



2026 CDISC China Interchange

SCORE (Sanofi CDISC Open Rules Engine)

Enterprise Best Practice of CDISC CORE Open Source Engine

Zoe Cai,

Analytics and Reporting Technology, Sanofi, Chengdu, China



Disclaimer

The opinions expressed in this presentation are those of the author(s) and do not necessarily reflect the official views and policies of Sanofi.

All data presented in this presentation are simulated. No real patient information is used or disclosed.



Speakers



M.Sc. in Biostatistics from the University of Toronto
With 5+ years of cross-therapeutic experience in SAS, R, Python, and Git for open-source automation and clinical reporting tools, currently also serving as Business Analyst for eWise, Sanofi's computing and storage platform.

蔡徐思乔 Zoe Cai

Sr Statistical Programmer
Sanofi

Agenda

01. The Engine

02. The Strategy

03. The Implementation





01. The Engine

Demystifying CDISC CORE Rules Validation Framework



What is CDISC CORE?



PROJECT ORIGIN

Derived from **the CDISC Open Rules Project**, aiming to eliminate textual ambiguity by transforming traditional PDF guidance into machine-readable execution.



TWO ESSENTIAL PILLARS

CDISC Open Rules: A governed, standardized, and unambiguous set of machine-executable Conformance Rules defined for each CDISC Foundational Standard (e.g., SDTM, ADaM)

CDISC Open Rules Engine (CORE): The actual open-source reference execution engine used to process and run those executable rules.

VALIDATION RULES FLOW & EXECUTION LOGIC



CDISC Community Rules

STANDARD

Authoritative, governed rules loaded dynamically via the official CDISC Library API.



Custom Rules

SPONSOR

Tailored criteria defined locally by users in YAML/JSON format using the standard rules template.



CORE Dynamic Rule Ingestion & Loading (CLI / Cache Setup)



Data Input

Parameter Configuration



CORE Engine

Executes selected rules against clinical datasets via command line input.



Report Output

Comprehensive error & warning summary (XLSX/JSON)

CDISC Open Rules Coverage

Rule Authoring Status for v1.0

Standards Track	# Rules	# Complete	% Complete
SDTMIG v3.2 / v3.3 & 3.4	411 / 505	376 / 461	91% / 91%
SENDIG v3.1.1	312	222	71%
FDA Business Rules SDTMIG / SENDIG	186 / 181	103 / 118	55% / 65%
TIG v.1.0	484	409	85%
USDM v.3.0 &v4.0	260	257	99%
Unique Rules	1883	1550	82%



Open Rule v 1.0 & 2.0

Target: Late 2026

Project Release Scope (Ref: Appendix 1):

- SDTMIG, SENDIG, and ADaMIG rules
- FDA Business Rules (SDTMIG & SENDIG)
- USDM
- Define-XML v2.1

| The CORE Engine's Dual Model



Executable Package

A standalone command-line package that requires zero local configuration and is ready for immediate extraction and execution.

- **Pros:** Minimal deployment overhead; eliminates the need for local Python environment setup.
- **Cons:** Substantial file size, low extensibility, and potential updates latency compared to the core repository.

</> From Source Code

Running the execution engine by pulling the source code directly from the GitHub repository via a local Python interpreter.

- **Pros:** Maximum agility and seamless customization; enables real-time synchronization with the latest upstream main branch features, optimized for deep pipeline integration and debugging.
- **Cons:** Higher technical entry barrier; requires proactive management of complex Python runtime environments and package dependencies.



| The Interface Hurdle

The Friction of Native CLI Implementations



No Standard GUI

- 100% reliant on Command Line Interface (CLI).
- Highly user-unfriendly for non-technical teams



Zero-Tolerance Syntax

- Requires precise input of dozens of complex CDISC parameters.
- Any misplaced space, missing forward slash, or simple typo triggers an immediate execution error.

```
PS C:\Users\e0496855\OneDrive - Sanofi\Desktop\ART\12 CORE\02 exe\core-windows\core> .\core.exe --help
Usage: core.exe [OPTIONS] COMMAND [ARGS]...
```

```
Options:
  --help  Show this message and exit.
```

```
Commands:
  list-ct           Command to list the ct packages available in the...
  list-dataset-metadata  Command that lists metadata of given datasets.
  list-rule-sets      Lists all standards and versions for which rules...
  list-rules
  test-validate       Verify CORE with a test validation.
  update-cache
  validate            Validate data using CDISC Rules Engine
```

```
PS C:\Users\e0496855\OneDrive - Sanofi\Desktop\ART\12 CORE\02 exe\core-windows\core> .\core.exe validate -s sdtmig -v 3-4 -d
p "C:\Users\e0496855\OneDrive - Sanofi\Desktop\ART\12 CORE\TEST_FILE\dm.xpt"
```

```
[██████████] 100%
[DEBUG 2026-06-04 13:53:49,762 - excel_report.py:101] - Report written to: CORE-Report-2026-06-04T13-53-21.xlsx
Output: CORE-Report-2026-06-04T13-53-21.xlsx
```

