CDISC CORE Project
(CDISC Open Rules Engine)
Challenge / Opportunity

- CDISC Conformance Rules are an integral part of the CDISC standards
- CDISC and the Community must govern the Rules
  - Validated by the Community
  - Remain current over time
- Rules must be published in CDISC Library
  - The single source
  - In a common specification
- Executable component must be developed for every Rule
- Open-source solution to ensure Community can run the checks
- Anticipated that Health Authorities will accept and adopt the Rules
CORE Objectives

1. Each standard has a set of unambiguous, executable conformance rules
2. Ensure consistency across conformance rule implementations
3. Expedite the availability of executable conformance rules for new standards
4. Create executable reference rules blessed by the CDISC standards team
5. Publish these reference rules under COSA as open-source
6. Create an open-source execution engine that works with the reference rules
CORE – Further Considerations

• CORE will be released as open source under the MIT license
  • Not offered by CDISC as a commercial product or service

• Executable rules - next step in the evolution of the conformance rules that CDISC publishes with every standard

• Executable rules published by CDISC should make it much easier for rule vendors to adapt these rules for use in their own software

• Existing rule vendors are free to contribute to or use the CORE engine software

• CDISC Core Press Release
CORE Concept

COMMUNITY GOVERNANCE

CDISC Community

Executable

Specification

Conformance rules

Data Standards & rules

Load

Conformance rules

User Specific Rules

Configure and run

Data & Metadata

End User

Report

INTERFACE

3rd Party application

API

CORE in COSA

Engine

API

Azure

CORE Business Requirements

- High-level Requirements have been prepared
- Requirements include a plan for 3 releases
- A release package includes
  - Open-source engine
  - Conformance Rules
  - Rules executable component
- The releases are: Minimum Viable Product, Release 1, Release 2
- Minimum Viable Product (MVP)
  - Engine – has enough features to be usable by early users who can then provide feedback for future product development
  - Rules Specification and executable component – focuses on SDTM conformance rules
  - SDTM 2.0 and SDTMIG 3.4 in scope for MVP
  - Does not exclude other (ADaM, SEND, Define.xml) but not critical for MVP
Stakeholders

• **The Community:**
  • Collaborates with CDISC Standards Team on creation, governance and maintenance of conformance rules specification

• **CDISC:**
  • Responsible for specification of the rules (in collaboration with the Community), and provisioning of the rules in the CDISC Library

• **Microsoft: Software Engineering Partner**
  • Responsible for development of the rules’ execution engine and interface
CORE team structure

**CORE Leadership**
- Product Owner
  - CORE System Architect
  - Program manager
  - Software engineering partner
- Community leaders

**Architecture**
- Technical environment
- DevOps

**CORE Software Engineering**
- DEV team
  - CORE engine
  - Software engineering partner (CDISC)
  - CDISC Community contributors

**CDISC Library**
- DEV team
  - Load rules
  - Surface rules
  - Provide API
  - CDISC VP Data Science
  - CDISC Data Science team

**Conformance Rules**
- DEV team
  - Analyze
  - Identify
  - Maintain
  - Rule SME Analysts (CDISC)
  - Rule SME Analysts (Community)
  - SME programmers (CDISC)
  - SME programmers (Community)

**CORE QA team**
- Testing strategy
- Validation plan

**CDISC Data Science team**
- CDISC Standards team
- Community Leads
- Community members
CORE Leadership

- Decision Log
- Prioritize
- Communication

Product Owner
CORE System Architect
Program manager
Software engineering partner
Community leaders

• Responsibility:
  - Leadership on all aspects of project execution
  - Project methodology
  - Project structure
  - Project scope
  - Product roadmap
  - Product backlog
  - Release plan

• Recruitment – volunteers
• Recruitment – staff
• MVP scope
• Groom product backlog
• Governance
CORE Architecture

