

CDISC CORE AND 'HOMEMADE CORE': ENHANCING DATA CHECK OPERATIONS

June 28, 2023



AGENDA

CORE

3

RULES

21

Homemade CORE

29

CORE



CORE - Introduction

CDISC CORE is a project that aims to deliver clear and enforceable Conformance Rules for each Foundational Standard. It provides an open-source execution engine as a Reference Implementation for these rules.



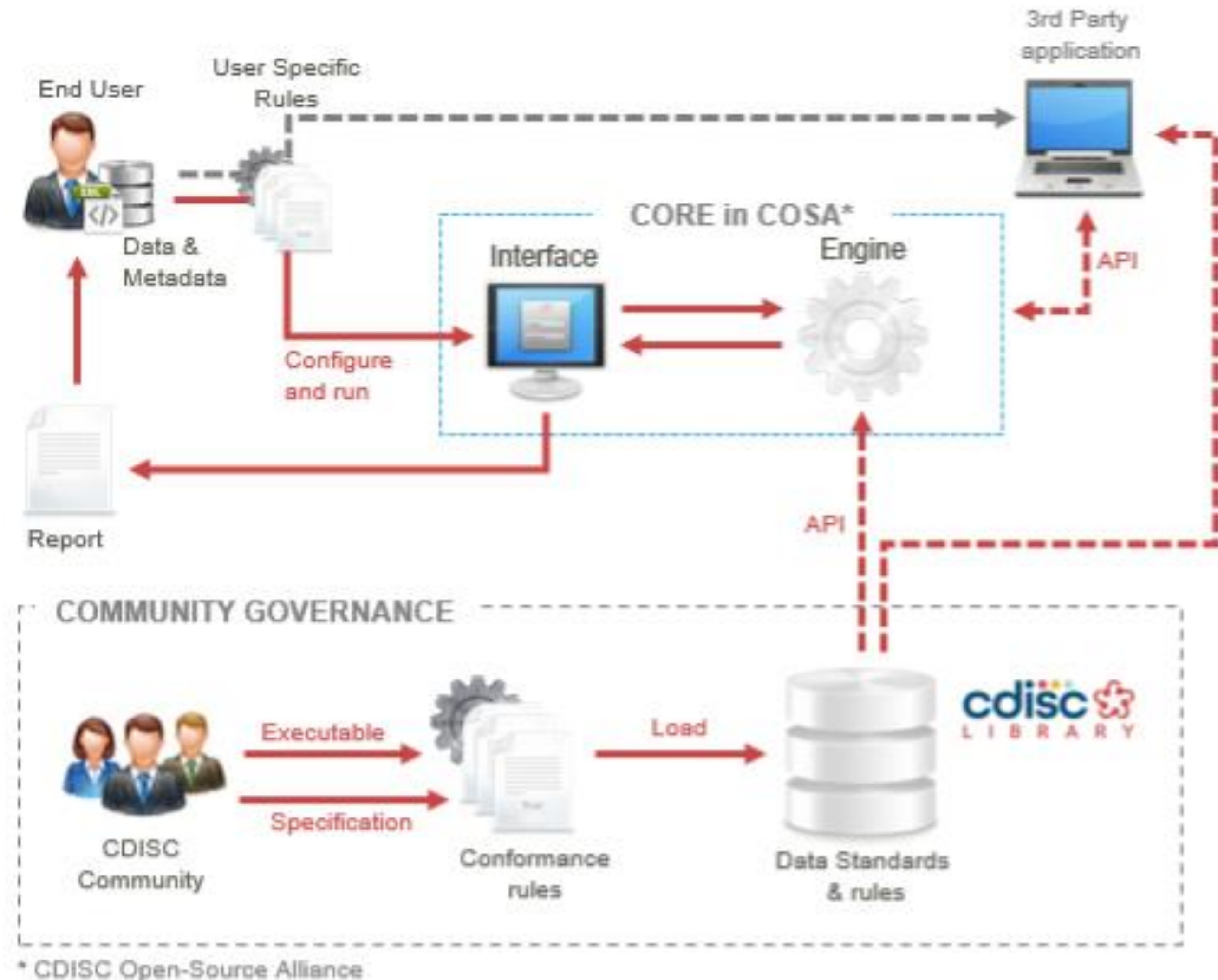
CORE - Objectives

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)

CORE - Project Concept Diagram

The following diagram illustrates the concept of the CORE project, including the Conformance Rules, the executable form of the Rules, and the Rules execution engine:



CORE Program Roadmap



Q3 2023 – Q2 2024

Production Release 2: Enhanced Engine and Rules

- • **Engine:** Open-Source under COSA; evolved; maintained by the open-source software community
- • **Conformance Rules:** New CDISC Standards released with Conformance Rules
- • **Functionality:** Advanced functionality
- • **Deployments:** Vendor- or user-provided cloud & local production environments

Q3 2022 – Q2 2023

Production Release 1: Submission-ready Engine and Rules

- • **Engine:** Open-Source under COSA; evolved; maintained by the open-source software community
- • **Conformance Rules:** Remainder of CDISC Foundational Standards
- • **Functionality:** Complete conformance checking functionality
- • **Deployments:** Vendor- or user-provided cloud & local production environments

Establish CORE Roadmap Board

Q3 2021- Q2 2022

Evaluation Release

- • **Engine:** Open-Source, developed by CDISC, published under COSA
- • **Conformance Rules:** SDTM 2.0 and SDTMIG 3.4
- • **Functionality:** Basic conformance checking functionality
- • **Deployments:**
 - CDISC cloud evaluation
 - Azure Marketplace evaluation
 - Desktop evaluation – **not achieved yet**