02. The Strategy

Why should Sanofi build our own SCORE?



| Driving Factors for SCORE



Industry Authority

- Leverages the direct compliance tool provided officially by CDISC.
- Enables instant dataset validation against global regulatory submission standards.



TA-Level/Study-Level Control

- Utilizes CORE's native Custom Rules functionality.
- Empowers teams to build specific validation rules tailored to distinct Therapeutic Areas or individual protocols.



FDA Real-Time Clinical Trial

- Real-time trials strictly demand continuous, Real-Time Validation instead of traditional retrospective cleaning.
- CORE is a ready-to-use solution for real-time validation.

| Breaking Vendor Lock-in

100%
OPEN-SOURCE

Eliminating Proprietary Automation Friction

No API Paywalls: Commercial community suites restrict CLI/API integration to expensive enterprise licensees.

Native System Integration: Open architecture enables direct injection into corporate pipeline automation layers, triggering checks instantly update data ingestion.

Empowered Intelligence: Enables easy interfacing with LLMs to automatically diagnose, explain, and recommend programmatic corrections.

03. The Implementation

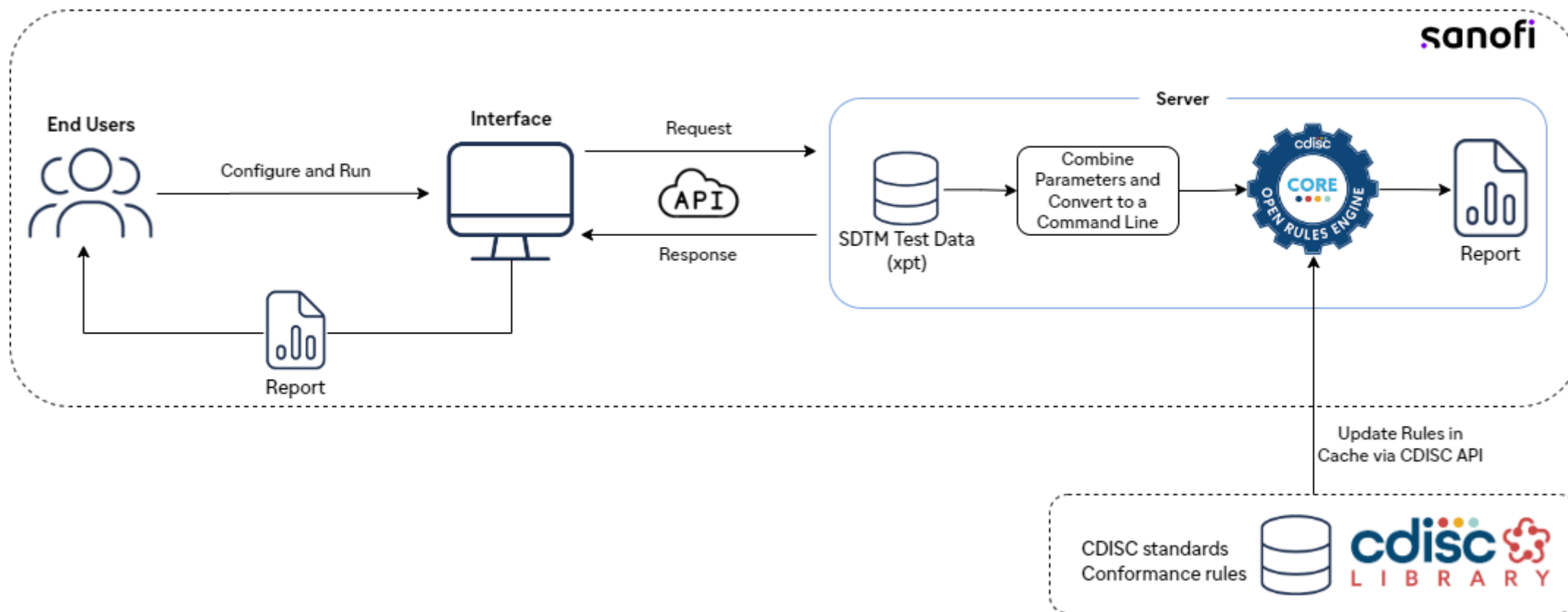
SCORE Development & Autonomous AI Integration Roadmap



