Overview of SHACL

https://book.validatingrdf.com/bookHtml011.html

SHACL Example

You can test the next two examples (slides) with the following data graph

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ex: <http://example.org/> .

ex:Donal_Trump
    a ex:User;
    foaf:name
        "Donald Trump" ;
    ex:birthdate
        "1946-06-14"^^xsd:date;
    foaf:knows
        ex:Joe_Biden.
```

SHACL Example

Every user has exactly one birthdate and the value (birthdate) must be xsd:date.

```
@prefix ex: <http://example.org/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

ex:UserShape a sh:NodeShape;
        sh:targetClass ex:User;
        sh:property [
                sh:path ex:birthdate;
                sh:minCount 1;
                sh:maxCount 1;
                sh:datatype xsd:date;
        ].
```

- Prefix
- User defined
- All nodes that are instances of ex:User
- Predicat
- Exactly one
- Value String

SHACL Example

The value in a foaf:knows property has to be a URI

```
@prefix ex: <http://example.org/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

ex:UserShape a sh:NodeShape;
        sh:targetClass ex:User;
        sh:property [
                sh:path foaf:knows;
                sh:nodeKind sh:IRI;
        ].
```

- Prefix
- User defined
- All nodes that are instances of ex:User
- Predicat
- Exactly one
- Value String

Some Concepts

General Concepts

- sh:NodeShape
- sh:targetClass
- sh:property
- sh:path

Others

- sh:class
- sh:minCount
- sh:maxCount
- sh:datatype
- sh:or
- sh:and
- sh:hasValue
- sh:value
- sh:pattern
- sh:in
- sh:nodeKind

Check **table 5.3** under **5.6.3 Constraint Components** in the documentation for more core concepts of SHACL

pySHACL Example

https://pypi.org/project/pyshacl/

```
from pyshacl import validate

results = validate(data_graph,
                   shacl_graph=sg,
                   ont_graph=og,
                   inference='rdfs',
                   abort_on_first=False,
                   allow_infos=False,
                   allow_warnings=False,
                   meta_shacl=False,
                   advanced=False,
                   js=False,
                   debug=False)

conforms, results_graph, results_text = results
```

data_graph - is an rdflib Graph object or file path of the graph to be validated

shacl_graph - is an rdflib Graph object or file path or Web URL of the graph containing the SHACL shapes to validate with

inference - is a Python string value to indicate whether or not to perform OWL inferencing expansion of the data_graph before validation. Options are ***'rdfs', 'owlrl', 'both', or 'none'.*** The default is **'none'**.

results - a three-component tuple containing:

- conforms - a bool value
- results_graph - graph
- results_text - string of Validation report