cdisc®

2026 Europe Interchange

THE FUTURE IS CONNECTED: STANDARDS AND AI POWERING DIGITAL TRANSFORMATION

JSONata: Unlocking Next-Generation CDISC Open Rules

Roman Radelicki, Head Data Technology, SGS Pharma Clinical Research

Speaker

Roman Radelicki started his career as Programmer in 2006 and joined SGS in 2009. During his career at SGS he held several positions and became Head Data Technology in 2021. He is mainly responsible for managing the data engineering and application development teams, providing support to the different departments

Roman Radelicki
Head Data Technology
SGS Pharma - Clinical Research

Agenda

1. CDISC Open Rules
2. CDISC Ecosystem
3. JSONata
4. Use Cases
5. Opportunity for AI
6. Conclusion

CDISC Open Rules

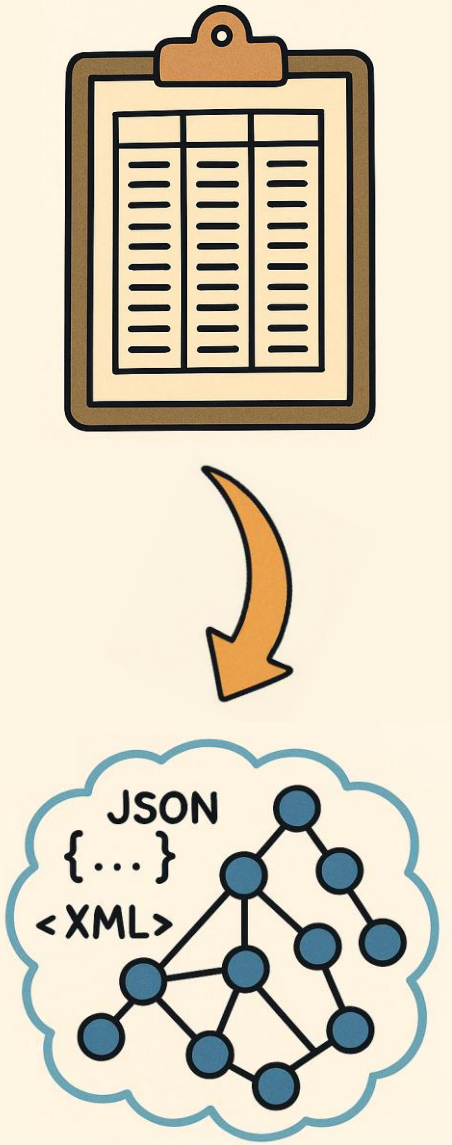
CDISC Open Rules

- Submission Ready
- CDISC Governed – Single source of truth
- Community Driven
- Executable Rules
- Open-Source