- CDISC
- Vendor/User

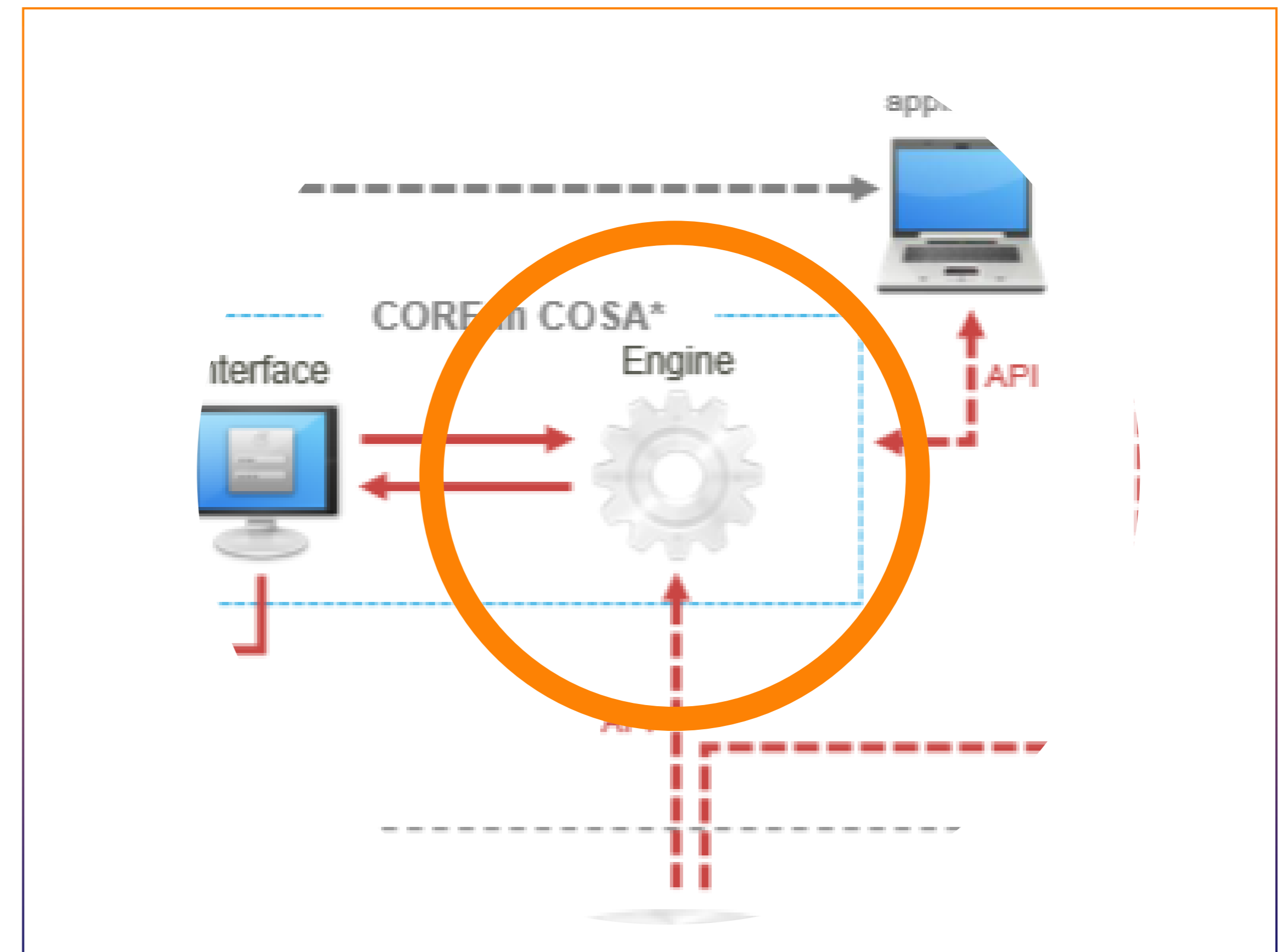
CORE – Engine

Functionality:

- > Executes CORE Rules (YAML) against clinical data and returns results
- > Deployment agnostic
- > Open-source, available in GitHub

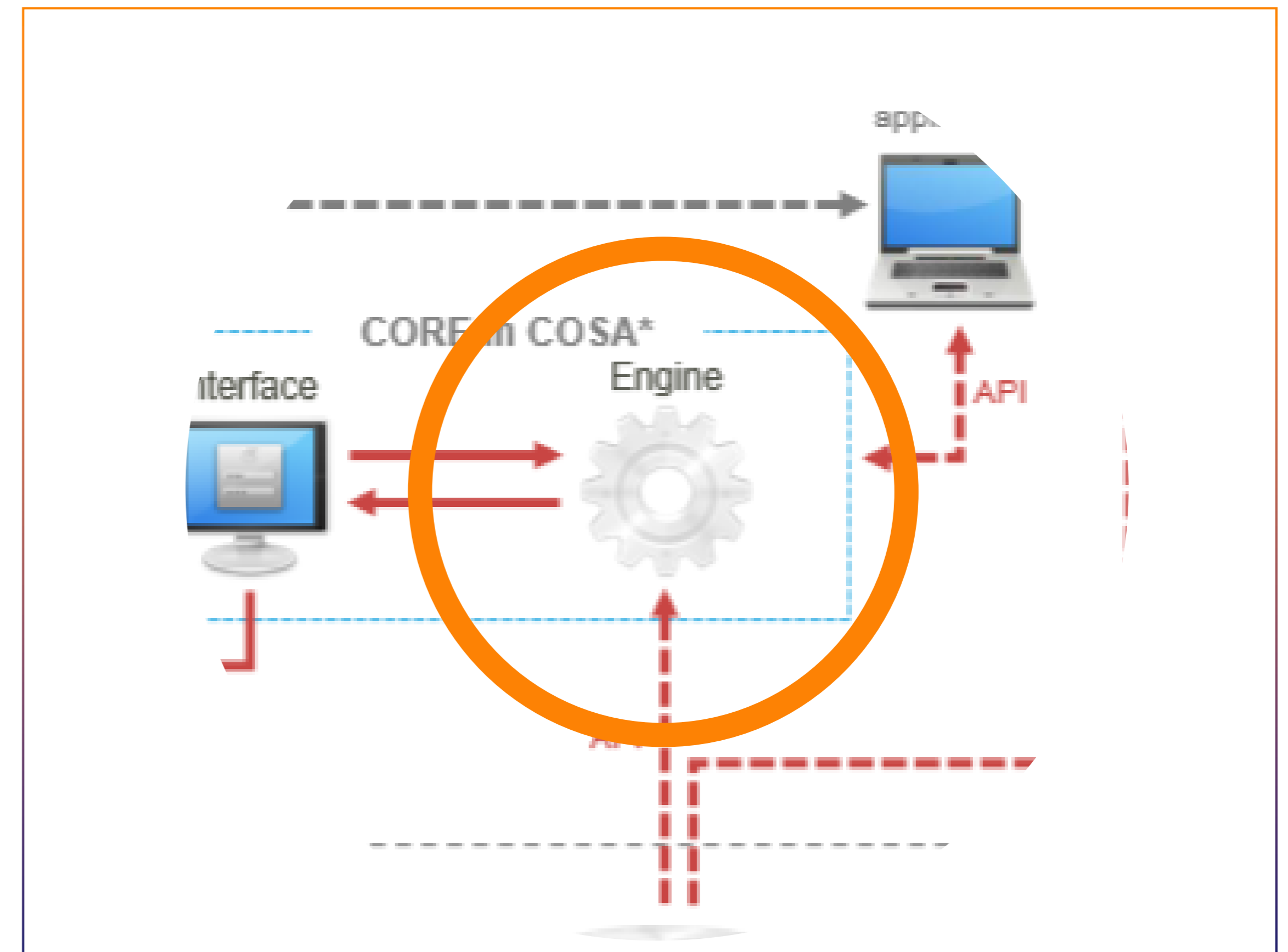
Current focus:

- > Process new YAML operators added to express new rules
- > Process new clinical data formats
- > Support Define xml crosschecking

