



2025 CDISC + TMF
EUROPE INTERCHANGE

GENEVA

CONFERENCE & EXPO: 14-15 MAY | TRAININGS: 12, 13, 16 MAY



FDA business rules and CDISC Open Rules, the road to adoption

Nick De Donder, CDISC Open Rules Program Manager, CDISC
Jozef Aerts, Owner, XML4Pharma



Meet the Speakers

Nick De Donder

Title: CDISC Open Rules Program Manager

Organization: CDISC

Nick has a diverse work experience in the field of life sciences and data standards. Nick started working at Business & Decision Life Sciences (now Enovalife) in 2008, where he held various roles. Recently he became an independent consultant. Since June 2021 Nick has been CDISC Open Rules Program Manager. Since 2017, Nick has been a CDISC Authorized Trainer and progressively took on roles such as E3C member, E3C Chair, and PHUSE Committee member.



Jozef Aerts

Title: Owner

Organization: XML4Pharma

Passionate about standards in clinical research and healthcare, and their implementation in IT systems.



Disclaimer and Disclosures

- The views and opinions expressed in this presentation are those of the author(s) and do not necessarily reflect the official policy or position of CDISC.*



Agenda

1. Research Collaboration Agreement
2. FDA Business Rules
3. Into Practice
4. Technical Implementation of CORE
5. Conclusions



FDA Business Rules



Background

- Sponsors should evaluate their study data before submission against the conformance rules published by an SDO, the eCTD Technical Rejection Criteria for Study Data, and the FDA Business Rules.
- The Business Rules v1.5 (May 2019) help ensure that the study data are compliant, useful, and will support meaningful review and analysis. This applies to SDTM formatted clinical studies and SEND formatted non-clinical studies. For more information, see Section 8 of the Technical Conformance Guide.
- All business rules should be followed where applicable.

Research Collaboration Agreement with US FDA CDER & CBER



CDISC and FDA working together to develop machine-executable formats of the FDA Business Rules and on the development and ongoing governance of this set of executable rules within the CDISC Open Rules project that can be used by industry



The benefits of creating a single source of truth for all FDA validation needs increases transparency for all stakeholders on how FDA checks data for conformance to CDISC standards and to existing FDA Business Rules

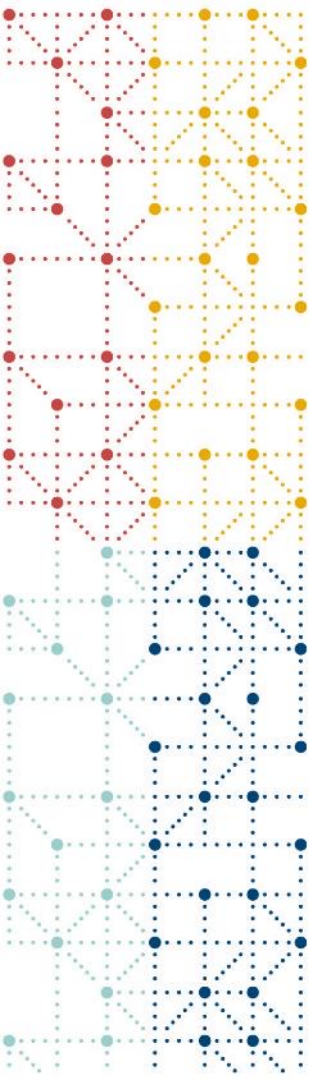
CDISC is Proud to Announce a Research Collaboration to Incorporate FDA Business Rules into CORE

Austin, TX – January 16, 2024 – CDISC is proud to announce a research collaboration with the U.S. Food and Drug Administration's Office of Translational Sciences in the Center for Drug Evaluation and Research and Office of Regulatory Operations in the Center for Biologics Evaluation and Research to incorporate FDA Business Rules into CDISC's Open Rules Engine (CORE).

CDISC's CORE project provides an open-source version of the CDISC Conformance Rules in a machine-executable format. These rules, published and managed by CDISC, create a single source for conformance rules and allow external vendors and sponsor companies to implement and extend these rules within their tools. [FDA Business Rules](#) are currently written in a plain text, non-machine executable format and describe the business requirements for regulatory review to help ensure that clinical trial study data is compliant and useful and supports meaningful review and analysis.