CDISC Ecosystem

CDISC Ecosystem

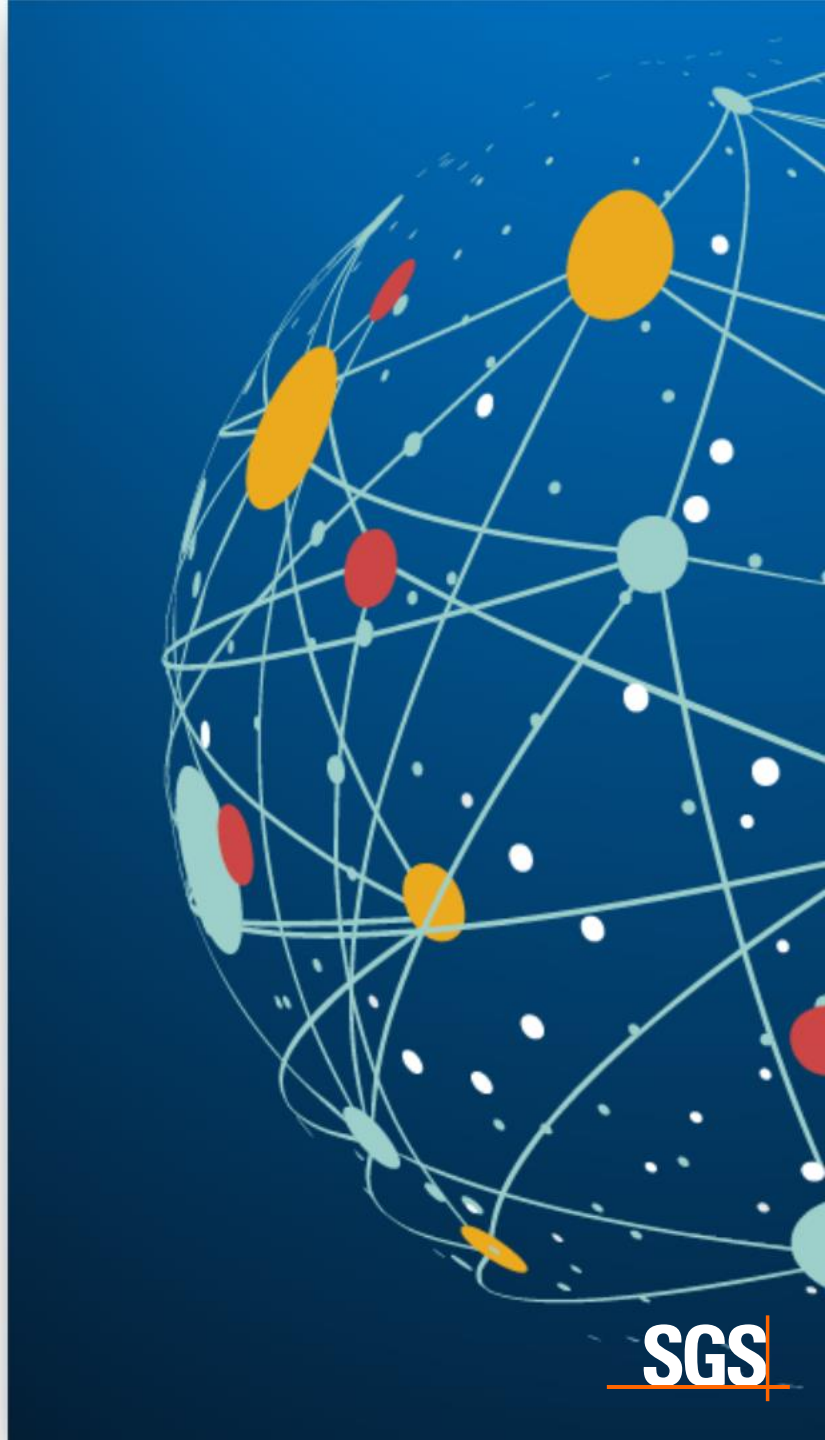
- Evolution in CDISC data structures
  - ODM
  - Define-XML
  - Dataset-JSON
  - (Meta)data increasingly distributed by JSON
    - E.g. CDISC library APIs
    - USDM
- As standards evolve, validation logic must evolve
  - Tabular rule engine insufficient