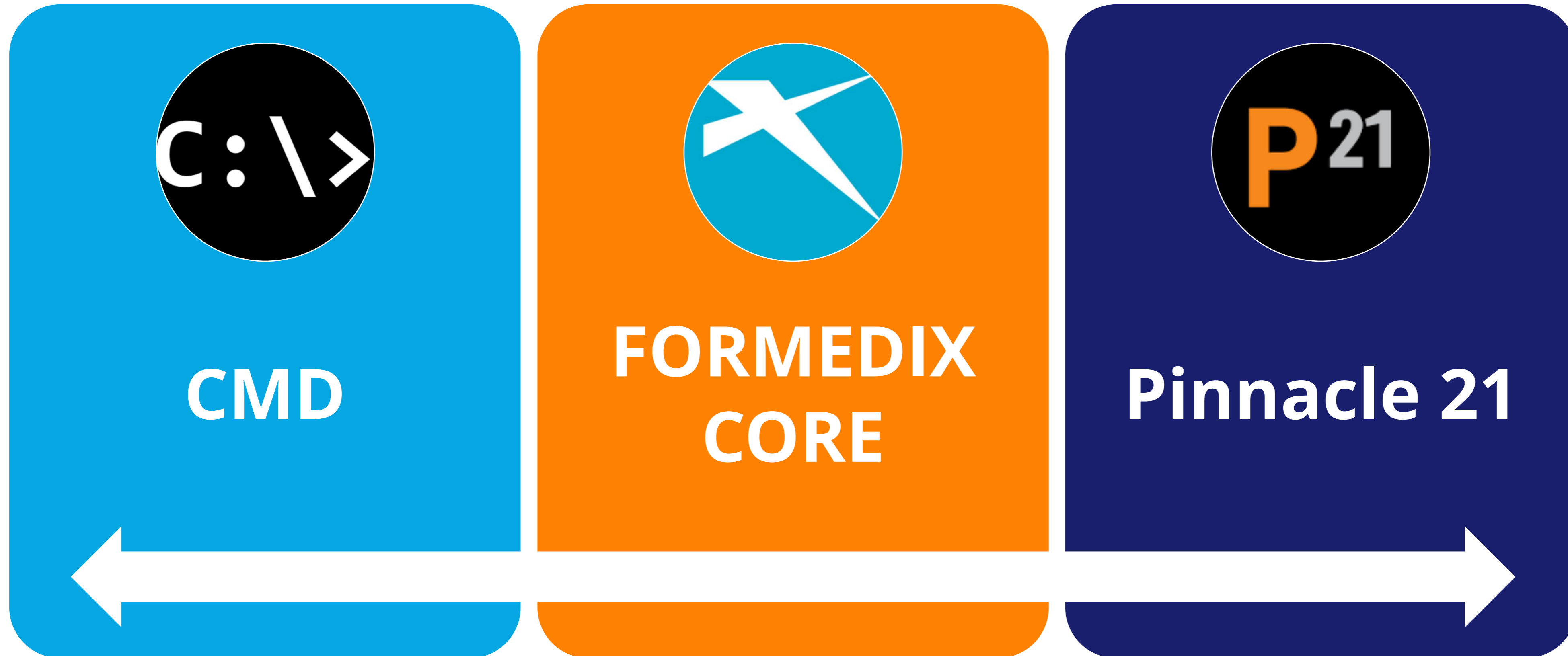


CORE – Is Open-Source

- > Open-source framework
- > Listed in the COSA (CDISC Open-Source Alliance) directory
- > Permissive MIT open-source license
- > Provided via GitHub
- > Free to all in CDISC community
- > Very flexible implementation options



CORE – In Action



CORE – CMD

The CDISC Core Engine provides a command-line interface (CMD), which serves as an interface for users to interact with the software and execute a wide range of operations

```
C:\Users\Shadow\Desktop\CORE\core-windows>.\core validate -s sdtmig -v 3-2 -d .\xpt\  
[████████████████████████████████████████] 10%
```

- .\CORE
- VALIDATE
- -S SDTMIG
- -V 3-2
- -D .\XPT\

--HELP

```
-ca, --cache TEXT           Relative path to cache files containing pre  
                             loaded metadata and rules  
-ps, --pool-size INTEGER    Number of parallel processes for validation  
-d, --data TEXT             Path to directory containing data files  
-dp, --dataset-path TEXT    Absolute path to dataset file. Can be specified multiple times.  
-l, --log-level [info|debug|error|critical|disabled|warn] Sets log level for engine logs, logs are  
                             disabled by default  
-rt, --report-template TEXT File path of report template to use for  
                             excel output  
-s, --standard TEXT         CDISC standard to validate against  
                             [required]  
-v, --version TEXT          Standard version to validate against  
                             [required]  
-ct, --controlled-terminology-package TEXT Controlled terminology package to validate  
                             against, can provide more than one  
-o, --output TEXT           Report output file destination  
-of, --output-format [JSON|XLSX] Output file format  
-rr, --raw-report           Report in a raw format as it is generated by  
                             the engine. This flag must be used only with  
                             --output-format JSON.  
-dv, --define-version TEXT  Define-XML version used for validation  
--whodrug TEXT              Path to directory with WHODrug dictionary  
                             files  
--meddra TEXT               Path to directory with MedDRA dictionary  
                             files  
-r, --rules TEXT            Rule core id. ex: CORE-000001. Can be specified multiple times.  
-vo, --verbose-output       Specify this option to print rules as they  
                             are completed  
-p, --progress [verbose_output|disabled|percents|bar] Defines how to display the validation  
                             progress. By default a progress bar like  
                             "[████████████████████████████████████████]-----"  
                             "78%" is printed.  
--help                       Show this message and exit.
```

CORE – CMD Results

Conformance Details

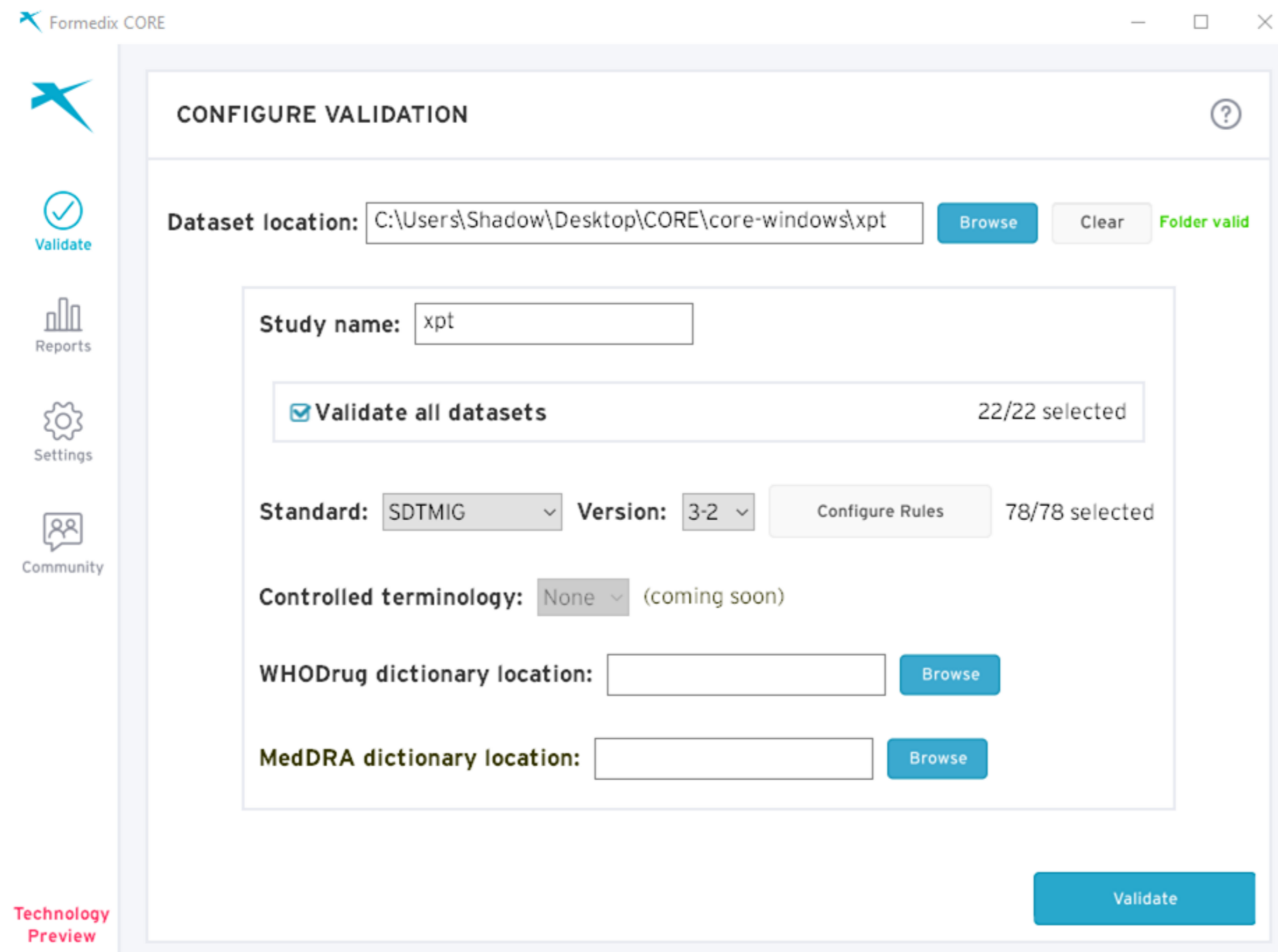
A	B
Conformance Details	
Report Generation	2023-05-22T05:39:17
Total Runtime	63.5 seconds
CORE Engine Version	0.6.1
Standards Details	
Standard	SDTMIG
Version	V3.2
CT Version	
Define-XML Version	
UNII Version	Not configured
Med-RT Version	Not configured
MedDRA Version	Not configured
WHODRUG Version	Not configured
SNOMED Version	Not configured