The goal of this effort, which began on November 3, 2023 and has term of three years, is to collaborate on providing input on machine-executable formats of the FDA Business Rules and on the development and ongoing governance of this set of executable rules within CORE that can be used by sponsors of medical product applications.

FDA Business Rule ID		FDA Business Rule
Clinical and Nonclinical		
DAB008		All treatment exposure date/time should be between first and last study treatment date/time.
DAB009		All paired variables should have a one-to-one relationship. Examples include short name and name of test; parameter name and parameter code or number; variable name and variable label, etc.
DAB011		All trial design data should be submitted as specified in the FDA Study Data Technical Conformance Guide (TCG).
DAB012		Deprecated in v1.5.
DAB013		Deprecated in v1.5.
DAB015		Character values should not have leading spaces or only have a period character.
DAB016		Collection study day should be populated when date/time of collection is available.
DAB017		Controlled terms should use the exact term (case, spelling, and punctuation) used by the terminology maintenance organizations (e.g., MedDRA, CDISC controlled terminology).
DAB018		A variable's length across a study should be no longer than the maximum length of the actual data (except for SUPPQUAL).
DAB019		SUPPQUAL variable length should be no longer than the maximum length of the actual data.



Into Practice

FDA Business Rules

Version 1.5 finalized June 2019

FDA Business Rule ID	FDA Business Rule
Clinical and Nonclinical (SDTMIG, SEND)	
FDAB008	All treatment exposure date/time should be between first and last study treatment date/time.
FDAB009	All paired variables should have a one-to-one relationship. Examples include short name and name of test; parameter name and parameter code or number; variable name and variable label, etc.
FDAB011	All trial design data should be submitted as specified in the FDA Study Data Technical Conformance Guide (TCG).
FDAB015	Character values should not have leading spaces or only have a period character.
FDAB016	Collection study day should be populated when date/time of collection is available.
FDAB017	Controlled terms should use the exact term (case, spelling, and punctuation) used by the terminology maintenance organizations (e.g., MedDRA, CDISC controlled terminology).
FDAB018	A variable's length across a study should be no longer than the maximum length of the actual data (except for SUPPQUAL).
FDAB019	SUPPQUAL variable length should be no longer than the maximum length of the actual data within the dataset.
FDAB024	Large datasets should be split into smaller datasets no larger than 5 GB in size.
FDAB026	Records with a baseline flag should have a corresponding standard result with a standardized unit where available.
FDAB030	Standard units should be consistent within the same assessment (having the same --TESTCD, --CAT, --SCAT, --SPEC, --METHOD values).

Identifiers				Scope of Rule					
Rule ID	Rule ID Version (represents any change to the rule)	Related Rule(s) (See Also, Compare Against)	Rule Set (Generally IG Version, OCCDS v1.0, ADNCA v1)	Class	Subclass	Dataset or Domain	Variable	Element	Scope Section
FB0801	1			INT		EC, EX (merged w/DM)	--STDTC, --ENDTC, RFXSTDTC, RFXENDTC	ItemGroup Def ItemDef	
FB0901	1		All SDTMIG, SENDIG	ALL		ALL	Variable name and variable label		
FB0902	1		All SDTMIG, SENDIG	FND, TDM		All	--TESTCD, --TEST		
FB0903	1		All SDTMIG, SENDIG	SPC, TDM		DM, TA, TV, TE	DM.ARM-DM.ARMCD, DM.ACTARM-DM.ACTARMCD, TA.ARM-TA.ARMCD, TA.ACTARM-TA.ACTARMCD, TV.ARM-TV.ARMCD, TV.ACTARM-TV.ACTARMCD, TE.ELEMENT-TE.ETCD		
FB0904	1		All SDTMIG, SENDIG	TDM		TS	TS.TSVAL-TS.TSVALCD-TS.TSVALNF		
FB0905	1		All SDTMIG	INT		AG, CM, SU	--CLAS, --CLASCD		
FB0910	1		All SDTMIG	TDM		TI	TI.IETEST, TI.IETESTCD		

Harmonized spreadsheet

Identifiers							Release Notes		
Rule ID ▼	Rule ID Version (represents any change to the rule) ▼	Related Rule(s) [See Also, Compare Against] ▼	Rule Set (Generally IG Version, OCCDS v1.0, ADNCA v1) ▼	Scope Section ▼	Rule Section ▼	Standards Section ▼	Guidance Section ▼	Release Notes ▼	Authoring section ▼