JSONata

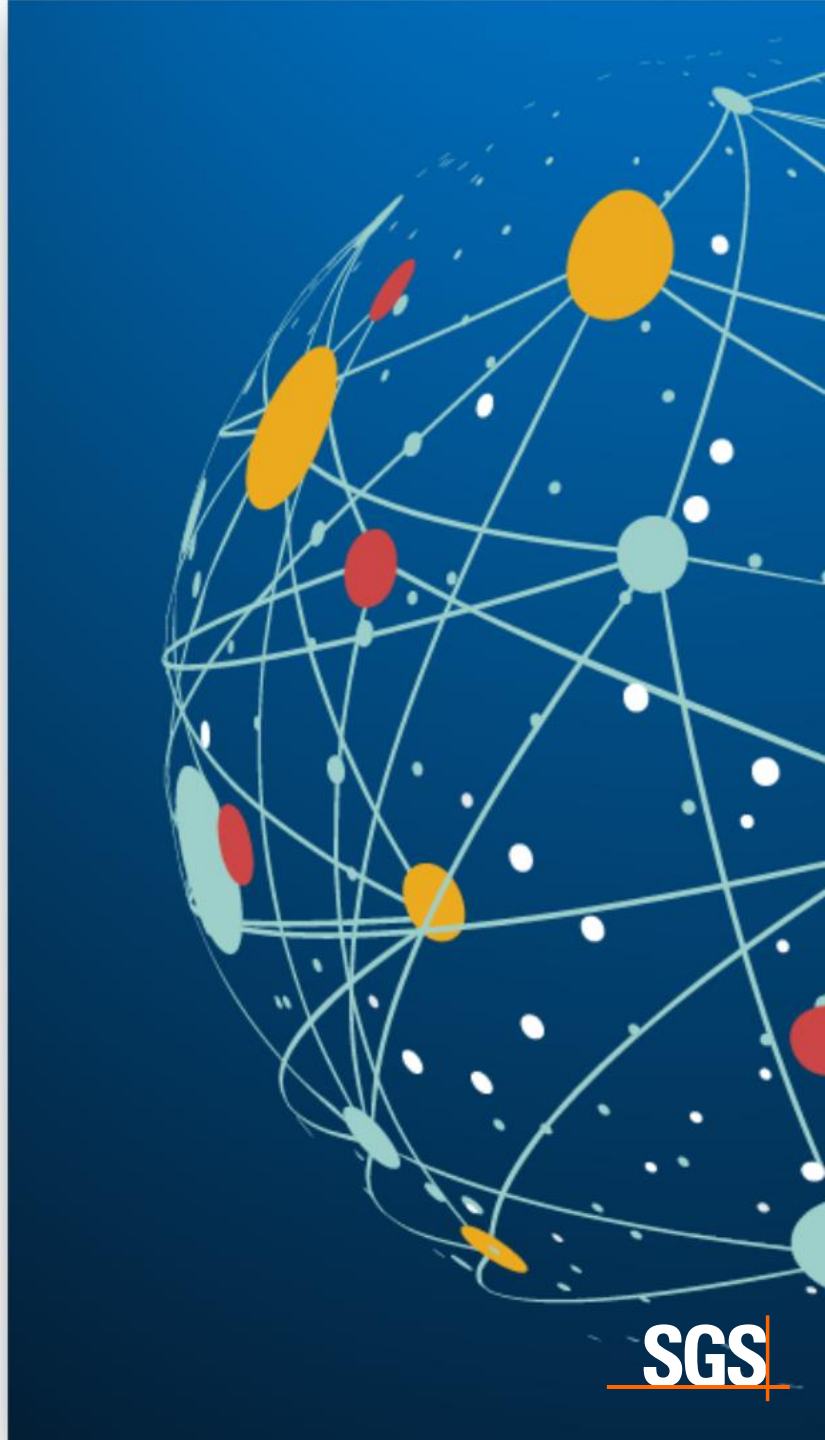
YAML

- Rules expressed in YAML
  - Non-proprietary language
  - No programming language knowledge required
  - Limited by the operators programmed for CORE
  - Mainly focused on tabular structures e.g. XPT datasets



JSONata

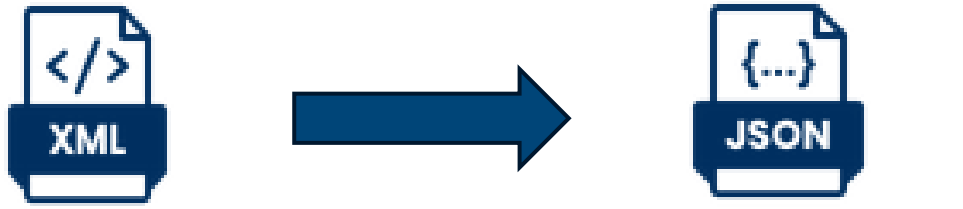
- Lightweight, declarative and readable query and transformation language
  - Built to navigate nested, object-based structures efficiently
  - Powerful operators, functions and customizable expressions
- Open source – MIT License
- https://try.jsonata.org/



Use Cases

Define-XML

- JSONata applied to Define-XML
  - XML is first transformed to a JSON structure
  - Github project: odmlib from Sam Hume https://github.com/swhume/odmlib



define.xml define.json

Define-JSON

For controlled terminologies provided by third parties, an **ExternalCodeList** element must be provided to identify the name and version of the terminology

5.3.13.4 ExternalCodeList Element

The ExternalCodeList element identifies the source of a third-party controlled terminology.

