CORE: Checking USD for Conformance

25 June 2024
Agenda

- Digital Data Flow and USDM
- Conformance Rule Coverage for USDM
- CORE Demo
- Planning for phase 4
Digital Data Flow and USDM
TODAY: Document-based paradigm for protocol creation, interpretation, and transcription into consuming systems

TOMORROW: Digital paradigm for protocol creation, with fully automated data flow and interoperability between systems
The USDM Standard

**Logical Model**
The UML logical model (a class diagram) that provides the basis for the USDM standard.

**CDISC Controlled Terminology**
Provides further semantics, complementing the UML model. Includes the definition of classes and attributes along with the definition of value sets.

**API Specification**
Provides the means to exchange a single study between machines using a JSON API.

**Examples**
Example protocols implemented in the USDM with associated JSON files and visualisations.

**Implementation Guide**
Guidance on using the USDM model and ensuring conformance with the standard.
CDISC DDF / USDM: Phases One, Two and Three

- **Phase One**
  - 25 Classes
  - Solid foundation
  - The protocol document was an external entity into which the structured content could be exported

- **Phase Two**
  - 35 Classes
  - Focused on the structured elements of the protocol, e.g. the Schedule of Activities (SoA) & Biomedical Concepts (BCs)
  - The protocol document still an external entity

- **Phase Three**
  - 58 Classes
  - Now contains structured and unstructured elements
  - The entire protocol document can be held within the USDM
  - Allows for the protocol document to be generated from the model
USDM Content

- Controlled Terms
- Estimands
- Populations
- Inclusion & Exclusion
- Interventions & Indications
- Objectives & Endpoints
- Procedures, Biomedical Concepts
- Study, Identifiers, Amendments
- Unstructured Content
- Study Designs, Arms, Epochs
- Detailed Study Logic, Encounters
• Conformance Rule Coverage for USDM
Introduction

CDISC Conformance Rules are an integral part of the Foundational Standards and serve as the specific guidance to Industry for the correct implementation of the Standards in clinical studies. An emerging Industry best practice is to use Conformance Rules on an ongoing basis, throughout the study, to keep the data as close to submission ready as possible and to ensure quality in all data exchange scenarios.

Current CDISC Conformance Rules need to be expressed in a common specification to be loaded to the CDISC Library. In addition, an executable component must be developed for every Conformance Rule.

Project Goals and Objectives

The overall goal of the CORE Project is to deliver a governed set of unambiguous and executable Conformance Rules for each Foundational Standard, and to provide a Reference Implementation of an open-source execution engine for the executable Rules.

The global clinical research community will be able to leverage the free and open CORE software to test study data for conformance to CDISC standards as well as to regulatory and sponsor-specific conformance rule sets.

The CORE Project objectives are to:

- Ensure each standard has a set of unambiguous, executable Conformance Rules
- Ensure consistency across Conformance Rule implementations
- Expedite the availability of executable Conformance Rules for new Foundational Standards
- Create executable Conformance Rules vetted by the CDISC standards development teams
- Create a Reference Implementation of an open-source engine that executes the Rules
- Release the open-source engine under the CDISC Open-Source Alliance (COSA)

https://www.cdisc.org/core
USDM CORE rules

• Purpose
  • Check that exchanged JSON API file is correct according to USDM logic and implementation guide
  • Inform users of incorrect and/or unlikely content via pre-specified rules
  • Inform users of correct implementation via logical representation of content rules

• Phase 3
  • Proof Of Concept

• Phase 4
  • Rule set aligning with USDM 3.0
  • Rule set aligning with USDM 4.0
DDF3A CORE POC Use Case and Scope

• Demonstrating that USDM JSON files are USDM compliant,
  • e.g., Transfer of USDM JSON file from one organization to another (e.g., Vendor to Sponsor)

• Scope
  • Develop a representative set of conformance rule specifications (up to 100)
  • In addition, develop a representative set of machine executable Conformance rules (in YAML) to cover a wide breadth of the different types of rules to demonstrate DDF conformance

• Details
  • Enhance the CORE Rule Editor and schema to handle various DDF rule types
  • Develop and test a representative set of machine executable conformance rules (in YAML format)
  • Enhance the CORE open-source engine to run these Conformance Rules against a USDM JSON file
  • Scope CDISC Library modifications (to store DDF conformance rules and the USDM model)
Test Data Template

- Excel workbook based on existing SDTM test data template
  - Designed to align with current Rules Editor functionality (e.g., .xpt suffix for sheet names) to minimize effort required for POC
  - Link between json and Core engine
- Programmatically generated from USDM UML, CT and API
  - In general, 1 sheet per USDM class/entity
  - Linking attributes retained

|---------------|-----------|------------|-------------|-----------------|--------------------|----------------------------|-------------------------------|-------------------|
Core engine process updates in POC
Running the CORE Engine

• CLI executable available in GitHub
  • Cached rules
  • Windows, Mac, and Linux install packages
  • Unzip and run
  • Will need datasets to validate

• Engine available on PyPI
  • Engine is a component that can be used in your own code
Running the CORE Engine

- Desktop versions
  - Vendor released versions of CORE
  - Includes a user-friendly UI
  - Easier for non-technical users to evaluate
Conversion of USDM JSON File

- The CORE engine converts each USDM class in the selected JSON file into tabular format before applying the rules.