• **Responsibility:**
  • Project DevOps environment
  • High-level requirements
  • Recruitment - staff
  • Tool selection
  • Deliver Non-Functional Requirements
  • System architecture
  • Data architecture
  • Deliver Product Backlog Items

• **Membership:**
  • System Architect: CDISC VP Data Science
  • Rules execution engine developer leads
  • CDISC IT lead

Architecture

- Technical environment
- DevOps

CORE System Architect
CDISC Library Architect
Azure System Administrator
Software engineering partner
Software Engineering DEV Team

- **Responsibility:**
  - Development of the open-source rules’ execution engine
  - Beta release package definition
  - Deliver Product Backlog Items
  - MVP release deployment

- **Membership:**
  - Software engineering partner
  - Software engineer (x2)
CDISC Library DEV Team

**Responsibility:**
- Load conformance rules to the Library, both the specification and its executable form
- Surface the rules from the Library
- Library rules analysis
- Library architecture updates
- Beta release package definition

**Membership:**
- Software engineering Library
- CDISC Library Architect
- Software engineer (x2)
Conformance Rules DEV Team

- **Responsibility:**
  - The rules' specification and the executable form of the rules
  - Executable rules development
    - Includes testing
    - Test Documentation
  - Deliver Product Backlog Items
  - User stories / input in Interface
  - Input in High-level requirements
  - Conformance Rules Governance process

- **Membership:**
  - SDTM/SDTMIG SMEs
  - Rule Developer
  - Rule Programmers
QA Team

• Responsibility:
  • Analysis and development of validation plan
  • Analysis and development of test data
  • Execution of validation plan
  • Execute CORE engine for executable rules and test data
  • Report and analyze test results
  • Coordinate with Software Engineering DEV team on test results

• Membership:
  • Validation Lead
  • Validation SMEs
  • Technical Writers
  • Security Engineer (3rd party)
Scrum of Scrums

- **Responsibility:**
  - Ongoing scrum of scrums for the agile methodology used in the project
  - Sprint Planning meeting
  - Daily stand-ups
  - Sprint Review meeting
  - Sprint Retrospective meeting
  - Release Plan updates
  - Governance

- **Membership:**
  - CDISC Scrum Master
  - CDISC VP of Data Science
  - Workstream team leads
Project Governance

- CDISC will provide
  - The Product Owner
  - The CORE System Architect
  - A Program Manager
  - A Project Manager / Scrum Master

- These CDISC roles will manage and prioritize the product backlog and deliverables developed in partnership with the software engineering partner

- Project teams report on progress & issues at Scrum of Scrums, Sprint Planning, and Sprint Review meetings

- Unresolved issues will be addressed at CORE Leadership Team meetings
  - Leadership Team will maintain a log of project issues and their resolution
Project Governance: Activities

CORE Leadership Governance Activities
- Establish governance plan
- Resolve escalated issues
- Maintain decision log
- Communicate to stakeholders

Scrum Master / Program Manager Governance Activities
- Maintain Product backlog
- Identify and resolve issues
- Escalate unresolved issues to CORE Leadership

Team Leads Governance Activities
- Maintain Team backlog
- Identify and resolve issues
- Escalate unresolved issues to Scrum Master

CORE Leadership

Scrum Master / Program Manager (Scrum of Scrums)

CORE System Architect
- Software engineering team operational lead
- Library team operational lead
- Conformance Rules team operational lead
- QA team operational lead

Architecture
Engine
Library
Rules
QA
Project Governance: Meeting Cadence

**CORE Leadership**
- **Team Lead Meeting Cadence**
  - Weekly

  **CORE Leadership Meeting Cadence**
  - Weekly

  **Team Member Meeting Cadence**
  - Architecture, Engineering & Library:
    - Scrum meeting - daily
    - Sprint Review - fortnightly
  - Rules & QA:
    - Team meeting - weekly
    - Sprint Review - fortnightly

**Scrum Master / Program Manager (Scrum of Scrums)**

**Architecture, Engineering & Library**
- **Team Lead Meeting Cadence**
  - Scrum meeting - daily
  - Sprint Review - fortnightly