Element Name	ExternalCodeList
Element XPath	/ODM/Study/MetaDataVersion/CodeList/ExternalCodeList
Element Textual Value	None
Usage	• Requirement: Conditional. For controlled terminologies provided by third parties, an ExternalCodeList element must be provided to identify the name and version of the terminology. • Cardinality: One. Each CodeList element must contain either 1 or more EnumeratedItem elements, 1 or more CodeListItem elements, or 1 ExternalCodelist element. • Business Rule: Required for regulatory submissions to the FDA to provide the reference to the medical dictionaries used.
Attributes	Dictionary, Version, ref, href
Child Elements	None

Attribute	Usage	Allowable Values	Description
Dictionary	Required	Text CDISC Submission Value (=CodedValue) as listed in the NCI/CDISC Controlled Terminology, CodeList Dictionary Name, NCI Code C66788. This CodeList is extensible. Samples: COSTART, ICD, LOINC, MedDRA, SNOMED, WHOART, WHODD	The name of the external codelist
Version	Required	Text	The version designator of the external codelist
ref	Optional	Text	Reference to a local instance of the dictionary
href	Optional	Text	URL of an external instance of the dictionary

Define-JSON

```
<ItemDef OID="IT.AE.AEDECOD" Name="AEDECOD" DataType="text" Length="34" SASFieldName="AEDECOD">
  <Description>
    <TranslatedText xml:lang="en">Dictionary-Derived Term</TranslatedText>
  </Description>
  <CodeListRef CodeListOID="CL.MEDDRA" />
  <def:Origin Type="Assigned" Source="Sponsor" />
</ItemDef>
```

```
<CodeList OID="CL.MEDDRA" Name="Medical Dictionary for Regulatory Activities" DataType="text">
  <ExternalCodeList Dictionary="MedDRA" Version="27.0" />
</CodeList>
```

Introduce an error: removal of the version

```
<CodeList OID="CL.MEDDRA" Name="Medical Dictionary for Regulatory Activities" DataType="text">
  <ExternalCodeList Dictionary="MedDRA" />
</CodeList>
```

Define-JSON

Transform define.xml to define.json

```xml
<CodeList OID="CL.MEDDRA" Name="Medical Dictionary for Regulatory Activities" DataType="text">
  <ExternalCodeList Dictionary="MedDRA" />
</CodeList>
```

```json
{
    "OID": "CL.MEDDRA",
    "Name": "Medical Dictionary for Regulatory Activities",
    "DataType": "text",
    "ExternalCodeList": {
        "Dictionary": "MedDRA"
    }
}
```

Define-JSON

```
{
    "OID": "CL.MEDDRA",
    "Name": "Medical Dictionary for Regulatory Activities",
    "DataType": "text",
    "ExternalCodeList": {
        "Dictionary": "MedDRA"
    }
}
```

JSONata code

Element XPath	/ODM/Study/MetaDataVersion/CodeList/ExternalCodeList

```
Study.MetaDataVersion.CodeList[
  ExternalCodeList and $not($exists(ExternalCodeList.Version))
].{
  "OID": OID,
  "Name": Name,
  "Dictionary": ExternalCodeList.Dictionary,
  "Version": ExternalCodeList.Version
}
```

```
{
  "OID": "CL.MEDDRA",
  "Name": "Medical Dictionary for Regulatory Activities",
  "Dictionary": "MedDRA"
}
```

cdisc

SGS

Annotated eCRF

ODM.xml stylesheet.xls

[GitHub - lexjansen/cdisc360i-pocs: CDISC 360i POCs · GitHub]

