

ISA² Action 2016.07: SEMIC Promoting semantic interoperabilities amongst the EU Member States

SHACL template files Webinar

Brussels, 26/06/2020

Speaker:
Bert Van Nuffelen



Webinar practicalities



Click on 'connect audio' but please mute your microphones



You can also share your questions for the Q&A sessions via the chat*

* One question after each speaker + Q&A discussion at the end



The webinar will be recorded

Table of contents

- Introduction, Tour de table & objectives  15'
- SHACL templates discussion points  1h20
- Wrap-up session & closing  10'

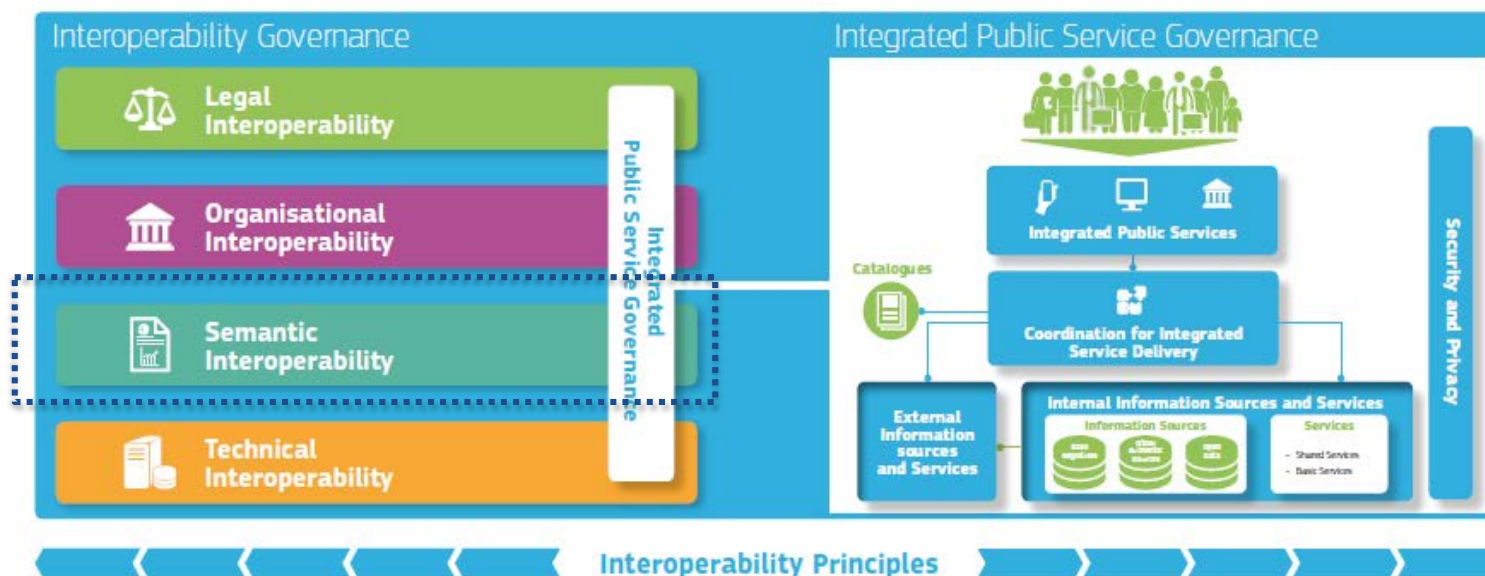


About the ISA² programme and SEMIC action



The ISA² programme supports the **development of digital solutions** that enable public administrations, businesses and citizens in Europe to benefit from **interoperable** digital public services.

European Interoperability Framework (EIF)



Objectives of the SEMIC action

- Supporting alignments and agreements on common definitions and specifications at the semantic layer for the Member States and the EU institutions.
- Increasing the visibility of existing data standards and promoting the use of ISA² specifications, such as Core Vocabularies, ADMS and DCAT-AP at the European, national and local level.
- Identifying needs for developing new common data models.
- Supporting projects executed by European Commission services and EU countries to promote semantic interoperability with pilots and expert advice.
- Promoting best practices, experiences and lessons-learned in the area of semantic interoperability and raising awareness on the importance of semantic interoperability and appropriate metadata management policies.
- Engaging with the SEMIC community, in particular by organising community-building events.