Phase 1: Human-Centric

Hiding Command Line Complexity

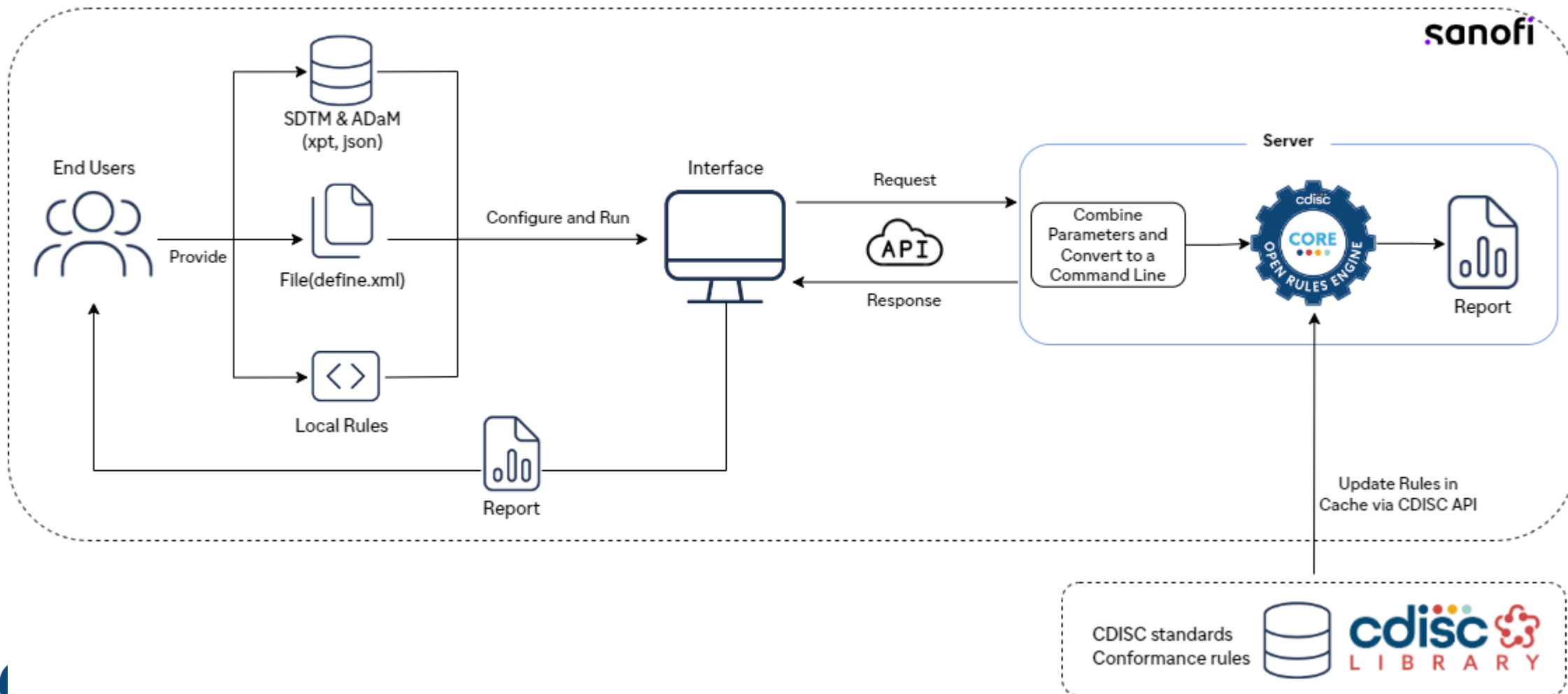
Step 1 – Prototype, Everything is in Server



