Paper PP07

# Building CORE Strength: An Implementation Exercise

Addie Nina Olsen, Novo Nordisk A/S, Copenhagen, Denmark
Stine Winkler Bertelsen, Novo Nordisk A/S, Copenhagen, Denmark

## ABSTRACT

Validation of clinical data using CDISC conformance rules is essential for ensuring consistency, quality and interoperability of data. The CDISC Open Rules Engine (CORE) project provides both machine-readable rules and an open-source engine for data validation.

We present our learnings from implementing the CORE engine in an automated SDTM generation flow on a SAS Viya® platform: the challenges that we have encountered and the promising aspects of the implementation. In addition, we outline plans for enhancing our CORE implementation with explanations and problem-solving tips that can assist the end user in aligning to CDISC standards.

## INTRODUCTION

International standards for clinical study data accelerate drug development by enabling automation of data processing and by simplifying submissions to regulatory authorities. Standards also facilitate pooling and reuse of data. Adherence to standards developed by the Clinical Data Interchange Standards Consortium (CDISC) is a requirement for submission to both FDA and PMDA. Consequently, it is important for pharmaceutical companies to have tools for ensuring compliance with these standards.

Though the first widely used tool for compliance checks was open-source software, the area has since become dominated by commercial software suppliers [1]. Recently, the CDISC Open Rules Engine (CORE) project [2] was established with the goal of defining a set of unambiguous and executable conformance rules for CDISC standards and creating an open-source engine for rule execution. It is an initiative that holds great promise, and this paper presents our plans for integrating CORE in an automated SDTM generation flow along with learnings from the initial implementation.

## CORE INTEGRATION

### PLAN OUTLINE

SDTM generation in Novo Nordisk is highly standardised, centralised and automated. It is metadata-driven and based on SAS macros. The SDTM generation flow is currently in the process of being moved to a cloud-based SAS Viya® platform, and it is on this platform that the CORE engine will be integrated.

The intention is to execute CORE from SAS code as part of the automated SDTM generation flow (see ***Figure 1***), so that the CORE output is readily available for study teams throughout the conduct of a clinical study. The expectation is that this will help the study teams address data issues faster and that a continuous cycle of issue identification and resolution will be adopted instead of a resource-heavy, intense effort near the end of a study.

In addition to automated execution of CORE, the ambition is to make it easy for non-programmers to manually execute CORE on datasets of their selection. This could be useful for evaluation of SDTM-structured data delivered from CROs or for ad hoc checks on submission data.

### INITIAL IMPLEMENTATION

The CORE project has made the CORE Engine Reference Implementation available in GitHub [3]. As a proof of concept for the use of the engine on a SAS Viya® platform, the CORE repository was cloned, and the relevant Python dependencies were installed. The installation resulted in a fully functional CORE engine that was successfully used to validate data. However, it will be evaluated whether it is feasible and preferable to use one of the Linux-tailored executable versions of the engine that are also available on GitHub.

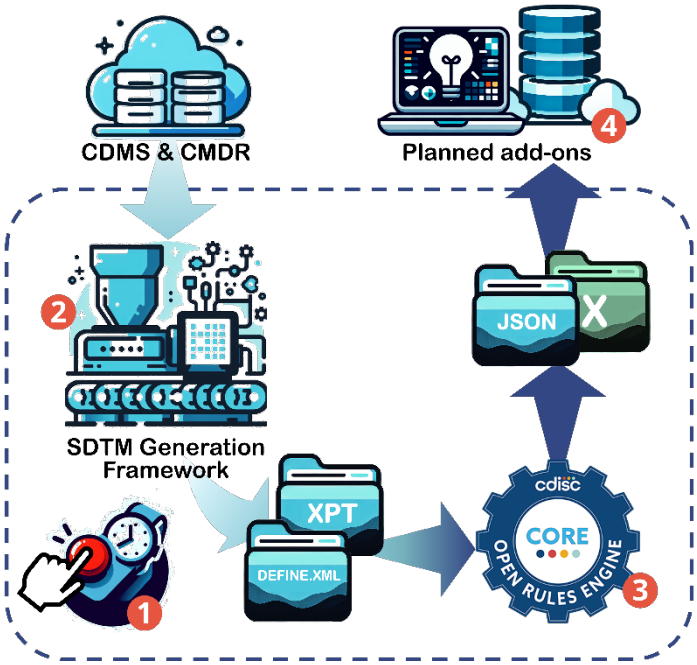


**Figure 1: Integration of CORE into SDTM generation flow**

1) SDTM generation is initiated by a scheduler or on demand.

2) SDTM data is automatically generated by a framework of SAS macros based on pre-SDTM data extract from CDMS and metadata. Part of the output is submission-ready XPT files and define.xml.