Study: CORE-001

Protocol: CORE-RULES

Description: A Study to Evaluate the Effectiveness of CORE Rules on Annotated ODM.xml

Vital Signs

DOMAIN = VS

VS - Performed

Were vital signs performed?	○ No ○ Yes	VSSTAT = NOT DONE when VSTESTCD = VSALL; [NOT SUBMITTED] ***CodeList:*** *No Yes Response Only*
What was the date of the measurement(s)?	dd/mm/yyyy	VSDTC

* Mandatory field

VS - Vital Signs

What was the result of the height measurement?		VSORRES when VSTESTCD = HEIGHT
What was the unit of the height measurement?	○ Centimeter ○ Inch	VSORRESU when VSTESTCD = HEIGHT ***CodeList:*** *Units for Vital Signs Results Height*
What was the result of the weight measurement?		VSORRES when VSTESTCD = WEIGHT
What was the unit of the weight measurement?	○ Kilogram ○ Pound	VSORRESU when VSTESTCD = WEIGHT ***CodeList:*** *Units for Vital Signs Results Weight*
What was the result of the temperature measurement?		VSORRES when VSTESTCD = TEMP
What was the unit of the Temperature Unit measurement?	○ Celsius ◉ Fahrenheit	VSORRESU when VSTESTCD = TEMP ***CodeList:*** *Units for Vital Signs Results Tempature*
What was the anatomical location where the temperature was taken?	-- Select --	VSLOC when VSTESTCD = TEMP ***CodeList:*** *Anatomical Location Temperature*

* Mandatory field

ODM annotations

ODM.xml → ODM.json

- Create JSONata rule that checks if all items on eCRF are annotated

VS - Vital Signs

What was the result of the height measurement?		VSORRES when VSTESTCD = HEIGHT

```
<ItemDef OID="IT.HEIGHT_VSORRES" Name="HEIGHT_VSORRES" DataType="text" Length="3">
    <Description>
        <TranslatedText xml:lang="en">[VSTESTCD]_VSORRES</TranslatedText>
    </Description>
    <Question>
        <TranslatedText>What was the result of the height measurement?</TranslatedText>
        <TranslatedText xml:lang="en">Height</TranslatedText>
    </Question>
    <Alias Name="VSORRES when VSTESTCD = HEIGHT" Context="SDTM"/>
</ItemDef>
```

ODM annotations

```
<ItemDef OID="IT.HEIGHT_VSORRES" Name="HEIGHT_VSORRES" DataType="text" Length="3">
    <Description>
        <TranslatedText xml:lang="en">[VSTESTCD]_VSORRES</TranslatedText>
    </Description>
    <Question>
        <TranslatedText>What was the result of the height measurement?</TranslatedText>
        <TranslatedText xml:lang="en">Height</TranslatedText>
    </Question>
    <Alias Name="VSORRES when VSTESTCD = HEIGHT" Context="SDTM"/>
</ItemDef>
```

REMOVED

- Create JSONata rule that checks if all items on eCRF are annotated

```
Study.MetaDataVersion.ItemDef[
    $not($exists(Alias[Context="SDTM"]))].{
        "OID": OID,
        "Question": Question.TranslatedText._content
    }
```

```
{
  "OID": "IT.HEIGHT_VSORRES",
  "Question": [
    "What was the result of the height measurement?",
    "Height"
  ]
}
```