SEMIC solutions

e-GOVERNMENT
CORE
VOCABULARIES

DCAT-AP

ADMS
ASSET

Specifications

- Core Person
- Core Business
- Core Location
- Core Criterion and Core Evidence
- Core Public Organisation
- DCAT-AP
- ADMS



Tour de table

- Brief introduction of ISA² and SEMIC action
- All other participants, please your name and affiliation / country in the chat window

Objectives of the meeting

- Interaction with the community
- Sharing concerns, expectations and expertise
- Get to align on a base for a solution on the many challenges w.r.t. the SHACL template files

Useful reminder

SHACL is a language for validating RDF graphs against a set of conditions.

Shapes Constraint Language (SHACL)

W3C Recommendation 20 July 2017



This version:

<https://www.w3.org/TR/2017/REC-shacl-20170720/>

Latest published version:

<https://www.w3.org/TR/shacl/>

Latest editor's draft:

<https://w3c.github.io/data-shapes/shacl/>

Implementation report:

<https://w3c.github.io/data-shapes/data-shapes-test-suite/>

Previous version:

<https://www.w3.org/TR/2017/PR-shacl-20170608/>

Data validation using SHACL

SHACL validator

This service allows you to validate arbitrary RDF content against [SHACL shapes](#). It is also offered via a [SOAP](#) or [REST](#) API with further information available [here](#). Questions and feedback can be sent to DIGIT-ITB@ec.europa.eu.

Content to validate

File



Select file...

Content syntax [?]

Turtle

☒ Include external shapes [?]

External shapes

File



Validate

This service is powered by the ISA² Interoperability Test Bed, a conformance testing service offered by the European Commission's DG DIGIT for projects involved in the delivery of cross-border public services. Find out more [here](#).

<https://www.itb.ec.europa.eu/shacl/any/upload>

On the agenda

1. Importance of providing background knowledge with SHACL template for validation
2. Expressing the rules and constraints in the DCAT-AP specification in SHACL
3. Optimize the organization of the SHACL templates files (files, per constraint, message texts, ...) for enhancing its reusability
4. DCAT-AP as specification in an implementation context

Concerns github issues

- Issue 117/125/116/115: background knowledge
- Issue 118: expected messages for constraints
- Issue 119: MS extensions use alternative codelists
- Issue 55: dct:LinguisticSystem and skos:Concept, proposal NAL instances are not instances of dct:LinguisticSystem
- Issue 121: request for examples

Expressions of concerns to deliver correct/compliant data catalogs, especially from Germany towards the European Data Portal.

1. Background knowledge



Providing background knowledge alternates the result

- SHACL constraints take into account `rdfs:subClassOf` relations
 - <https://www.w3.org/TR/shacl/#targetClass>
- Demonstration:
 - A valid catalog with a publisher declared as `foaf:Organization`
 - According to the foaf vocabulary `foaf:Organization` `rdfs:subClassOf` `foaf:Agent` holds

Scenario 1 : validation
without the foaf
vocabulary included



Error

Scenario 2 : validation
with the foaf
vocabulary included



Valid



Conformant DCAT-AP validation

- Presence or absence of background knowledge has serious impact in the validation process.
- Including background vocabularies require decisions on:
 - Only the `rdfs:subClassOf` relationships
 - The whole vocabulary file
 - The dependencies of the imported vocabularies (transitive imports)
 - Version management of the dependencies

1. Background knowledge



Proposal

- DCAT-AP specification SHACL files published without background knowledge included
- A dependency import list is created capturing the subclass relationships that are considered to be acceptable for usage within DCAT-AP context
- The DCAT-AP shacl validator will include an option including the dependency imports.

1. Background knowledge



Any question about this topic ?

Should we exclude background knowledge from SHACL validation process ?

Should we provide SHACL validator with a dependency import list ?