Issue Details

A	B	C	D	E	F	G	H	
CORE-ID	Message	Executability	Dataset	USUBJID	Record	Sequence	Variable(s)	Value(s)
CORE-000022	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	fully executable	AE	01-701-1192	108	7	AESCAN, AESCONG, AESDISAB, AESDTH, AESER, AESHOSP, AESLIFE, AESMIE, AESOD, AESTDTC, AETERM	N, N, N, N, N, Y, N,
CORE-000022	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	fully executable	AE	01-701-1192	109	10	AESCAN, AESCONG, AESDISAB, AESDTH, AESER, AESHOSP, AESLIFE, AESMIE, AESOD, AESTDTC, AETERM	N, N, N, N, N, Y, N,
CORE-000022	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	fully executable	AE	01-701-1211	121	9	AESCAN, AESCONG, AESDISAB, AESDTH, AESER, AESHOSP, AESLIFE, AESMIE, AESOD, AESTDTC, AETERM	N, N, N, Y, N, N, Y, I
CORE-000022	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	fully executable	AE	01-704-1008	312	2	AESCAN, AESCONG, AESDISAB, AESDTH, AESER, AESHOSP, AESLIFE, AESMIE, AESOD, AESTDTC, AETERM	N, N, N, N, N, Y, N,

CORE – FORMEDIX

Formedix CORE is a free, downloadable Windows desktop application that provides an easy way to run validations on local data and identify standards conformance issues.



RESULT

CORE-ID	Message	Executability	Dataset	USUBID	Record	Sequence	Variable(s)	Value(s)
2	CORE-000022 AESEER = N or empty.	*fully executable	AE	01-701-1192	108		7 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-09-07, PNEUMONIA
3	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-701-1192	109		10 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-09-07, PNEUMONIA
4	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-701-1211	121		9 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, Y, N, Y, Not in dataset, N, 2013-01-14, SUDDEN DEATH
5	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-704-1008	312		2 AESMIE, AESOD, AESTDTC, AETERM	*ATTACK
6	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-704-1445	409		1 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, Y, N, N, Not in dataset, N, 2014-10-31, COMPLETED SUICIDE
7	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-705-1393	432		3 AESMIE, AESOD, AESTDTC, AETERM	*Y, N, N, N, Y, N, Not in dataset, N, 2013-01-21, COLON CANCER
8	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-706-1049	447		2 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-06-18, SYNCOPSE
9	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-708-1178	489		7 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2014-03-24, ATRIAL FIBRILLATION
10	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-708-1178	491		6 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2014-03-24, DIZZINESS
11	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-709-1259	635		5 AESMIE, AESOD, AESTDTC, AETERM	*HYPERPLASIA
12	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-709-1259	636		6 AESMIE, AESOD, AESTDTC, AETERM	*HYPERPLASIA
13	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-709-1259	640		9 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-04-13, HYPOTENSION
14	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-709-1285	650		1 AESMIE, AESOD, AESTDTC, AETERM	*N, N, Y, N, N, N, Not in dataset, N, 2013-04-14, HEMIANOPIA HOMONYMOUS
15	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-709-1326	683		4 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-05-21, SYNCOPSE
16	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1002	690		1 AESMIE, AESOD, AESTDTC, AETERM	*ATTACK
17	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1002	691		2 AESMIE, AESOD, AESTDTC, AETERM	*ATTACK
18	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1070	737		6 AESMIE, AESOD, AESTDTC, AETERM	*HAEMORRHAGE
19	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1070	738		5 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-01-22, HIP FRACTURE
20	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1083	747		1 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, Y, Y, N, Not in dataset, N, 2013-08-02, MYOCARDIAL INFARCTION
21	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1142	754		4 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-10-19, MYOCARDIAL INFARCTION
22	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1166	759		4 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-03-19, COMPLEX PARTIAL SEIZURES
23	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1166	763		5 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-03-20, SYNCOPSE
24	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1271	785		3 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-11-21, ATRIAL FIBRILLATION
25	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1271	786		4 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-11-22, CARDIAC FAILURE CONGESTIVE
26	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1271	787		1 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-11-21, DYSYPNOEA
27	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1271	788		5 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-11-22, HYPONATRAEMIA
28	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1271	789		2 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2012-11-21, MYOCARDIAL INFARCTION
29	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1368	816		1 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-12-25, HIP FRACTURE
30	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-710-1368	817		4 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-12-25, HIP FRACTURE
31	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-713-1141	868		4 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-07-01, DELIRIUM
32	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-713-1179	878		9 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-10-10, GASTROENTERITIS VIRAL
33	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-716-1151	991		1 AESMIE, AESOD, AESTDTC, AETERM	*HISTIOCYTOMA
34	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-716-1151	992		4 AESMIE, AESOD, AESTDTC, AETERM	*HISTIOCYTOMA
35	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-716-1189	1008		4 AESMIE, AESOD, AESTDTC, AETERM	*Y, N, N, N, N, N, Not in dataset, N, 2013-02-07, PROSTATE CANCER
36	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-718-1066	1103		1 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-07-16, SYNCOPSE
37	CORE-000022 Y, but AESER = N or empty.	*fully executable	AE	01-718-1066	1104		3 AESMIE, AESOD, AESTDTC, AETERM	*N, N, N, N, Y, N, Not in dataset, N, 2013-07-30, SYNCOPSE
38	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1015	1		1 AEBDSYCD, AEBODSYS	None, GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS
39	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1015	2		2 AEBDSYCD, AEBODSYS	None, GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS
40	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1015	3		3 AEBDSYCD, AEBODSYS	None, GASTROINTESTINAL DISORDERS
41	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1023	4		3 AEBDSYCD, AEBODSYS	None, CARDIAC DISORDERS
42	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1023	5		1 AEBDSYCD, AEBODSYS	None, SKIN AND SUBCUTANEOUS TISSUE DISORDERS
43	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1023	6		2 AEBDSYCD, AEBODSYS	None, SKIN AND SUBCUTANEOUS TISSUE DISORDERS
44	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1023	7		4 AEBDSYCD, AEBODSYS	None, SKIN AND SUBCUTANEOUS TISSUE DISORDERS
45	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1028	8		1 AEBDSYCD, AEBODSYS	None, GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS
46	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1028	9		2 AEBDSYCD, AEBODSYS	None, GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS
47	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1034	10		1 AEBDSYCD, AEBODSYS	None, GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS
48	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1034	11		2 AEBDSYCD, AEBODSYS	None, GENERAL DISORDERS AND ADMINISTRATION SITE CONDITIONS
49	CORE-000024 AEBODSYS is not empty and AEBDSYCD is empty	fully executable	AE	01-701-1047	12		4 AEBDSYCD, AEBODSYS	None, CARDIAC DISORDERS