Opportunity for AI

Opportunity for AI

AI-assisted Rule Authoring

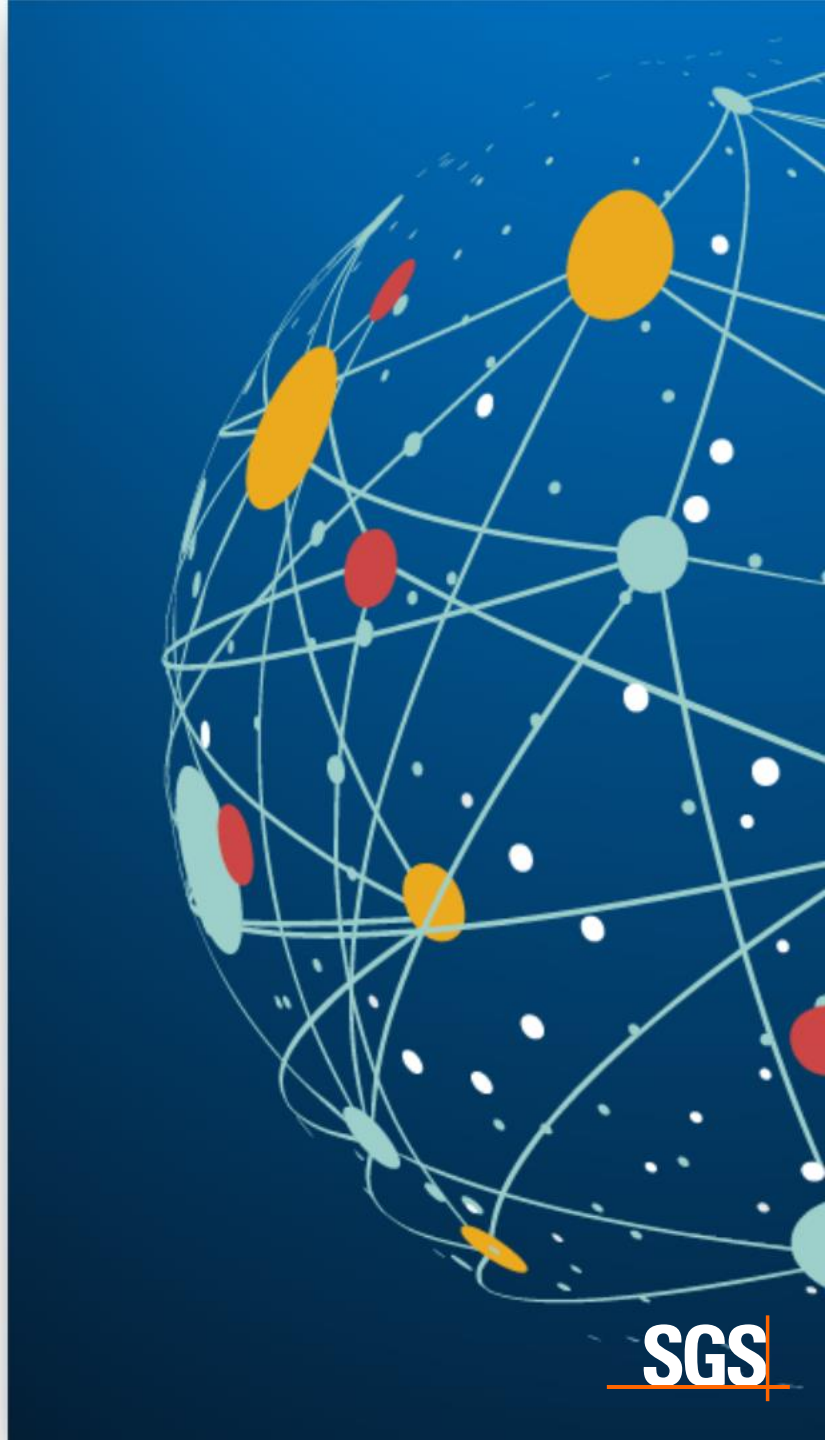
- YAML rule and test data generation (CDISC EU 2025)