Should this dependency import list be optional or mandatory in the validator?

2. Differences between implementation and a specification



2. Differences between implementation and a specification

(Re)Using the SHACL templates

Specification

The range of property *licence* is a *dct:LicenseDocument*.

Implementation

```
{  
  ...  
  "licence" : "https://creativecommons.org/publicdomain/zero/1.0/",  
  ...  
}
```

Implementation

- requires a URL which is considered to be the URI of the licence
- does not provide explicit evidence that the URI is an instance of class *dct:LicenseDocument*
- assumes / interprets this as instance of the class *dct:LicenseDocument*



2. Differences between implementation and a specification

(Re)Using the SHACL templates

Specification

The range of property *licence* is a *dct:LicenseDocument*.

Implementation

```
{  
  ...  
  "licence" : "https://creativecommons.org/publicdomain/zero/1.0/",  
  ...  
}
```

```
], [  
  sh:class dct:LicenseDocument ;  
  sh:maxCount 1 ;  
  sh:path dct:license ;  
  sh:severity sh:Violation  
], [  
  ...  
]
```



Problem

Errors reported for values that are not explicitly annotated as a *dct:LicenseDocument*.

2. Differences between implementation and a specification



Bridging the gap

Scenario 1 :
validation checks for class
membership

Scenario 2 :
validation does not check for
class membership

- Implementation requires addition of the class membership explicitly
- Class membership could be added automatically by the implementation (cfr GeoNetwork).
- Need for a global approach; not a “cherry pick approach”
- Valid for other application profiles ?
Other vocabularies ?

2. Differences between implementation and a specification



Proposal: DCAT-AP specification

- SHACL constraints for class membership in a separate file
- Create options in the DCAT-AP validator with/without class-membership

2. Differences between implementation and a specification



Proposal: for implementations

- Any implementation publishes its constraints & assumptions against which it validates the input.
- Can be done by an aggregated SHACL file based on the DCAT-AP SHACL files.

2. Differences between implementation and a specification



Any question about this topic ?

Should we collect all information regarding class membership in a separate file ?

Should we create insert an option in the validator to consider class membership or not?

Should implementations (e.g. the EDP) publish a document explaining how is the validation process working ?

3. Rules and constraints

3. Rules and constraints



Combined rules triggers confusion and generic error message

`:Agent_Shape`

```
a sh:NodeShape ;
sh:name "Agent"@en ;
sh:property [
  sh:minCount 1 ;
  sh:nodeKind sh:Literal ;
  sh:path foaf:name ;
  sh:severity sh:Violation
], [
  sh:class skos:Concept ;
  sh:maxCount 1 ;
  sh:path dct:type ;
  sh:severity sh:Violation
] ;
sh:targetClass foaf:Agent .
```

✖ Value must be an instance of foaf:Agent

Location: [Focus node] - [https://myorg.eu/opendata/catalog] - [Result path] - [http://purl.org/dc/terms/publisher]

Test: [Shape][Value]

- Is there a need for a more human-readable version of combined rules ?
- Are there global rules to be applied ?

3. Rules and constraints



Expressing usage levels in SHACL

Classes

Mandatory class: a receiver of data MUST be able to process information about instances of the class; a sender of data MUST provide information about instances of the class.

Recommended class: a sender of data SHOULD provide information about instances of the class; a sender of data MUST provide information about instances of the class, if such information is available; a receiver of data MUST be able to process information about instances of the class.

Optional class: a receiver MUST be able to process information about instances of the class; a sender MAY provide the information but is not obliged to do so.

Properties

Mandatory property: a receiver MUST be able to process the information for that property; a sender MUST provide information for that property.

Recommended property: a receiver MUST be able to process the information for that property; a sender SHOULD provide the information for that property if it is available.

Optional property: a receiver MUST be able to process information for that property; a sender MAY provide the information for that property but is not obliged to do so.