Business specification


Identifiers					Statement of Rule							
Rule ID	Rule ID Version (represents any change to the rule)	Related Rule(s) [See Also, Compare Against]	Rule Set (Generally IG Version, OCCDS v1.0, ADNCA v1)	Scope	Natural Language Rule (Success Criteria)	Rule (Success Criteria)	Condition (Success)	Natural Language Rule (Failure Criteria)	Rule (Failure Criteria)	Condition (Failure)	Error Message	Executable
FB0801	1				All treatment exposure date/time should be between first and last study treatment date/time.	(RFXSTDTCTC <= --STDTCTC <= --ENDTCTC <= RFXENDTCTC	Variables exists and are populated; otherwise, account for all possibilities and treat with ANY conditions. Note: need functionality 'check for RFX--DTC availability in IG' for generic rule implementation. Note: NA for earlier SENDIGs that don't have RFX--DTC.	All treatment exposure date/time outside first and last study treatment date/time.	not (RFXSTDTCTC <= --STDTCTC <= --ENDTCTC <= RFXENDTCTC	Variables exists and populated; otherwise, account for all possibilities and treat with ANY conditions.	All treatment exposure date/time outside first and last study treatment date/time.	Fully executable
FB0901	1		All SDTMIG, SENDIG			One-to-one		One-to-one mapping is not maintained.	Variable name and label pairing is not unique		One-to-one mapping is not maintained.	Fully executable
FB0902	1		All SDTMIG, SENDIG		TESTCD and TEST pair should have one-to-one relationship	One-to-one		One-to-one mapping is not maintained.	Pairing is not unique		One-to-one mapping is not maintained.	Fully executable
FB0903	1		All SDTMIG, SENDIG			One-to-one		One-to-one mapping is not maintained.	Pairing is not unique		One-to-one mapping is not maintained.	Fully executable
FB0904	1		All SDTMIG, SENDIG			One-to-one	TSVALCD ^= null	One-to-one mapping is not maintained.	Pairing is not unique		One-to-one mapping is not maintained.	Fully executable
FB0905	1		All SDTMIG			One-to-one		One-to-one mapping is not maintained.	Pairing is not unique		One-to-one mapping is not maintained.	Fully executable
FB0910	1		All SDTMIG			One-to-one		One-to-one mapping is not maintained.	Pairing is not unique		One-to-one mapping is not maintained.	Fully executable

Business rule to executability

FDAB036

The value for study day should not be negative for exposure treatments.

Identifiers					Statement of Rule							
Rule ID	Rule ID Version (represents any change to the rule)	Related Rule(s) (See Also, Compare Against)	Rule Set (Generally IG Version, OCCDS v1.0, ADNCA v1)	Scope Section	Natural Language Rule (Success Criteria)	Rule (Success Criteria)	Condition (Success)	Natural Language Rule (Failure Criteria)	Rule (Failure Criteria)	Condition (Failure)	Error Message	Executable
FB3601	1				The value for study day should not be negative for exposure treatments.	--STDY > 0 and --ENDY > 0		The value for study day is negative for exposure treatments.	--STDY < 0 or --ENDY < 0			Fully executable

 CORE Rules / CORERULES-9240
FB3601

[Edit](#) [Add comment](#) [Assign](#) [More](#) [Published](#)

Details

Type: [Review Comments](#) Resolution: Unresolved

Priority: To be assigned Fix Version/s: None

Affects Version/s: None

Component/s: FDA SDTMIG v3.2, FDA SDTMIG v3.3, FDA SDTMIG v3.4, FDA SENDIG DART v1.1, FDA SENDIG DART v1.2, FDA SENDIG GENETOX v1.0, FDA SENDIG v3.0, FDA SENDIG v3.1, FDA SENDIG v3.1.1, FDA SENDIG-AR v1.0 ...