Next Step

- Extend AI support to JSONata rule creation

Vision

- Agentic architecture for AI-assisted rule authoring



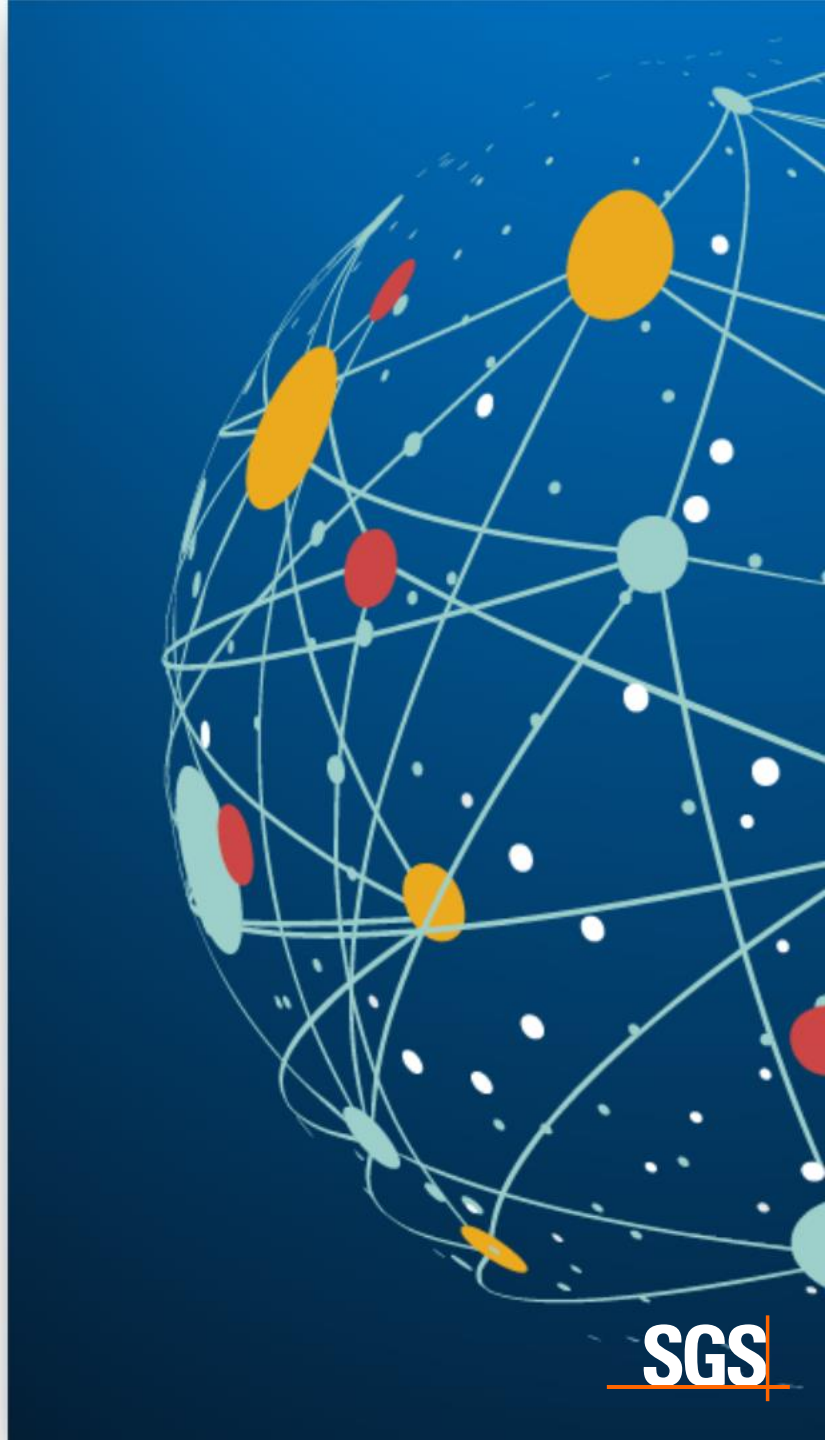
Conclusion

Conclusion

- JSONata makes CORE ready for current and future standards

- Only minor engine updates needed
  - Define-JSON / ODM.json support
  - XML → JSON transformation

- JSONata complements YAML
  - Handles complex, nested structures
  - More flexible rules, fewer constraints for rule authors



Thank You!

Contact us

clinicalresearch@sgs.com

www.sgs.com/pharma

sgs.com/linkedinpharma

Visit us at the SGS Booth