3. Rules and constraints



Expressing usage levels in SHACL

Usage level	Cardinality	severity	comment
Mandatory class	sh:minCount 1	sh:Violation	
Recommended class	sh:minCount 1	sh:Warning	<i>At the exchanged data level there is no difference between I do not want to provide or I do not know.</i>
Optional class	sh:minCount 0		No rule required
Mandatory property	sh:minCount 1	sh:Violation	
Recommended property	sh:minCount 1	sh:Warning	<i>At the exchanged data level there is no difference between I do not want to provide or I do not know.</i>
Optional property	sh:minCount 0		No rule required

3. Rules and constraints



Expressing usage levels in SHACL

Case: mandatory class Dataset (dcat:Dataset)



- Exchange with explicit types expressing class membership
- Exchange with 1 dataset explicitly typed as dcat:Dataset



Problem

Should this case be detected?
Mitigate the result by providing a warning ?



- Exchange without explicit types expressing class membership

3. Rules and constraints



Expressing usage levels in SHACL

Case: recommended class Licence Document



- Exchange with explicit types expressing class membership
- Exchange with 1 dataset explicitly typed as `dcat:Dataset`



- Exchange without explicit types expressing class membership

4.11. Licence Document

4.11.1. Recommended property for Licence Document

Property	URI	Range	Usage note	Card.
licence type	dct:type	skos:Concept	This property refers to a type of licence, e.g. indicating 'public domain' or 'royalties required'.	0..n

A licence document has recommended properties

3. Rules and constraints



Expressing usage levels in SHACL

Case: recommended class Licence Document

4.11. Licence Document				
4.11.1. Recommended property for Licence Document				
Property	URI	Range	Usage note	Card.
licence type	dct:type	skos:Concept	This property refers to a type of licence, e.g. indicating 'public domain' or 'royalties required'.	0..n

How to ensure that all Licence Documents have a licence type?

If no check for range class membership

- No Licence Documents are enforced
- Shacl rules on Licence Documents will never fire.

Does the usage level enforces to share also the descriptions of the supportive entities?

3. Rules and constraints



Severity level of the message

- A description has as range xsd:string
- A licence has as range Licence Document.

ex:catalogue1 dct:description "2020-04-01"^^xsd:date.
ex:catalogue1 dct:license ex:licence1.

ex:catalogue1 dct:description ex:description1.
ex:catalogue1 dct:license "This licence allows free reuse of the data".



today	Property is mandatory/ Recommended
Sh:Violation Sh:Violation	Sh:Violation Sh:Warning
Sh:Violation Sh:Violation	Sh:Violation Sh:Warning

Independent of the usage level
the resulting data structure is
incorrect.

3. Rules and constraints



Proposal handling usage levels

- Community feedback required to create proposal
- Approach should be applicable for other core vocabularies.

3. Rules and constraints



Any question about this topic ?

Message severity level only applicable for cardinality constraints?

Should there be an additional usage explanation for recommendation usage level in the context of a validation proces?

Should there be an additional usage explanation to rule out the case with 1 instance and others not?

3. Rules and constraints



SHACL and codelist integration

- **Demonstration:**
 - A valid catalog with as publisher
<http://publications.europa.eu/resource/authority/corporate-body/DIGIT>
 - the URI is a concept within the mandatory conceptscheme
<http://publications.europa.eu/resource/authority/corporate-body> for EU institutions