Labels: None

Executability: Fully Executable

Description

Natural Language Rule (Success Criteria)	Rule (Success Criteria)	Condition (Success)
The value for study day should not be negative for exposure treatments.	--STDY > 0 and --ENDY > 0	

```
Check:
any:
  - name: --STDY
    operator: less_than
    value: 0
  - name: --ENDY
    operator: less_than
    value: 0
Core:
  Id: CORE-000516
  Status: Published
  Version: '1'
  Description: Study Day variables (--DY) value should not be negative in Exposure (EC/EX) datasets.
  Executability: Fully Executable
  Outcome:
    Message: Negative value of Study Day variable
    Output Variables:
      - --STDY
      - --ENDY
  Rule Type: Record Data
  Scope:
    Classes:
      Include:
        - INTERVENTIONS
    Domains:
      Include:
        - EC
        - EX
  Sensitivity: Record
```

Test data

- Positive test data does not show issues

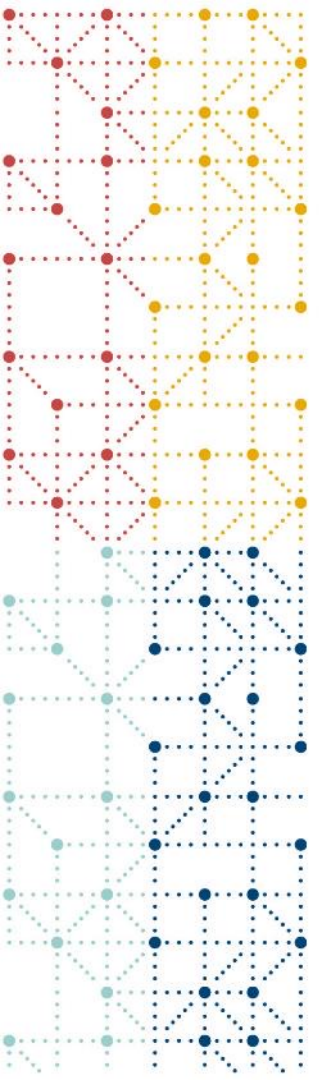
EPOCH	EXSTDTC	EXENDTC	EXSTDY	EXENDY
Epoch	Start Date/Time of Treatment	End Date/Time of Treatment	Study Day of Start of Treatment	Study Day of End of Treatment
Char	Char	Char	Num	Num
50	50	50	8	8
TREATMENT	2012-12-01	2012-12-01	2	2
TREATMENT	2012-12-02	2012-12-02	0	3
TREATMENT	2012-12-03	2012-12-03	4	4
TREATMENT	2012-11-30	2012-11-30	1	1

- Negative test data contains issues

EPOCH	EXSTDTC	EXENDTC	EXSTDY	EXENDY
Epoch	Start Date/Time of Treatment	End Date/Time of Treatment	Study Day of Start of Treatment	Study Day of End of Treatment
Char	Char	Char	Num	Num
50	50	50	8	8
TREATMENT	2012-12-01	2012-12-01	2	-2
TREATMENT	2012-12-02	2012-12-02	0	-3
TREATMENT	2012-12-03	2012-12-03	-4	-4
TREATMENT	2012-11-30	2012-11-30	-1	1

Executability

- Fully executable
- Partially executable
 - FDAB057: When collecting ethnicity demographic data from clinical trial participants, the following two minimum choices should be offered: "HISPANIC OR LATINO" or "NOT HISPANIC OR LATINO"
- Not executable
 - FDAB055: Trial participants should self-report race and ethnicity and they should not be assigned by the study team.



Technical Implementation of CORE

How can I run CORE or implement it in my own software?

CORE executables

- All CORE code can be obtained from the CORE Github site
 - CORE is mostly written in Python
- Executables (Windows, Mac, Ubuntu-Linux) available for each published version
- Quick start through a "Command Line Interface" (CLI)
- Excellent documentation through "-help" commands
- Always choice between running as Python or using the executables

▼ Assets 6

core-mac.zip

core-ubuntu-20-04.zip

core-ubuntu-latest.zip

core-windows.zip

Source code (zip)

Source code (tar.gz)

Using the Command Line Interface

- Use "cmd" or Powershell (Windows) to start the CLI
- Navigate to the folder where CORE was put

```
Eingabeaufforderung
Microsoft Windows [Version 10.0.19045.5737]
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\Jozef>d:
D:\>cd D:\CDISC-CORE-Engine\core-windows-0.0.2\core-windows\core
```

- When using Windows
- When using the

```
PS D:\PycharmProjects\cdisc-rules-engine> python core.py --help
Usage: core.py [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
  list-rules
  test
  update-cache
  validate               Validate data using CDISC Rules Engine
  version
```

Using the Command Line Interface: simple validation

- To learn about the validation parameters, use:

`".\core.exe validate --help"`

or

`"python core.py validate --help"`

- The most important ones are:

- `-s`: standard name, e.g. "sdtmig"
- `-v`: standard version, e.g. "3-4"
- `-d`: path where your XPT or Dataset-JSON files reside

Congratulations!
Your first CORE validation is running!

```
D:\CDISC_CORE_Engine\core-windows_0_9_3\core-windows\core>.\core.exe validate --help
Usage: core.exe validate [OPTIONS]
```

```
Validate data using CDISC Rules Engine
```

```
Example:
```

```
python core.py -s SDTM -v 3.4 -d /path/to/datasets
```

```
Options:
```

<code>-ca, --cache TEXT</code>	Relative path to cache files containing pre loaded metadata and rules
<code>-ps, --pool-size INTEGER</code>	Number of parallel processes for validation
<code>-d, --data TEXT</code>	Path to directory containing data files
<code>-dp, --dataset-path TEXT</code>	Absolute path to dataset file
<code>-l, --log-level [info debug error critical disabled warn]</code>	Sets log level for engine logs, logs are disabled by default
<code>-rt, --report-template TEXT</code>	File path of report template to use for excel output
<code>-s, --standard TEXT</code>	CDISC standard to validate against [required]
<code>-v, --version TEXT</code>	Standard version to validate against [required]
<code>-ss, --substandard TEXT</code>	CDISC Substandard to validate against
<code>-ct, --controlled-terminology-package TEXT</code>	Controlled terminology package to validate against, can provide more than one
<code>-o, --output TEXT</code>	Report output file destination
<code>-of, --output-format [XLSX JSON]</code>	Output file format
<code>-rr, --raw-report</code>	Report in a raw format as it is generated by the engine. This flag must be used only with --output-format JSON.
<code>-dv, --define-version [2-1 2-0 2.0 2.1]</code>	Define-XML version used for validation
<code>--whodrug TEXT</code>	Path to directory with WHODrug dictionary files
<code>--meddra TEXT</code>	Path to directory with MedDRA dictionary files
<code>--loinc TEXT</code>	Path to directory with LOINC dictionary files

Your first CORE validation

Windows PowerShell (x86)

```
PS D:\CDISC_CORE_Engine\core-windows_0_9_3\core-windows\core>  
>> .\core.exe validate -s sdtmig -v 3-4 -d D:\MetaDataSubmissionGuide_2_0_SASXPTFiles\  
[ ██████████ ] -----] 29%
```

This will generate a validation report in Excel format in the same folder as where you have the software.

You can use "-of JSON" to get the report in JSON format

And "-o ..." to get the report in another format

Many other options are available ... Use them!

Getting the rules themselves

- You can get the rules themselves and storing them to file, using:

`".\core.exe list-rules > rules.json"`

- This can be very interesting when developing a GUI on top of CORE

```
rules.json x
1  [
2    {
3      "core_id": "CORE-000351",
4      "author": "CDISC",
5      "reference": [
6        [
7          {
8            "Citations": [
9              {
10                "Cited_Guidance": "Identifier used to uniquely identify a subject a",
11                "Document": "IG v3.4",
12                "Item": "Specification",
13                "Section": "5.2"
14              }
15            ],
16            "Origin": "SDTM and SDTMIG Conformance Rules",
17            "Rule_Identifier": {
18              "Id": "CG0151",
19              "Version": "1"
20            },
21            "Version": "2.0"
22          }
23        ],
24        [
25          {
26            "Citations": [
27              {
28                "Cited_Guidance": "Identifier used to uniquely identify a subject a",
29                "Document": "IG v3.2",
30                "Item": "Specification",
31                "Section": "5"
32              }
33            ],
34            "Origin": "SDTM and SDTMIG Conformance Rules",
35            "Rule_Identifier": {
36              "Id": "CG0151",
37              "Version": "1"
```

Implementing CORE in your own software

- Lots of background information on the LinkedIn "CDISC CORE Collaboration" forum
<https://www.linkedin.com/groups/12986635/>



Anthony Chow • 1st

#加油油 #FuelUp Career geek specializes in data standards, data modeling, and ...
5d

Want to run CORE but can't execute the CLI in your SAS environment?

You are in for a treat! Running CORE as Python in a Base SAS session is possible! Check out this proof-of-concept implementation using Python 3.10 and CORE 0.9.3 by Lex Jansen.