**Library**
- **Team Lead Meeting Cadence**
  - Team meeting - weekly
  - Sprint Review - fortnightly

**Rules**
- **Team Lead Meeting Cadence**
  - Team meeting - weekly
  - Sprint Review - fortnightly

**QA**
- **Team Lead Meeting Cadence**
  - Team meeting - weekly
  - Sprint Review - fortnightly

**CORE System Architect**

**Software engineering team operational lead**

**Library team operational lead**

**Conformance Rules team operational lead**

**QA team operational lead**
### Meeting Cadence

<table>
<thead>
<tr>
<th>Week 1</th>
<th>Week 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SUNDAY</strong></td>
<td><strong>SUNDAY</strong></td>
</tr>
<tr>
<td>Team Scrums</td>
<td>Team Scrums</td>
</tr>
<tr>
<td>Scrum of Scrums</td>
<td>Backlog Grooming</td>
</tr>
<tr>
<td><strong>MONDAY</strong></td>
<td><strong>MONDAY</strong></td>
</tr>
<tr>
<td>Team Scrums</td>
<td>Team Scrums</td>
</tr>
<tr>
<td><strong>TUESDAY</strong></td>
<td><strong>TUESDAY</strong></td>
</tr>
<tr>
<td>Team Scrums</td>
<td>Team Scrums</td>
</tr>
<tr>
<td>CORE Lead Team</td>
<td>CORE Lead Team</td>
</tr>
<tr>
<td><strong>WEDNESDAY</strong></td>
<td><strong>WEDNESDAY</strong></td>
</tr>
<tr>
<td>Team Scrums</td>
<td>Team Scrums</td>
</tr>
<tr>
<td>Scrum of Scrums</td>
<td>Backlog Grooming</td>
</tr>
<tr>
<td><strong>THURSDAY</strong></td>
<td><strong>THURSDAY</strong></td>
</tr>
<tr>
<td>Team Scrums</td>
<td>Team Scrums</td>
</tr>
<tr>
<td><strong>FRIDAY</strong></td>
<td><strong>FRIDAY</strong></td>
</tr>
<tr>
<td>Team Scrums</td>
<td>Team Scrums</td>
</tr>
<tr>
<td><strong>SATURDAY</strong></td>
<td><strong>SATURDAY</strong></td>
</tr>
</tbody>
</table>

* Some Teams will scrum less than daily; actual cadence TBD
Ask for Industry Participation

• Focal teams:
  - Conformance Rules DEV team
  - QA team

• Conformance Rules DEV team
  • Develop the rules Specification and the executable form of the rules

• Work areas:
  • Specification structure and content
  • Executable rules content

• Team members:
  • SDTM/SDTMIG SMEs
  • Rule Developer
  • Rule Programmers
Ask for Industry Participation (2)

• QA team
  • Develop and execute validation plan for rules Specification and executable form of rules
  • Work areas:
    • Analysis and development of validation plan
    • Analysis and development of test data
    • Execution of validation plan
      • Execute CORE engine for executable rules and test data
      • Report and analyze test results
      • Coordinate with Software Engineering DEV team on test results

• Team members:
  • Validation Lead
  • Validation SMEs
  • Technical Writers
  • Security Engineer (3rd party)
Ask for Industry Participation (3)

- Project execution period
  - 2021 Q3 – 2022 Q1 (about 9 months)

- Expected FTE level
  - Minimum 20%

- Meeting types and cadence
  - Based upon agile Kanban methodology
  - Meetings
    - Team meeting once or more per week, as determined by team
    - Attend other DEV teams’ Sprint Review once fortnightly (every other week)
Timeline

For the minimum viable product:

- Initiation: 2021 Q2
- Planning: 2021 Q2
- Execution: 2021 Q3 - 2022 Q1
- Closure: 2022 Q1
Thank You!