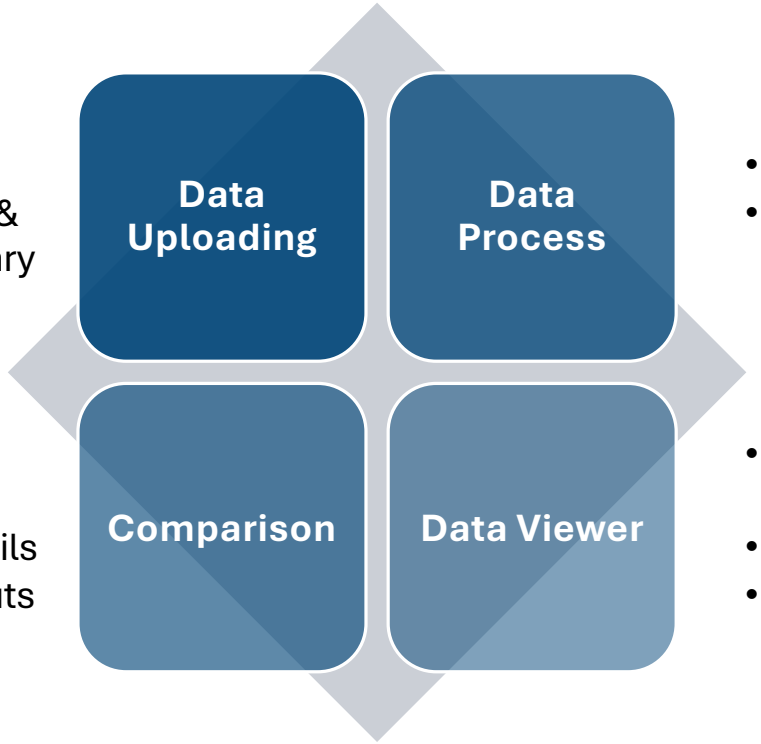
Data Viewer

- Side-by-side aligned tables (RAW vs SDTM)
- Enables detailed, record-level review with synchronized filters