Lex Jansen • 1st

Senior Director, Data Science Development at CDISC
5d • Edited •

Want to run CDISC CORE rules in SAS, but you do not have access to operating system commands (CLI)?
During the upcoming #PharmaSUG2025 conference I will be ...more

lexjansen/cdisc-core-sas



GitHub - lexjansen/cdisc-core-sas: Proof of Concept for running CDISC CORE within SAS

github.com



Samuel Johnson • 1st

Software Engineer @ CDISC
6d

💡 Did you know?

The CDISC CORE Rules Engine is available as a Python package on PyPI — giving you another option to integrate conformance checks into your clinical data [pipeline.In](#) addition to CLI, executable, and Docker options, you can now:

✳ Import CORE directly into your workflows

📁 Validate against data in dataframe objects

⚙ Automate standards checks inside existing systems

📖 Quick start guide: https://lnkd.in/eh_nUXY7

#CDISC #CORE #PyPI #ClinicalData #DataValidation #SDTM #OpenSource #Python

👤 Nick De Donder and 5 others





Implementing CORE in your own software

Many options ...


- Docker (see poster Sam Johnson)
- In existing Python applications (easy ...)
- SAS: see Lex's post on LinkedIn and: <https://github.com/lexjansen/cdisc-core-sas>
- R: one can use the "reticulate" package
- Java / C# / ...:
Many software languages have a mechanism to start an external process in a separate thread.

Implementing CORE in your own software - example

- One can use the generated "rules.json" file to allow selection of (groups) of rules to be applied on all, or a selection of, the XPT or Dataset-JSON files
- We have implemented this in several of our software offerings
- Essentially, the only limitation is ... lack of imagination

Implementing CORE in your own software - example

Message ×

 Folder with SAS-XPT files: D:\temp

SAS-XPT files to be validated:

- ☒ dm.xpt
- ☒ lbbl.xpt
- ☒ lbch.xpt
- ☒ vs.xpt

Rules used for Validation:

- ☒ CORE-000351
- ☒ CORE-000252
- ☒ CORE-000094
- ☒ CORE-000012
- ☒ CORE-000358
- ☒ CORE-000672
- ☒ CORE-000201
- ☒ CORE-000236
- ☒ CORE-000370
- ☒ CORE-000087
- ☒ CORE-000025
- ☒ CORE-000156

☐ Select All

☐ Select None

☐ Select All

☐ Select None

[Explore Validation Rules](#)

Report format: ☒ Excel ☐ JSON

Report Folder: D:\temp [Browse...](#)

[Execute CORE Validation](#)

[OK](#)

Implementing CORE in your own software:

Rules filtering

Core Rules Exploration

CDISC
FDA

Filter Reset Select all rules shown
Clear selection

☐ Rule ID ☐ Sensitivity ☒ Authorities ☐ Standards and Versions ☐ Classes Included ☐ Classes Excluded
☐ Domains Included ☐ Domains Excluded ☐ Rule Description ☐ Message ☐ References

Include	Rule ID	Authority	Sensitivity	Standards and Versions	Classes Included	Classes Exclud.	Domains Includ.	Domains Exclu...	Description	Message	References
<input type="checkbox"/>	CORE-000252	CDISC - FDA	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.2	SPECIAL PURPOSE		DM		When DS.DSDECOD is 'DEATH', DTHFL must be 'Y'	DTHFL is not 'Y' when...	FDA Busin...
<input checked="" type="checkbox"/>	CORE-000672	FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	FINDINGS		ALL			The value of --STNRH...	FDA Busin...
<input type="checkbox"/>	CORE-000305	FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	ALL						
<input checked="" type="checkbox"/>	CORE-000689	CDISC - FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.2 SDTMIG 3.3	ALL						
<input type="checkbox"/>	CORE-000522	FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	EVENT						
<input type="checkbox"/>	CORE-000517	FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	INTER						
<input checked="" type="checkbox"/>	CORE-000699	FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	FINDING FINDING						
<input type="checkbox"/>	CORE-000321	FDA	Dataset	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	ALL						
<input type="checkbox"/>	CORE-000253	CDISC - FDA	Record	SDTMIG 3.4 SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.2	SPECIAL PURPOSE						
<input type="checkbox"/>	CORE-000642	FDA	Record	SDTMIG 3.2 SDTMIG 3.3 SDTMIG 3.4	INTERVENTIONS EVENTS FINDINGS		ALL		--TPTREF must be populated when --ENINT is populated	--TPTREF must be po...	FDA Busin...