---

**Study**

<table>
<thead>
<tr>
<th>parent_entity</th>
<th>parent_id</th>
<th>parent_rel</th>
<th>rel_type</th>
<th>id</th>
<th>name</th>
<th>description</th>
<th>label</th>
<th>versions</th>
<th>documentedBy.id</th>
<th>documentedBy.name</th>
<th>documentedBy.label</th>
<th>documentedBy.description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wrapper</td>
<td>study</td>
<td>definition</td>
<td>Study_CDISC_PILOT - LZZT</td>
<td>TRUE</td>
<td>TRUE</td>
<td>StudyProtocolDocument_1</td>
<td>Pre</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Code**

<table>
<thead>
<tr>
<th>StudyVersion</th>
<th>parent_entity</th>
<th>parent_id</th>
<th>parent_rel</th>
<th>rel_type</th>
<th>id</th>
<th>type</th>
<th>definition</th>
<th>code</th>
<th>codeSystem</th>
<th>codeSystemVersion</th>
<th>C98388</th>
<th><a href="http://www.cdisc.org">http://www.cdisc.org</a></th>
<th>2023-09-29</th>
</tr>
</thead>
</table>

**AliasCode**

|--------------|---------------|-----------|------------|----------|----|-----------------|-------------------|-----------------------------|-----------------------------|-------|--------|---------|----|----------|

**GovernanceDate**

<table>
<thead>
<tr>
<th>StudyVersion</th>
<th>parent_entity</th>
<th>parent_id</th>
<th>parent_rel</th>
<th>rel_type</th>
<th>id</th>
<th>name</th>
<th>label</th>
<th>dateValues</th>
<th>GovernanceDate_1</th>
<th>D_APPROVE</th>
<th>Design Approval</th>
<th>Design approval</th>
<th>2023-09-29</th>
<th>ISO 3166 1 alpha3</th>
<th>2020-08</th>
<th>Europe</th>
</tr>
</thead>
</table>

---

The discontinuation rate associated with this oral dosing regimen was 58.6% in previous studies, and alternative clinical strategies have been sought to improve tolerance for the compound. To that end, development of a Transdermal Therapeutic System (TTS) has been initiated.
CORE Report Generated by the Test Run

CORE Report
- Generated in Excel
- Placed in the CORE folder
- Datetime stamp in name
Rule examples

• Schema
  • Model: data types and relationships according to json API specification
  • References:
    • Previous/next ordering of encounters, eligibility criteria, epochs etc
    • All specified BCs, procedures, categories referenced by activity
    • Target references exist.

• Conditional
  • Attributes/Content:
    • fully defined timing window
    • At least 1 address field specified
  • Content:
    • Subject enrolment units
    • isRequired and isEnabled dependency
    • 1 objective with level primary
    • Age, gender, enrolment number etc specified at either cohort or study population level
Rule examples

• **Conditional**
  • Complex:
    • Unique arm-epoch combination
    • Study type matches number of interventions
    • Alignment between encounter and epoch ordering with timeline

• **Unique**
  • Ids: within and between classes
  • Codes:
    • Only 1 entry per title type, signature type etc
    • No repeats of codes for 1 entry
  • Text:
    • Name description label within a class
    • Code system version
    • Section numbers
CORE rule features used for USDM

• Existing rule features like:
  • Equal_to
  • Does_not_contain_case_insensitive
  • Is_not_unique_set

• New rule features created like:
  • Enable test data format
  • Joining datasets
  • …

• New rule features still to be created
  • JSON schema check
  • XML format check
  • Links to DDF/external codelists
  • Complex joins and cross checks
  • … ?
• Planning for Phase 4
Scope of Phase 4

• USDM version 3.0 final set of rules
• Add new YAML functionality to enable all anticipated rules like for:
  • Complex cross-checks
  • Check CT
  • Null values checks
  • API Json schema check
  • XML format check
• Create improved reporting template
• USDM version 4.0 final set of rules
• Add an exemplar study