Scenario 1 : validation
without the corporate
bodies included



Warning

Scenario 2 : validation
with the corporate
bodies included



Valid

3. Rules and constraints



SHACL and codelist integration

```
], [  
  sh:node :CorporateBodyRestriction ;  
  sh:node :Publisher_ShapeCV ;  
  sh:nodeKind sh:IRI ;  
  sh:path dct:publisher ;  
  sh:description "A non EU managed concept is used to indicate the publisher, check if  
  sh:severity sh:Warning  
], [  

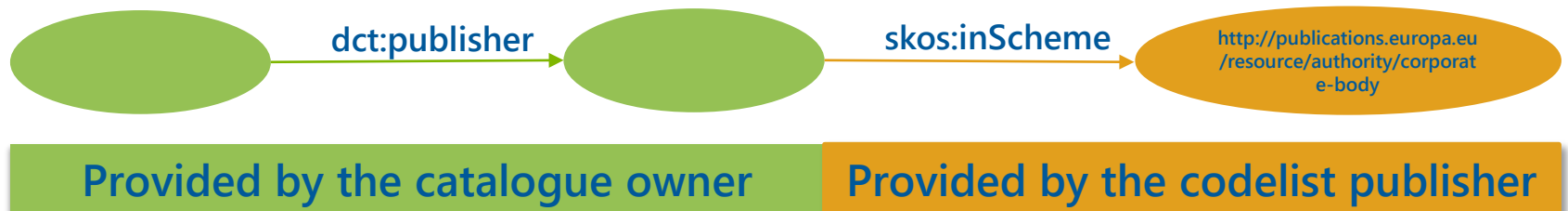
```

```
:CorporateBodyRestriction  
  a sh:NodeShape ;  
  rdfs:comment "Corporate Body Restriction" ;  
  rdfs:label "Corporate Body Restriction" ;  
  sh:property [  
    sh:hasValue <http://publications.europa.eu/resource/authority/corporate-body> ;  
    sh:minCount 1 ;  
    sh:nodeKind sh:IRI ;  
    sh:path skos:inScheme  
  ] .
```

3. Rules and constraints



SHACL and codelist integration: pattern checked





SHACL and codelist integration: pattern checked

- Alternative: using URI pattern checking instead of data structure checking
- Is life cycle information required to be checked? Awareness required for the situation: today correct, tomorrow wrong. Not all codelists have the same update frequency.
- Impact of the tool:
 - ISA testbed validation tool assumes “acceptable” sizes for the data to be checked. If the size of the background theories become too large, responsiveness goes below acceptable limits.
 - => Tool limitations determine the patterns to be checked.
 - Example: suppose the dereferencing of the conceptscheme would return the content of the file at <https://op.europa.eu/en/web/eu-vocabularies/at-dataset/-/resource/dataset/corporate-body> above tool limits could be hit.
 - Solutions:
 - publishers provide a more condensed response option
 - Users preprocess the data to filter out only the necessary information for the validation process

3. Rules and constraints



Any question about this topic ?

Applied approach is data structure checking?

Inclusion of life cycle information?

3. Rules and constraints



Codelists and class membership

- The range for *dct:language* is *dct:LinguisticSystem*.
- Recommended codelist is the language NAL:
<http://publications.europa.eu/resource/authority/language>
- Concepts in the NAL are not declared as instances of *dct:LinguisticSystem*.



Problem

Specification recommends a codelist which cannot be used without additional interpretation.

(technically it is the same challenge as on range class membership)



Codelists and class membership: proposal

- Ask publisher of the Language NAL to add class membership statement
- Provide support by the specification for this case
 - A dump of all instances of the Language NAL expressing class membership
 - Maintenance?
 - A tool creating the dump based on the Language NAL
 - Adapting the specification to enable better constraint expression.
 - Solution must honor the definition by dct:language which enforced the range
 - Introduction of a class dcatap:Language being a subclass of skos:Concept and dct:LinguisticSystem does not removes the need for the above enrichment.

3. Rules and constraints



Any question about this topic ?

Provide tooling to enrich the codelist with class membership?
Or provide alternative shacl description?

Wrap-up session & closing

Overview of the achievements of this meeting

Next steps

Availabilities for the next meeting ?

Suggestion

Announcement: Bugfix release 2.0.1 next week

To all who contributed

Thank you

**Thank you for your
participation !**

Join us in our first digital conference **#SEMIC 2020!**



#SEMIC2020

SEMIC
2020 Semantic Interoperability Community

15 October 2020
↳ 1ST DIGITAL

**Sustainable
data services**

Stay connected!



Share your views

[Satisfaction survey](#)



Join our community

SEMIC group on [LinkedIn](#)

SEMIC collection on [Joinup](#)



Contact us

contact@semic.eu



Follow our activity

SEMIC action on [ISA² website](#)



@EU_isa2



ISA² Programme