CORE – FORMEDIX Report Viewer

Formedix CORE

xpt validation report created on 2023-05-22 at 05:19:23

Dataset	Rule ID	Error Message	# Issues
TV	CORE-000168	VISITNUM is not among VISITNUM in SV domain.	1
MH	CORE-000264	Primary analysis used but MHBODSYS and MHSOC are not equal	1818
LB	CORE-000168	VISITNUM is not among VISITNUM in SV domain.	20
AE	CORE-000022	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	36
AE	CORE-000024	AEBODSYS is not empty and AEBDSYCD is empty	1191
AE	CORE-000184	There is not a one-to-one relationship between AEBODSYS and AEBDSYCD	1
AE	CORE-000268	AEDECOD and AEPTCD do not have a one-to-one relationship	1191

Showing 1 - 7 of 7 rows

Technology Preview

CORE – FORMEDIX RECAP

Features

- > Load data from local XPT datasets
- > Select subset of datasets to validate
- > Select rule set
- > Select subset of rules to run
- > Refresh rule sets from CDISC Library
- > Sorting/filtering of results
- > View Excel results report

Benefits

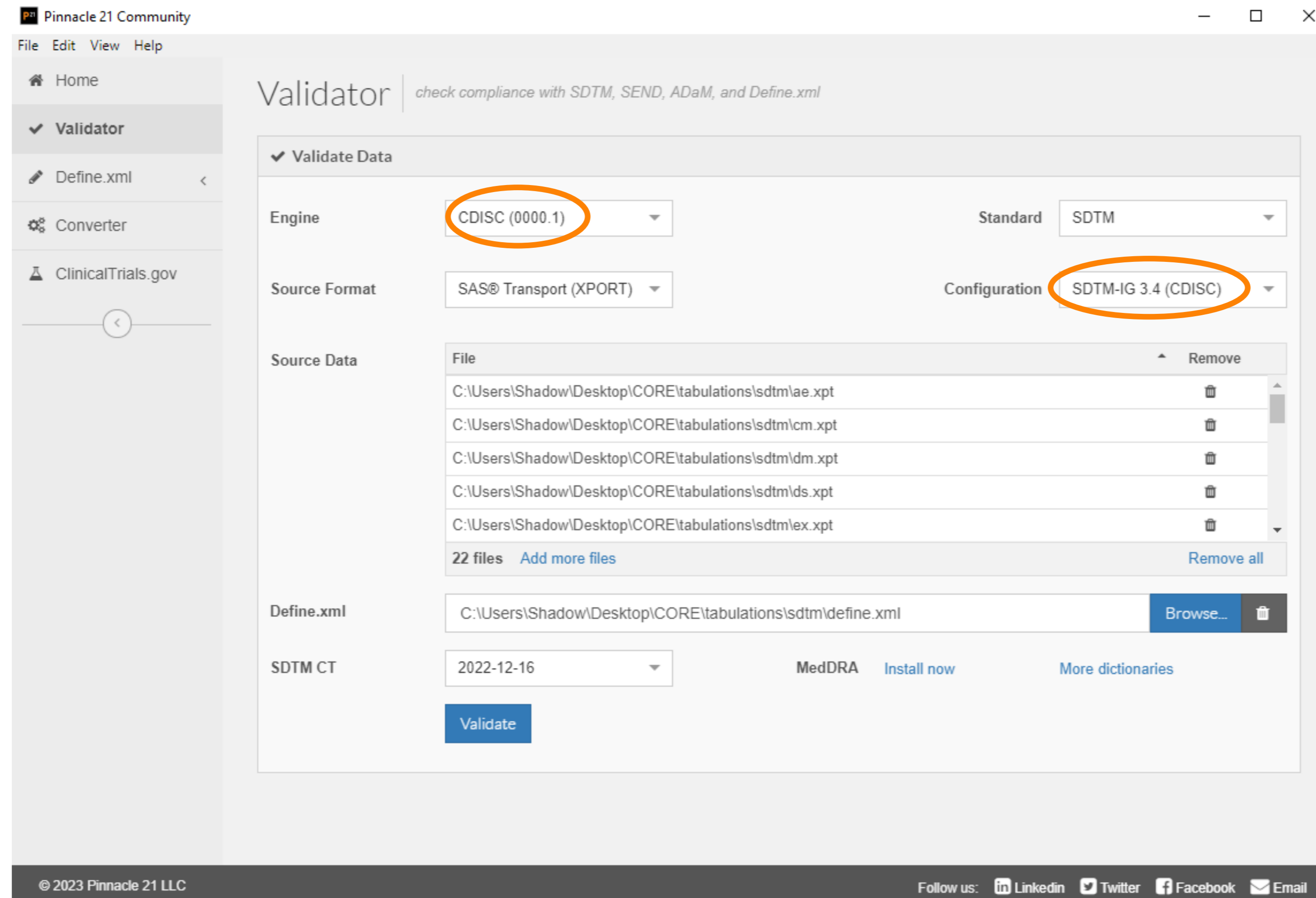
- > Early visibility of SDTM conformance
- > Early visibility of safety and efficacy data
- > Make informed decisions early on
- > Run more trials, focus on those showing promise
- > Ultimately, speed the delivery of life changing treatments

Future functionality

- > Final production version of CORE engine
- > Final production version of initial rulesets (e.g. SDTM-IG 3.4)
- > Additional rulesets (SEND, ADaM, FDA business rules etc)
- > Load data from CSV, Dataset-XML, Dataset-JSON
- > Validate against MedDRA / WHODrug and other dictionaries
- > (maybe) Linux / MacOS support

CORE – PINNACLE 21

P21 supports CDISC Open Rules Engine (CORE), enabling execution of machine-readable CDISC Conformance Rules via desktop GUI or CLI on any P21 Community platform.



CORE – PINNACLE 21 Report

Pinnacle 21 Validator Report			Variables	Values	Rule ID	Message	Category	Severity						
1	Pinnacle 21 Validator Report		AETERM, AESTDTC, AESER, AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD	PNEUMONIA, 2012-09-07, N, N, N, N, N, Y, N, N	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Functional Dependency	Error						
2			AETERM, AESTDTC, AESER, AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD	PNEUMONIA, 2012-09-07, N, N, N, N, N, Y, N, N	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Functional Dependency	Error						
3	Configuration: C:\Users\Shadow\Documents\Pinnacle 21 Community\configs\0000.1\SDTM-IG 3.4 (CDISC).xml		AETERM, AESTDTC, AESER, AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD	SUDDEN DEATH, 2013-01-14, N, N, N, N, N, Y, N, N	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Functional Dependency	Error						
4	Define.xml: C:\Users\Shadow\Desktop\CORE\tabulations\sdtml\define.xml		AETERM, AESTDTC, AESER, AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD	TRANSIENT ISCHAEMIC ATTACK, 2013-02-21, N, N, N, N, N, Y, N, N	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Functional Dependency	Error						
5	Generated: 2023-05-22T08:33:32-07:00		AETERM, AESTDTC, AESER, AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD	COMPLETED SUICIDE, 2014-10-31, N, N, N, N, N, Y, N, N	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Functional Dependency	Error						
6	CDISC SDTM CT Version: 2022-12-16		AETERM, AESTDTC, AESER, AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD	COLON CANCER, 2013-01-21, N, Y, N, N, N, Y, N, N	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Functional Dependency	Error						
7	UNII: 2023-04-13		<p style="text-align: center; color: red; font-weight: bold;">Problems with your Validator installation detected which may cause inaccurate validation results</p>											
8	MED-RT: 2023-05-03													
9	Validation Engine Name: CDISC 0000.1													
10	Software Version: 4.0.2													
11														
12														
13														
14														
15														
16														
<table border="1"> <thead> <tr> <th>Problem</th> <th>Details</th> <th>Link</th> </tr> </thead> <tbody> <tr> <td>Experimental Engine Used</td> <td>This experimental Engine is for developmental experimentation only. Use at your own risk. It has a limited Rule set and thus cannot detect the majority of data Issues ('false negatives'). No regulatory agency uses this Engine. For actual submissions, use a valid production Engine (FDA, PMDA, etc.). Refer to link for details.</td> <td>Experimental Engines Advisory</td> </tr> </tbody> </table>			Problem	Details	Link	Experimental Engine Used	This experimental Engine is for developmental experimentation only. Use at your own risk. It has a limited Rule set and thus cannot detect the majority of data Issues ('false negatives'). No regulatory agency uses this Engine. For actual submissions, use a valid production Engine (FDA, PMDA, etc.). Refer to link for details.	Experimental Engines Advisory						
Problem	Details	Link												
Experimental Engine Used	This experimental Engine is for developmental experimentation only. Use at your own risk. It has a limited Rule set and thus cannot detect the majority of data Issues ('false negatives'). No regulatory agency uses this Engine. For actual submissions, use a valid production Engine (FDA, PMDA, etc.). Refer to link for details.	Experimental Engines Advisory												
Pinnacle 21 Validator Report														
Issue Summary														
Source	Rule ID	Message	Severity	Found										
AE														
	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Error	36										
	CUSTOM.SDTMIG.CG00320	AEENDTC is empty	Error	473										
	CUSTOM.SDTMIG.CG00322	DOMAIN is not empty	Error	1191										
	CDISC.SDTMIG.CG0082	AEBODSYS is not empty and AEBDSYCD is empty	Warning	1191										
	CDISC.SDTMIG.CG0467	AESTDTC is present in a Findings general observation class	Warning	1										
CM														
	CDISC.SDTMIG.CG0032	Visit Day cannot be found in Trial Visit (TV) domain	Error	7510										
	CDISC.SDTMIG.CG0114	Missing value for CMDOSU, when CMDOSE, CMDOSTXT or CMDOSTOT is provided	Error	80										
	CUSTOM.SDTMIG.CG00322	DOMAIN is not empty	Error	7510										
	CDISC.SDTMIG.CG0467	CMSTDTC is present in a Findings general observation class	Warning	1										