3) CORE engine is executed during a subsequent step in the automated flow.

4) Planned add-ons based on collection of validation issues in a database. This will facilitate addition of summaries and visualisations, as well as issue-specific fix tips and explanations.

## INITIAL LEARNINGS

### CORE CHALLENGES

A challenge with the initial implementation of CORE is the limited guidance available for the engine. The CORE project relies on volunteers for development of the engine and the CORE rules. Understandably, guidance on installation and usage of the tool is therefore limited to a Readme file on GitHub. There are no instructions on how to get help if you have questions or challenges. This could become a barrier for widespread adoption of the engine.

From a practical perspective, it is also important to keep in mind that implementation of CORE in a data flow is not a one-off activity. The engine available on GitHub is still under development and new features are likely to become available on an ongoing basis. Implementation of a new engine version will require resources for tool validation and for the practical implementation tasks. In addition, the tool will have to be kept up to date with releases of new controlled terminology versions and with rules corresponding to new CDISC implementation guides. There will of course always be maintenance tasks related to a tool which is used to enforce standards that evolve over time, even for commercial products. However, maintenance could require more IT competencies for an open-source tool integrated in a bespoke data flow compared to off-the-shelf software.

### CORE PROMISES

Based on our proof-of-concept implementation, it is evident that CORE can easily be integrated in a data flow. The command-line interface can be used to execute CORE, irrespective of operating system (Linux, macOS or Windows). For our SDTM generation flow, executing CORE from a SAS program was the obvious choice and it was simple to do. The simplicity of integration means that the engine will fit well into diverse data flows without requiring big changes to accommodate it.

The flexibility of the CORE engine integration also extends to the engine output as it can produce validation results in both Excel and JSON formats. The JSON format makes it easy to consume the validation results with downstream add-ons where the results can be further processed.

Another promising aspect of the CORE project is the possibility of including rules from other rule authors than CDISC. It has been announced that CDISC and FDA will collaborate to incorporate FDA business rules in CORE [4]. Furthermore, the ability to include user-specific rules is an integral part of the CORE project concept [5], which will make it possible to perform checks for sponsor-specific extensions to CDISC standards. Combining CDISC rules with authority-specific rules and sponsor-defined rules, will make CORE a very powerful validation tool that can promote compliance with standards from multiple origins.

## BEYOND THE CORE

Identifying issues in clinical data is only the first step in ensuring compliance with standards. The second step is to resolve the issues, if possible. Towards this end, the plan is to maintain a set of fix tips linked to CORE rule identifiers

to give study teams easy access to tips on how the identified issues can be solved. Preferably, the issues produced by CORE execution should be collected in a database and presented to the study teams together with the relevant fix tips in a custom-built user interface. In scenarios where it is not possible to solve an issue, the interface should also provide access to standardised, explanatory text that can be included in the Study Data Reviewer's Guide.

Collection of CORE output in a database will furthermore make it possible to summarise issues within a study and across studies, enhanced by the addition of relevant visualisations and statistics. Summarising issues within a study will make it easier for the study team to prioritise their problem-solving efforts and to monitor development in issue count over time. Cross-study issue summaries will make it possible to spot trends in data issues, e.g. for a specific SDTM domain or a specific validation rule. This could reveal problems originating in data collection that need to be addressed with process optimisation. Evaluating trends in data issues could furthermore be used to optimise standard programs for SDTM generation.

## CONCLUSION

The initial implementation of CORE in our SDTM generation framework has demonstrated that it is a tool, which is still in its early phases of development, but also that it holds great promise due to its flexibility and extensibility. Our plan is to follow its development closely and to use it as a base for building a user-friendly validation flow that will facilitate compliance with standards and speed up the delivery of high-quality clinical data.

## REFERENCES


1. Sirichenko, S., DiGiantomasso, M. and Collopy, T. (2016). Usage of Pinnacle 21 Community Toolset 2.1.1 for Clinical Programmers. *PharmaSUG 2016 Conference Proceedings*. https://www.pharmasug.org/proceedings/2016/HT/PharmaSUG-2016-HT04.pdf
2. https://www.cdisc.org/sites/default/files/2021-06/CDISC_CORE_Project_Charter.pdf
3. https://github.com/cdisc-org/cdisc-rules-engine/blob/main/README.md
4. https://www.cdisc.org/news/cdisc-proud-announce-research-collaboration-incorporate-fda-business-rules-cdiscs-open-rules
5. https://www.cdisc.org/core


## CONTACT INFORMATION

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


- Addie Nina Olsen, Novo Nordisk A/S, aieo@novonordisk.com
- Stine Winkler Bertelsen, Novo Nordisk A/S, dsae@novonordisk.com