Dashboard

The 4 Main Tabs

- Batch file upload
- Smart domain detection & real-time domain summary

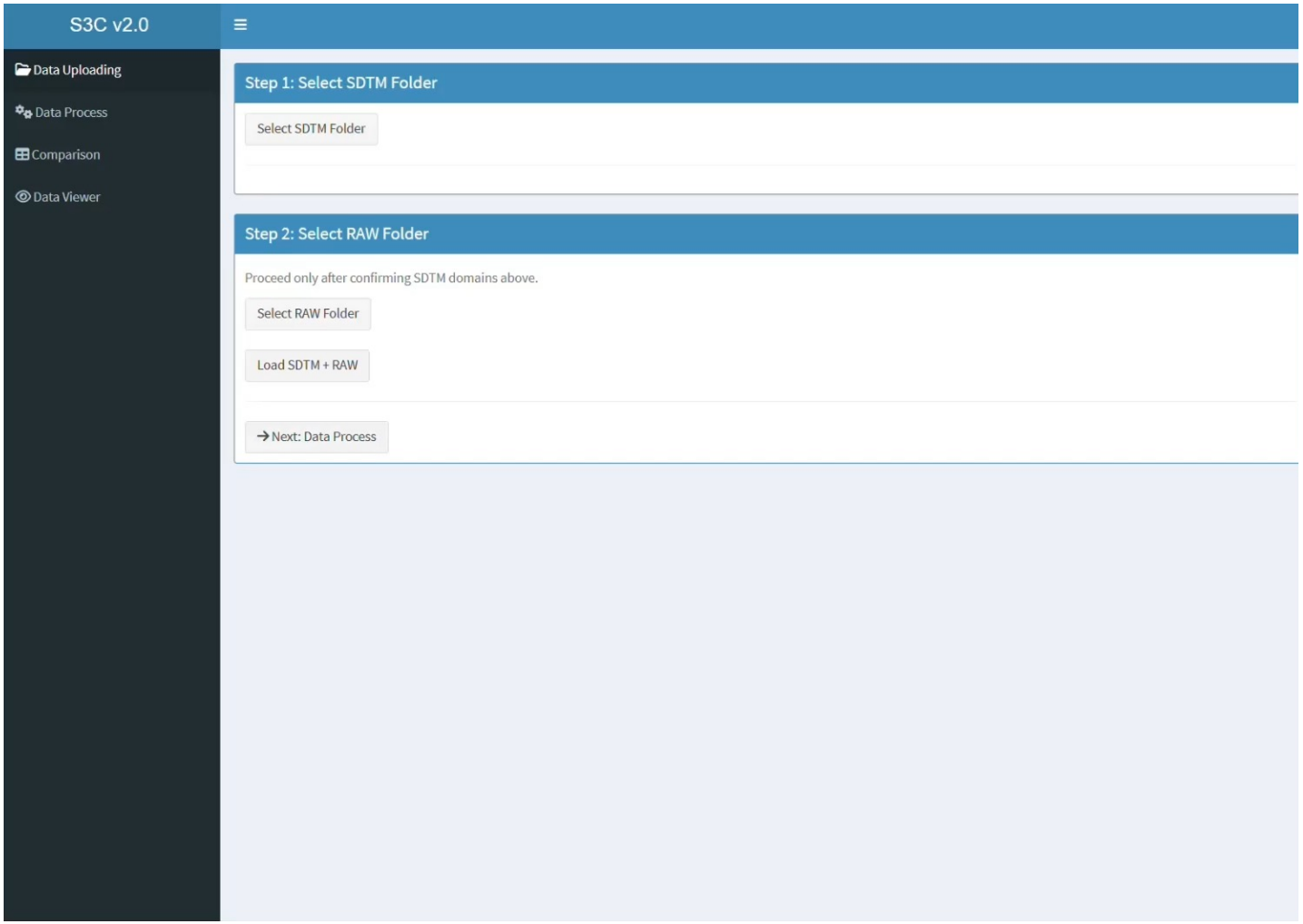


- Load RAW + SDTM variables
- Prepare data for mapping and comparison

- Domain-level & variable-level differences
- SAS-style mismatch details
- Export comparison outputs

- Side-by-side aligned tables (RAW vs SDTM)
- Synchronized filters
- Flexible column selection

Demo


S3C v2.0
Data Uploading
Data Process
Comparison
Data Viewer
Step 1: Select SDTM Folder
Select SDTM Folder
Step 2: Select RAW Folder
Proceed only after confirming SDTM domains above.
Select RAW Folder
Load SDTM + RAW
→ Next: Data Process

Modular Design

What is YAML?

YAML is a very simple, text-based, human-readable language used to exchange data between people and computers.

Y**A**ML is not a programming language. It is mostly used for storing configuration information.

YA**M**L files store metadata, and they do not include commands and instructions.

YAM**L** is a data serialization language similar to XML or JSON. However, it is much more human readable and concise.

Why Use YAML?

Human-readable

- YAML is very much human readable. It allows you to represent complex data structures in a human readable manner.

Simple and clean syntax

- YAML is easy to learn and simple to read. It can easily express wide variety of different data structures.

Easy to implement and use.

Unambiguous

- YAML unambiguously specifies the data structures of the serialized data, so there's no need to rely on comments or documentation. Because the data structures are unambiguous, it makes it easier to use automated tools (scripts) for reading and writing YAML. For example, you could write a script that reads the data structure from a file written in YAML and converts it to another format (such as JSON or XML).

The YAML Structure

```yaml
Metadata:
  Rule_Name: "DTHFL_Variable"
  Version: "1.0"
  Author: "Davide Marinucci"
  Date_Created: "2025-09-29"
  Last_Updated: "2025-09-29"
  Status: "Active"

Mappings:
  Sources:
    RAW_Corporate:
      SERAE:
        Variable:
          - "STUDYID"
          - "SUBJECT"
          - "AESDTH"
        Label:
          - "Study Code"
          - "Enrolment Code"
          - "Results in Death"
    SDTM:
      DM:
        Variable: "USUBJID"
        Label: "Unique Subject Identifier"
  Map_Definition: "Merges 'Y' to DM.DTHFL where SERAE.STUDY/SERAE.SUBJECT=DM.USUBJID and AESDTH='C49488' (source)"
  Map_Rule: "DTHFL"
  Map_Mode: "MACRO"
  Map_Order: 1013
  Target_Dataset: "DM"
  Target_Variable: "DTHFL"
  Target_Label: "Subject Death Flag"
  Type: "character"
  Core: "Exp"
```