CORE – PINNACLE 21 RECAP

Experimental, In-development Engine

- > CORE has a limited Rule set and thus cannot detect most data Issues. Expect "false negatives," meaning, your datasets will deceptively appear to have fewer Issues than they actually do.
- > More important, because no regulatory agency uses the CORE Engine, you should instead always use a valid production-grade Engine (FDA, PMDA, etc.) for actual submissions of real study data.

How to Run CORE Engine in P21 Community

- > Via the desktop GUI, go to Validator, set Engine to CDISC (0000.1) and choose a Configuration, e.g., SDTM-IG 3.4 (CDISC), which is the only one available from CDISC at this time. Drop your files into the Source Data box, and then Validate.
- > Via the CLI.

YAML

 **AliraHealth**



YAML – What is it?

YAML is a human-readable data serialization language that is often used for writing configuration files.

YAML is a popular programming language because it is designed to be easy to read and understand. It can also be used in conjunction with other programming languages. Because of its flexibility and accessibility.

The image shows the acronym 'YAML' in a bold, sans-serif font. The letters 'Y', 'M', and 'L' are black, while the letter 'A' is red. The letters are arranged in two rows: 'Y' and 'M' are on the top row, and 'A' and 'L' are on the bottom row. The 'A' is positioned between the 'Y' and 'M' of the top row and the 'L' of the bottom row.

YAML – SYNTAX

- > YAML files use a .yml or .yaml extension and follow a specific syntax rules.
- > YAML has features that come from Perl, C, XML, HTML, and other programming languages. YAML is also a superset of JSON, so JSON files are valid in YAML.
- > There are no usual format symbols, such as braces, square brackets, closing tags, or quotation marks.

```
#Comment: This is a supermarket list using YAML
#Note that - character represents the list
---
food:
  - vegetables: tomatoes #first list item
  - fruits: #second list item
    citrics: oranges
    tropical: bananas
    nuts: peanuts
    sweets: raisins
```

RULES



RULES - INTRODUCTION



Conformance Rule Specification Development

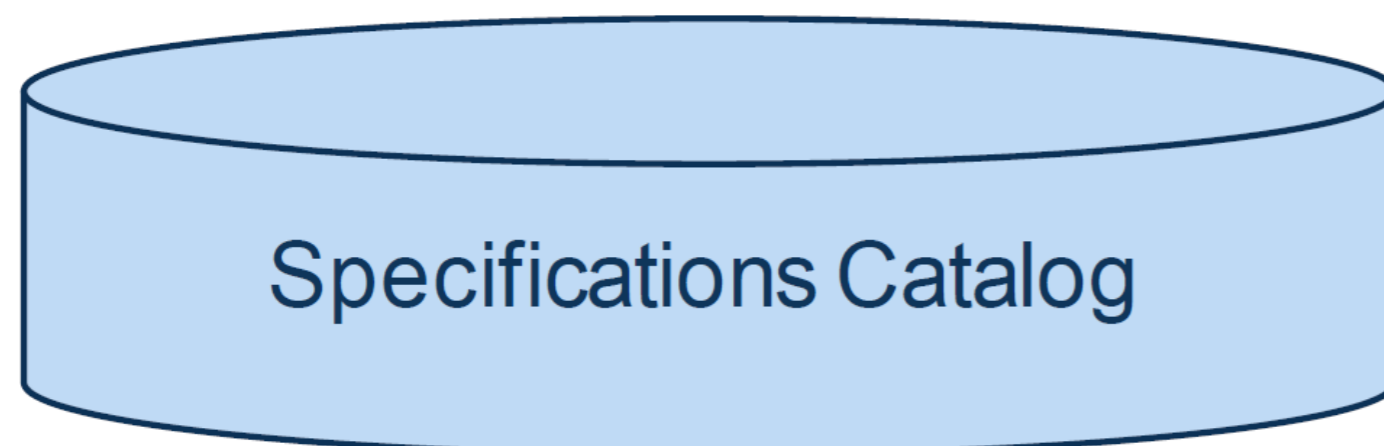
Human-readable Specification

Rule ID	SDTMIG Version	Rule Version	Class	Domain	Variable	Condition	Rule
CG0225	3.4	1	ALL	ALL	VISITDY	VISITNUM is NOT in <u>TV.VISITNUM</u>	VISITDY = null
Document	Section	Item	Cited Guidance				
IG v3.4	4.4.5		VISITDY must not be populated for unplanned visits, since VISITDY is, by definition, the planned study day of visit, and since the actual study day of an unplanned visit belongs in a --DY variable.				

Authoring Sources:

- CDISC Standards
- FDA Validation Rules
- Community proposals

Centralized



CORE Rule Development

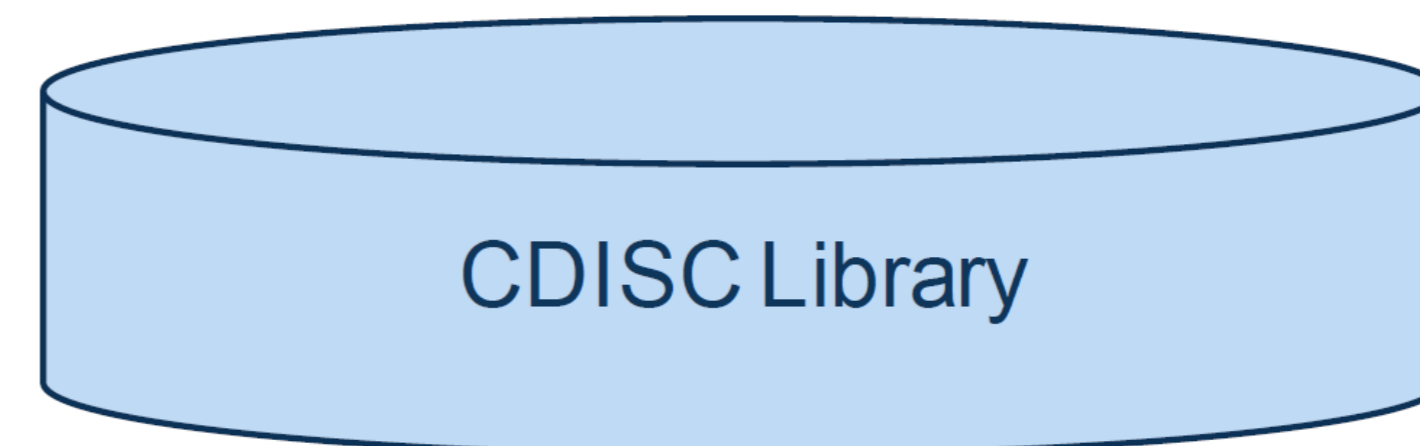
Executable Rule (YAML) in CORE Rule Editor

```

1 Core:
2   Id: CDISC_SDTMIG.CG0225
3   Version: '1'
4   Authority:
5   Organization: CDISC
6   Description: Trigger error if VISITDY is populated when VISITNUM is not in TV.
7   References:
8     - Origin: SDTM and SDTMIG Conformance Rules
9     Rule Identifier:
10      Id: CG0225
11      Version: '1'
12   Version: '2.0'
13   Sensitivity: Record
14   Severity: Warning
15   Rule Type: Value Presence
16   Scopes:
17     Classes:
18       Include:
19         All
20     Domains:
21       Include:
22         All
23     Standards:
24       - Name: SDTMIG
25         Version: '3.4'
26   Operations:
27     - domain: TV
28       Id: TV.VISITNUM
29       name: VISITNUM
30       operator: distinct
  
```

Rule developed and tested in CORE Rule Editor and CORE Engine

Publish



RULES – EXAMPLE: SDTM.CG0026.yml

```
Authority:
  Organization: CDISC
Check:
  all:
    - name: "--TPTREF"
      operator: empty
    - name: "--RFTDTC"
      operator: exists
    - name: "--RFTDTC"
      operator: non_empty
Citations:
  - Cited Guidance: "The actual date or date and time of a time point that acts as
    a fixed reference for a series of planned time points, represented in a standardized character format.
    |The fixed reference point is in --TPTREF"
    Document: Model v2.0
    Item: "--RFTDTC"
    Section: Timing
Core:
  Id: CDISC.SDTMIG.CG0026
  Version: '1'
Description: "Trigger error when --TPTREF = null and --RFTDTC is populated"
Outcome:
  Message: "--RFTDTC is populated when --TPTREF is null"
References:
  - Origin: SDTM and SDTMIG Conformance Rules
    Rule Identifier:
      Id: CG0026
      Version: '1'
      Version: '2.0'
Rule Type: Range & Limit
```

```
Scopes:
  Classes:
    Include:
      - All
  Domains:
    Include:
      - All
  Standards:
    - Name: SDTMIG
      Version: '3.4'
Sensitivity: Record
Severity: Error
```


RULES – EXAMPLE: CUSTOM RULE

```

SDTM.CG00320.yml x SDTM.CG00322.yml x SDTM.CG00321.yml x
1 Authority:
2   Organization: ALIRAHEALTH
3 Check:
4   all:
5     - name: DOMAIN
6     operator: empty
7 Citations:
8   - Cited Guidance: Value for End visit is empty
9     Document: IG v3.4
10    Item: ''
11    Section: 4.4.5
12 Core:
13   Id: CUSTOM.SDTMIG.CG00321
14   Version: '1'
15 Description: Verify that DOMAIN is empty
16 Outcome:
17   Message: DOMAIN is empty
18 Output Variables:
19   - "DOMAIN"
20 References:
21   - Origin: CUSTOM internal rule
22   Rule Identifier:
23     Id: CG00321
24     Version: '1'
25     Version: '2.0'
26 Rule Type: Presence
27 Scopes:
28   Classes:
29     Include:
30     - All
31   Domains:
32     Include:
33     - AE
34   Standards:
35     - Name: SDTMIG
36     Version: '3.4'
37 Sensitivity: Value
38 Severity: Error
  
```

Pinnacle 21 Validator Report

Issue Summary				
Source	Rule ID	Message	Severity	
AE				
	CDISC.SDTMIG.CG0041	At least one of the Seriousness criteria (AESCAN, AESCONG, AESDISAB, AESDTH, AESHOSP, AESLIFE, AESOD or AESMIE) = 'Y', but AESER = 'N' or empty.	Error	
	CUSTOM.SDTMIG.CG0032	AEENDTC is empty	Error	
	CUSTOM.SDTMIG.CG00322	DOMAIN is not empty	Error	
	CDISC.SDTMIG.CG0082	AEBODSYS is not empty and AEBDSYCD is empty	Warning	
	CDISC.SDTMIG.CG0467	AESTDTC is present in a Findings general observation class	Warning	
CM				
	CDISC.SDTMIG.CG0032	Visit Day cannot be found in Trial Visit (TV) domain	Error	
	CDISC.SDTMIG.CG0114	Missing value for CMDOSU, when CMDOSE, CMDOSTXT or CMDOSTOT is provided	Error	
	CUSTOM.SDTMIG.CG00322	DOMAIN is not empty	Error	
	CDISC.SDTMIG.CG0467	CMSTDTC is present in a Findings general observation class	Warning	
DM				
	CDISC.SDTMIG.CG0529	RFENDTC is missing when ARM is provided.	Error	
	CUSTOM.SDTMIG.CG0032	DOMAIN is not empty	Error	

CUSTOM.SDTMIG.CG00322	DOMAIN is not empty	Verify that DOMAIN is not empty	Presence	Error
-----------------------	---------------------	---------------------------------	----------	-------

Validation Summary | Dataset Summary | Issue Summary | Details | **Rules** (+)

CONCLUSION



Why CORE?



```
from cdisc_rules_engine.operation
from typing import List

class IsOdd(BaseOperation):
    def _execute_operation(self)
        """
        Returns True if the target
        """
        return self.params.data
```

```
authority:
  Organization: CDISC
check:
  all:
    - name: --BODSYS
      operator: non_empty
    - name: --BDSYCD
      operator: empty
citations:
  - Cited Guidance: Variable Qual
```



OPEN
SOURCE

CUSTOM
EXTENSION

CUSTOM
RULES

MULTIPLE
SOFTWARE

How to Volunteer

<https://www.cdisc.org/volunteer/form>

- > Select **CORE Rules** Team

- > Expected Engagement
 - Time Period: 3 - 6 months, or longer, if able
 - Hours per week: 2 - 4 hours, in addition to meeting attendance
 - Weekly Meetings and Workshops: Recorded and available for review
 - Rules Development Workshop Tuesdays 9am - 11am ET
 - Weekly Team Meeting – Thursdays 11am - 12pm ET



Homemade CORE



Why?

- > YAML Rule Handling: I wanted to understand how Python handles YAML rules and parsing.
- > CMD Environment: I aimed to create a command-line environment that allows users to conveniently check and validate their data.
- > Practical Application: The example serves as a demonstration of how Python can be used to apply YAML rules to a given dataset.
- > Information Sharing: Through this example, I aimed to provide insights and knowledge to those interested in YAML parsing in Python.