Phase 1: Human-Centric

Hiding Command Line Complexity

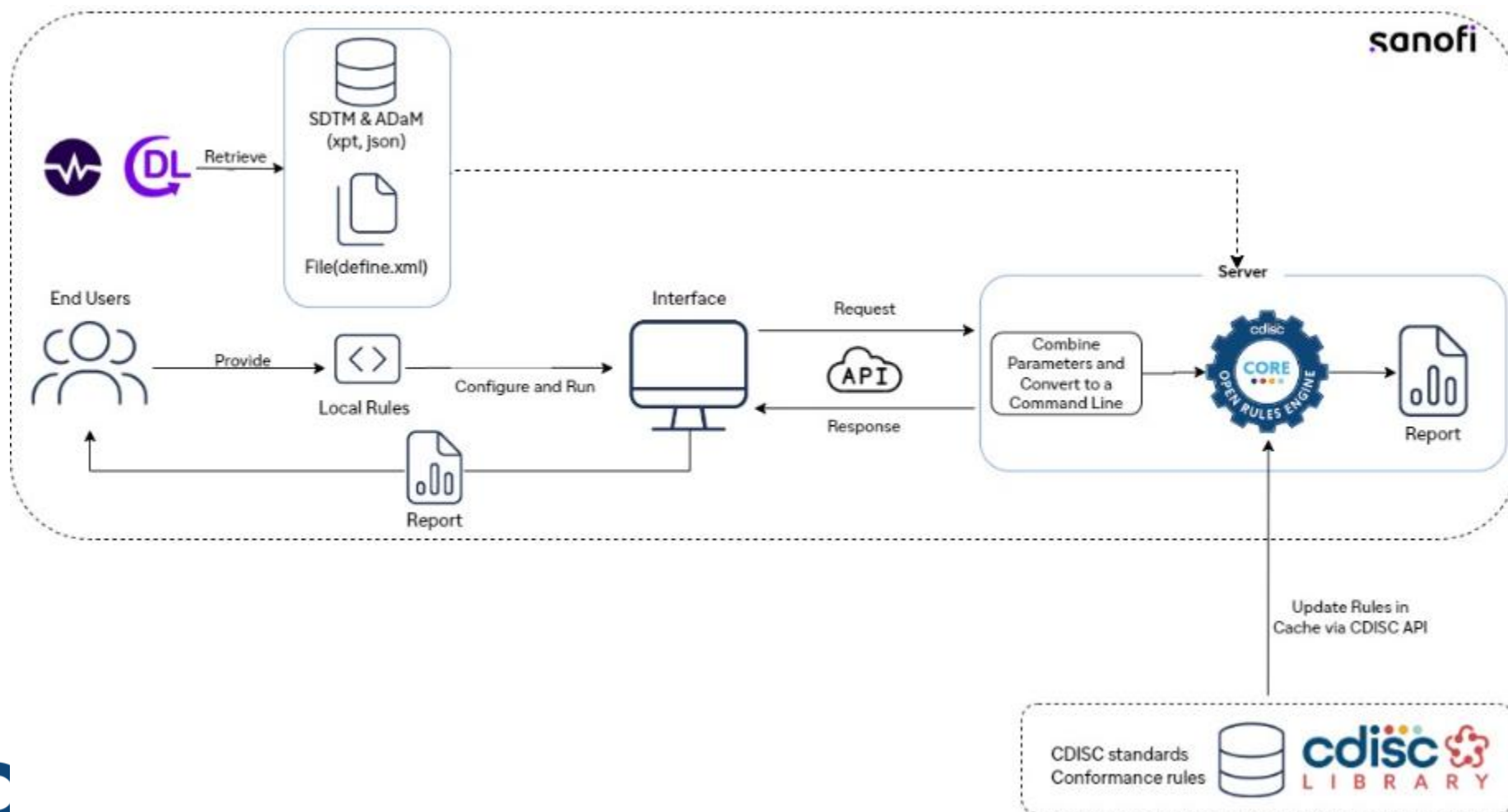
Step 2 – Data and files are from end user



Phase 1: Human-Centric

Hiding Command Line Complexity

Step 3 – Data and files are from eWise & Clinical Data Lake



Phase 1: Human-Centric

DEMO

 SCORE

    User

 Validator

 Reporter

OTHERS

 Documentation

 Raise Support

Engine: 0.13.0

Standard	Required SDTM-IG ▼	Version	Required SDTM-IG 3.4 ▼	Triplet	Required ▼
Data Source	☒ Default ☐ Upload your data				
Data	Required ▼				

Validate

Reset

Phase 2: Next-Gen AI Agent

Automated Data Remediation Loop

CORE PIPELINE INTEGRATION

1. Metadata Extraction (via Melodins)



2. CORE Validation (Engine Run)



3. Structured Conformance Report

STEP 01

Intelligent Diagnosis

AI Agent ingests CORE execution errors and cross-references CDISC rule contexts. Automatically generates deep-dive diagnostic insights and remediation proposals.

STEP 02

Isolated Sandbox Repair

To secure production data, the Agent code-generates script patches or data corrections inside a strictly **isolated sandbox environment**.

STEP 03

One-Click Review

Enforces **Human-in-the-Loop** governance. Programmers and DM review proposed changes through an intuitive visual Diff interface before approval.