1. **Header:**
   - **Metadata information about the rule**

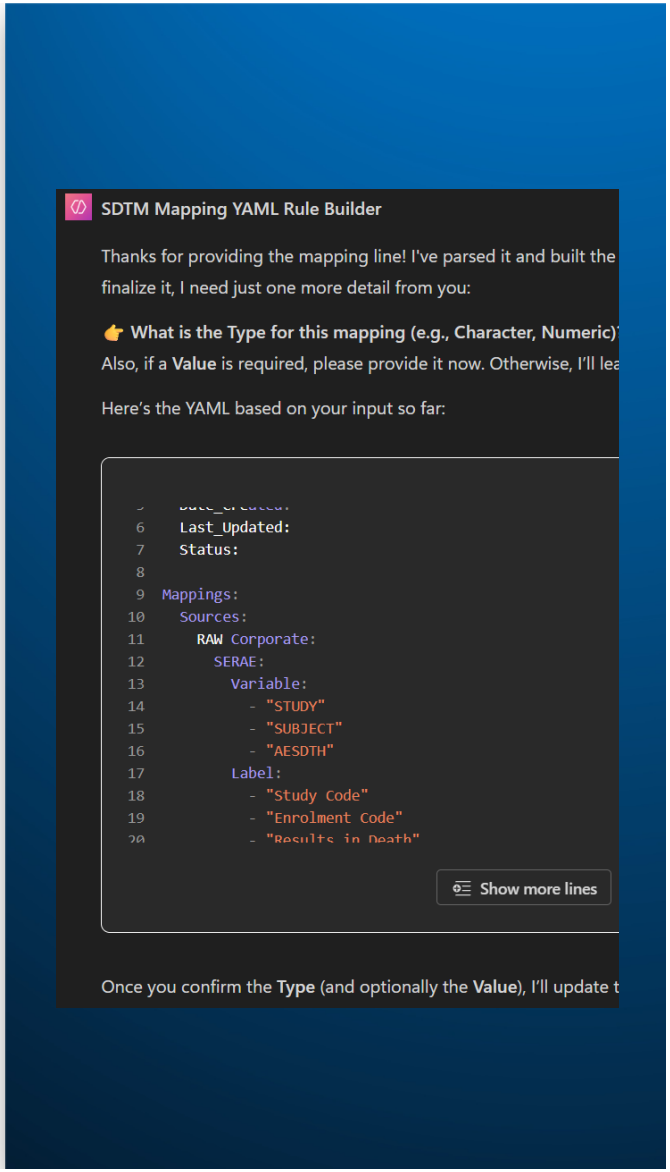
2. **Input:**
   - **Sources**
   - **Datasets/Domains**
   - **R Derivation function**

3. **Output:**
   - **Domain**
   - **Variable Name/Label**
   - **Type**

AI Agent

AI Agent — YAML Rule Builder

- **Automates** YAML rule creation
- Ensures **consistent** RAW→SDTM mappings
- **Reduces** manual effort & errors
- Generates **ready-to-use** domain rules for S3C



Results

Validation Status: Study 1

CMSPID:
Incorrect SDTM derivation

CMCAT:
Study-level derivation

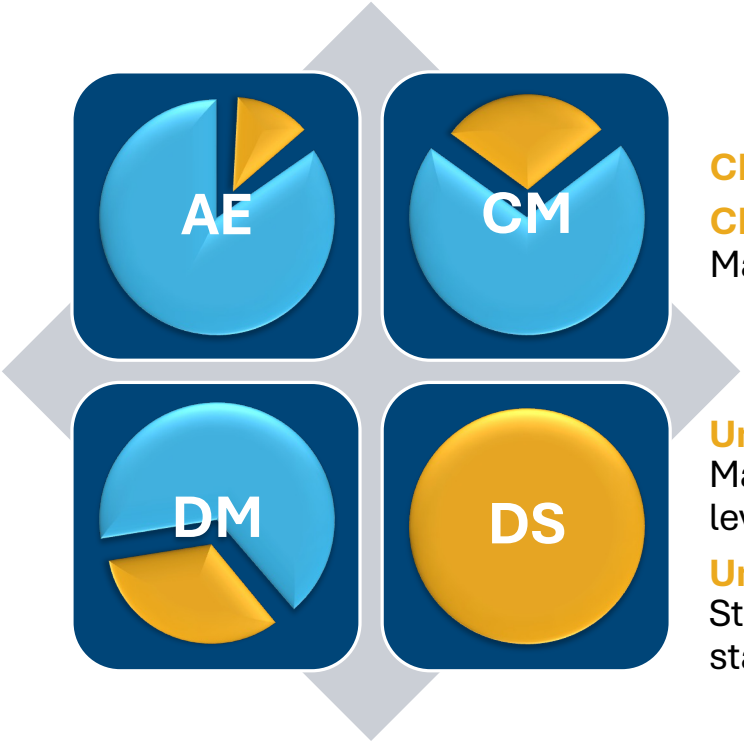
CMDOSU:
Mapping standards version

Validation Status: Study 2

AEREL/ AEACN :
Study-level derivation

AGE/ RACE/ RFPENDTC:
Study-level derivation

RFXSTDTC/ RFXENDTC/ RFSTDTC/ RFENDTC:
Mapping standards version



CMCAT: Study-level derivation

CMDOSE/ CMDOSU:
Mapping standards version

Unequal records:
Mapping standards version; Study-level derivation;

Unequal variables:
Study-level derivation; Mapping standards version

Learning & Limitations

What we learned	A lightweight, YAML-driven tool can meaningfully support SDTM vendor oversight without duplicating conformance engines.
	Centralized YAML + R logic scales better than ad-hoc scripts and makes checks more maintainable.
	Combining synthetic data and reuse data helped validate the approach and tune the rules.
What still hurts	Complex, study-specific derivations still demand manual, domain-expert validation.
	Unequal records/variables may reflect mapping changes or study-level derivations, not just “errors” → interpretation still needs humans.
	Scaling YAML coverage and keeping rule catalogs aligned with evolving standards is ongoing work.

Thank You!

Questions?