PACKAGES

- > Pandas
- > Yaml
- > Template
- > Argparse

```
1 import pandas as pd
2 import yaml
3 from jinja2 import Template
4 import argparse
```

PARSE COMMAND-LINE ARGUMENTS

```
6 # Parse command-line arguments
7 parser = argparse.ArgumentParser()
8 parser.add_argument('csv_path', help='Path csv dataset')
9 parser.add_argument('--output', help='Path to the output CSV file', default='output.csv')
10 args = parser.parse_args()
```

- > **parser = argparse.ArgumentParser():**
It creates an instance of the ArgumentParser class from the argparse module. The ArgumentParser class provides a way to specify the arguments that the program expects.
- > **parser.add_argument('name_parameter', help='Description_of_the_parameter'):**
This line adds a positional argument named 'name_parameter' to the argument parser. The help parameter is used to provide a description of the argument for the user.
- > **args = parser.parse_args():**
This line parses the command-line arguments provided by the user and stores the values in the args variable.

IMPORT

Terminal Local × + ▾

```
(venv) davidemarinucci@Davides-MBP Project % python main.py dataset.csv --output outputtest.csv
```

```
6 # Parse command-line arguments
7 parser = argparse.ArgumentParser()
8 parser.add_argument('csv_path', help='Path csv dataset')
9 parser.add_argument('--output', help='Path to the output CSV file', default='output.csv')
10 args = parser.parse_args()
```

```
12 # Import csv dataset
13 df = pd.read_csv(args.csv_path)
```

DEF: CHECK_CONDITIONS

- > **Check_conditions** takes two parameters:
 - Conditions;
 - Data.
- > FOR loop that iterates over each condition in the conditions list.
- > Inside the loop, the code extracts the values:
 - Field;
 - Operator;
 - Value from each condition.
- > If the operator is **'exists'**, it checks if the field exists in the data dictionary. If the field is not present in the data dictionary, the function returns False to indicate that the conditions are not met.
- > If the operator is **'less_than_or_equal'**, it uses a templating approach to evaluate the value. It renders the value by substituting variables from the data dictionary using a template engine. It then compares the rendered value with the corresponding value in the data dictionary for the given field. If the value in the data dictionary is greater than the rendered value, the function returns False.
- > If none of the conditions evaluated so far have returned False, the function reaches the end of the loop and returns True, indicating that all conditions have been met.

```
16 # Check if all conditions of the rule are satisfied
17 1 usage
17 def check_conditions(conditions, data):
18     for condition in conditions:
19         field = condition['field']
20         operator = condition['operator']
21         value = condition.get('value') # Value is optional for some operators
22
23         if operator == 'exists':
24             if field not in data:
25                 return False
26         elif operator == 'less_than_or_equal':
27             template = Template(value)
28             rendered_value = template.render(data)
29             if data[field] > rendered_value:
30                 return False
31         # Add more condition checks for other operators if needed
32
33     return True
```

RULE01.YAML

- > Name of the Rule: **StartDateCannotBeGreaterThanEndDate**
- > Conditions: specify the requirements that need to be met for the rule to be considered valid.
 - It checks the existence of the "start_date" field.
 - It checks the existence of the "end_date" field.
 - It compares the value of the "start_date" field with the "end_date" field using the "less_than_or_equal" operator.
- > Operator: less_than_or_equal: This line indicates the comparison operator used for the condition.
- > value: "{{ end_date }}": It specifies the value to compare against. It uses a template expression "{{ end_date }}" that will be rendered dynamically during runtime.
- > Message: Start date cannot be greater than end date. It defines the error message associated with this rule. If the conditions specified earlier are not met, this error message will be triggered.

```
1 - rule: StartDateCannotBeGreaterThanEndDate
2   conditions:
3     - field: start_date
4       operator: exists
5     - field: end_date
6       operator: exists
7     - field: start_date
8       operator: less_than_or_equal
9       value: "{{ end_date }}"
10  message: Start date cannot be greater than end date.
```

OPEN YAML RULE

This code reads the contents of a YAML file named "rule.yaml" and loads it into a Python data structure using the `yaml.safe_load()` function from the PyYAML library.

- > **With `open('rule.yaml')` as file:** This line opens the file named "rule.yaml" in the current directory using the `open()` function. It assigns the file object to the variable `file`.
- > **`Rules = yaml.safe_load(file)`:** This line uses the `yaml.safe_load()` function from the PyYAML library to parse the contents of the file. It takes the opened file object `file` as the argument and returns a Python data structure representing the YAML data. The parsed data is assigned to the variable `rules`.
- > **`Rules`:** it contains the Python data structure representing the YAML data from the "rule.yaml" file.

```
36 # Open YAML Rule
37 with open('rule.yaml') as file:
38     rules = yaml.safe_load(file)
```

APPLY RULES

This process a dataset represented by a DataFrame “df” and applies a set of rules to filter out observations that do not meet the conditions specified by the rules.

- > The code iterates over each observation in df using a for loop.
- > **Satisfies_rule**: It tracks whether the current observation satisfies any of the rules.
- > Nested loop: The second loop iterates over each rule in the rules list.
- > **Check_conditions**: it evaluates whether the current observation satisfies the conditions specified by the current rule.
- > If conditions satisfied for the current rule then **satisfies_rule = True** and breaks out of the nested loop.
- > After the nested loop, the code checks whether satisfies_rule is still False. If it is, it means that the current observation did not satisfy any of the rules. In this case, the observation is appended to the new_dataset list using new_dataset.append(observation).
- > **New_dataset** contain only those observations that did not satisfy any of the rules.

```
40 new_dataset = []
41 for observation in df:
42     satisfies_rule = False
43     for rule in rules:
44         if check_conditions(rule['conditions'], observation):
45             satisfies_rule = True
46             break
47     if not satisfies_rule:
48         new_dataset.append(observation)
```

EXPORT

- > Convert list (**df**) to dataframe.
- > Export as .csv

```
50 # Print the new dataset
51 df1 = pd.DataFrame(df)
52 df1.to_csv(args.output, index=False)
```

Example

```
☰ outputtest.csv ×
1 | start_date,end_date,other_field,num
2 | 2023-06-15,2023-06-13,value,1
3 | 2023-06-21,2023-06-20,value,2
4 |
```

EXAMPLE

Terminal

```
Terminal Local x + v
(venv) davidemarinucci@Davides-MBP Project % python main.py dataset.csv --output outputtest.csv
```

Output

```
outputtest.csv x
1 | start_date,end_date,other_field,num
2 | 2023-06-15,2023-06-13,value,1
3 | 2023-06-21,2023-06-20,value,2
4 |
```

`python main.py dataset.csv --output outputtest.csv`

CONCLUSION

PROs

- > We are able to produce a data check report in every stage of our study;
- > It gives us the opportunity to create our own custom rules;
- > It doesn't require an .xpt version of the dataset as per Pinnacle 21 in order to produce data checks;
- > Accessible through cmd;
- > Open source.

CONs

- > It requires YAML and PYTHON knowledge;
- > Deep dive into the definition of the rules;



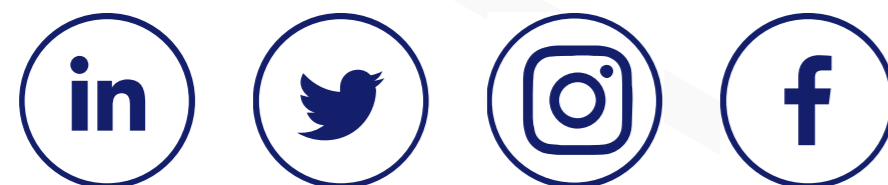
Thank You



LET'S KEEP IN TOUCH



davide.marinucci@alirahealth.com



NORTH AMERICAN OFFICES

Toronto | Canada
Boston | US
San Francisco | US

EUROPEAN OFFICES

Vienna | Austria
Paris, Bordeaux | France
Munich | Germany
Bologna, Milan, Verona | Italy

Zevenbergen | Netherlands
Barcelona | Spain
Basel | Switzerland
London | UK

ASIAN PACIFIC OFFICES

Sydney | Australia
Singapore