STEP 04

Audit & Re-validation

Approved fixes trigger a GCP-compliant Audit Trail. The system automatically loops back to Step 1, re-running extraction and CORE validation until errors hit zero.



Acknowledgement

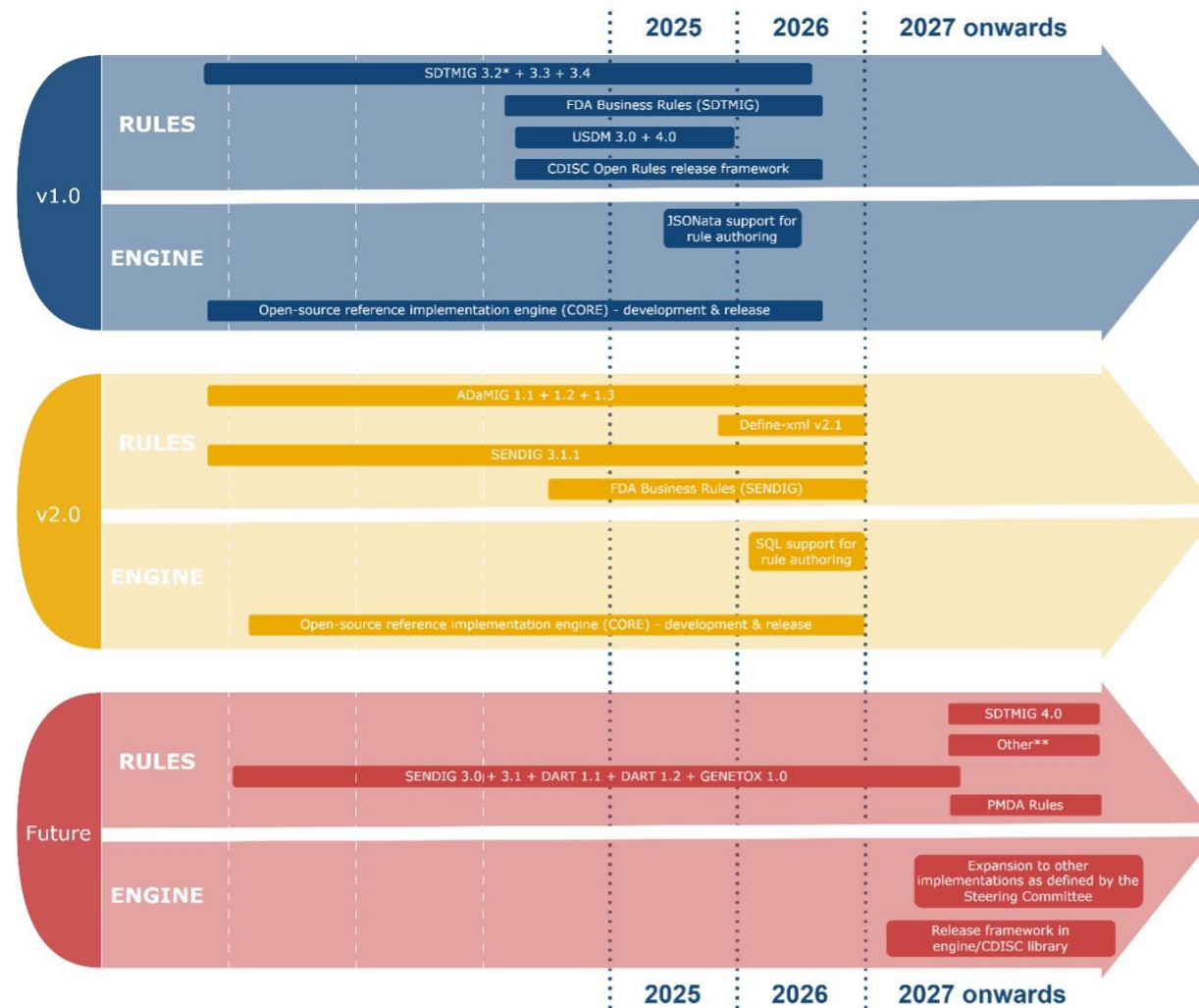
A sincere thank you to Sanofi colleagues and the development team:
Jianfeng Ye, Yuanlin Ma, Marco Li for their contribution in the development.

For further questions or feedback, please feel free to contact me:
Zoe.Cai @sanofi.com

Thank You!



APPENDIX 1: CDISC Open Rules Roadmap



CORE | CDISC

* SDTMIG 3.2 will be created but are not required to be complete for v1.0

** All remaining standards not regulatory required, updates to existing conformance rules, AI-driven rule authoring,...