OK

Displays only the FDA rules
and specifically selects the FDA business rules:
FDAB039 (--STNRHI >= --STNRLO)
FDAB009 (paired variables)
FDAB030 (standard units consistency)

Only these rules will be executed

Adding your

- Many components
- With CORE,

```
STUDYID_MY.yaml
1 Authorities:
2   - Organization: XML4Pha
3     Standards:
4       - Name: SDTMIG
5         References:
6           - Citations:
7             - Cited Guid
8               Document:
9                 Version: '3.4'
10 Check:
11   all:
12     - name: 'STUDYID'
13       operator: does_not_
14       value: 'XML4Ph'
15 Core:
16   Status: Draft
17   Version: '1'
18   Description: 'The value of
19   Executability: Fully Exec
20 Outcome:
21   Message: 'The value of
22   Rule Type: Record Data
23 Scope:
24   Classes:
25     Include:
26       - ALL
27   Domains:
28     Include:
29       - ALL
30 Sensitivity: Record
31
```

```
Check:
  all:
    # throw an error when AGE (in years) is higher than 65
    - name: AGEU
      operator: equal_to
      value: YEARS
    - name: AGE
      operator: greater_than
      value: 65

Core:
  Id: XML4P1
  Status: Draft
  Version: '1'
  Description: 'Age in years must be 65 or lower'
  Executability: Fully Executable

Outcome:
  Message: 'Age in years must be 65 or lower'
  Rule Type: Record Data

Scope:
  Classes:
    Include:
      - 'SPECIAL PURPOSE'
  Domains:
    Include:
      - DM
  Sensitivity: Record
```

ility assurance

YAML or JSON
er with your own

parameter to point
with your own rules

s sdtmig -v 3-4
s
XML4Pharma

Adding your own (QA) rules to CORE: Example Result

CORE-Report-2025-04-2...									
A	B	C	D	E	F	G	H		
CORE-ID	Message	Executability	Dataset	USUBJID	Record	Sequence	Variable(s)	Value(s)	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC001		1	1 STUDYID	CDISCPIL0T01	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC001		2	2 STUDYID	CDISCPIL0T01	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC002		3	1 STUDYID	CDISCPIL0T01	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC002		4	2 STUDYID	CDISCPIL0T01	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC002		5	3 STUDYID	CDISCPIL0T01	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC002		6	4 STUDYID	CDISCPIL0T01	
	The value of the STUDYID must contain or start with XML4Ph	fully executable	ae.xpt	CDISC002		7	5 STUDYID	CDISCPIL0T01	

CORE-Report-2025-04-2...									
A	B	C	D	E	F				
CORE-ID	Version	CDISC RuleID	FDA RuleID	Message	Status				
	1			The value of the STUDYID must contain or start with XML4Ph	SUCCESS				

One can also add the own rules to the "cache", i.e. add them to the CDISC/FDA rules

Automating the generation of custom rules

- Generation of custom rules can easily be automated, e.g. for:

- CDISC Code Tables
- Biomedical Concepts
- LOINC to CDISC Mappings (>10,000 available)

- Examples:

- When VSTESTCD=SYSBP,DIABP, VSORRSU must be one of mmHg, cmHg, Pa
- When LBLOINC=1751-7, LBTESTCD must be "ALB", LBTEST must be "Albumin", LBSPEC must be "SERUM" or "PLASMA"

```
4  [ ] Check:
5  [ ] all:
6  [ ] - name: FATESTCD
7  [ ]   operator: equal_to
8  [ ]   value: 'ACMITYPE'
9  [ ]
10 [ ] - name: FASTRESC
11 [ ]   operator: is_not_contained_by
12 [ ]   value:
13 [ ]     - 'TYPE 1 MYOCARDIAL INFARCTION'
14 [ ]     - 'TYPE 2 MYOCARDIAL INFARCTION'
15 [ ]     - 'TYPE 3 MYOCARDIAL INFARCTION'
16 [ ]     - 'TYPE 4A MYOCARDIAL INFARCTION'
17 [ ]     - 'TYPE 4B MYOCARDIAL INFARCTION'
18 [ ]     - 'TYPE 4C MYOCARDIAL INFARCTION'
19 [ ]     - 'TYPE 5 MYOCARDIAL INFARCTION'
20 [ ]
```



Thank You!

ndedonder@cdisc.org

jozef.aerts@xml4pharma.com

