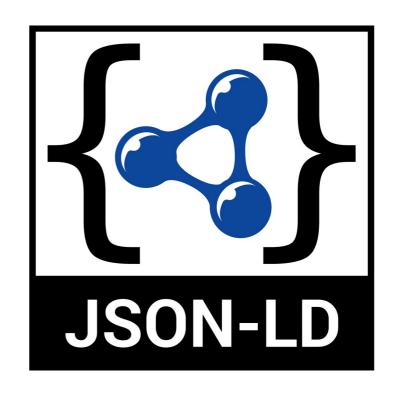
Gilles Landais (CDS) ASOV, mars 2024



JSON-LD motivations

Structurer des Données sérialisées en JSON

- Sérialisation en JSON de RDF
- Permet de structurer des documents JSON en proposant des mots clés qui suivent une sémantique
- Alternative au XML (XSD)
- Un des formats accepté par DataCite
- Format accepté par Google Dataset Search (basé su schema.org)

RDF

Sémantique IVOA : https://ivoa.net/rdf/

wikidata







Caractéristiques JSON-LD

- Première version 2010
- Standard W3C, actuellement V1.1: https://www.w3.org/TR/json-ld11/
- JSON linked Data (https://json-ld.org/)
- Internet media type : application/ld+json
- Permet de sérialiser une sémantiques
- Permet de décrire des données à partir d'une sémantique
- En théorie, il est possible d'annoter un document JSON avec une entête JSON-LD

Exemple

@context : URI vers la sémantique qui décrit les attributs utilisés : 'name', 'born', 'spouse'

{
 "@context": "https://json-ld.org/contexts/person.jsonld",
 "@id": "http://dbpedia.org/resource/John_Lennon",
 "name": "John Lennon",
 "born": "1940-10-09",
 "spouse": "http://dbpedia.org/resource/Cynthia_Lennon"



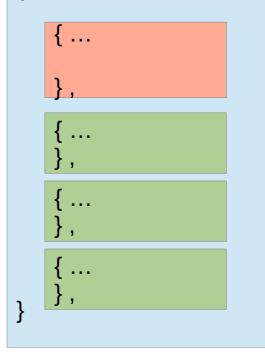
Structure d'un document JSON-LD

https://www.w3.org/TR/json-ld11/

- Définition des sémantique dans un contexte
- Structure de données :
 - Attribut /Valeur
 - Classe { ... } +- Objet d'un diag UML ex DCAT : https://www.w3.org/TR/vocab-dcat-2/
 - Liste [...]

Contexte

Data



Mots clés JSON-LD

@id : définit une IRI

@base : base d'une IRI utilisé comme préfix

@context : définit les semantique

@type : définit le type d'une resource

(~ dmtype en MIVOT) @value : spécifie une valeur @list : spécifie une liste

@set : list sans ordre
@version in @context

@language : pour les texte, spécifie un language

...



Sérialisation Turtle

```
<a href="http://dbpedia.org/resource/John_Lennon"> a <a href="http://son-ld.org/contexts/person.jsonld">http://son-ld.org/contexts/person.jsonld</a>; <a href="http://son-ld.org/contexts/person.jsonld">http://son-ld.org/contexts/person.jsonld</a>; <a href="http://son-ld.org/contexts/person.jsonld">http://son-ld.org/contexts/person.jsonld</a>; <a href="http://son-ld.org/contexts/person.jsonld</a>; <a href="http://sonld.org/contexts/person.jsonld</a>; <a href="http://sonld.org/contexts/person.jsonld.org/contexts/person.jsonld</a>; <a href="http://sonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.jsonld.org/contexts/person.j
```

Sérialisation JSON-LD (utilisation des URI comme attribut)

```
"@id": "http://dbpedia.org/resource/John_Lennon",
    "http://xmlns.com/foaf/0.1/name": "John Lennon",
    "http://schema.org/birthDate": {
        "@type": "http://www.w3.org/2001/XMLSchema#date",
        "@value": "1940-10-09"
    },
    "http://schema.org/spouse": {
        "@id": "http://dbpedia.org/resource/Cynthia_Lennon"
    }
}
```

Sérialisation JSON-LD compacte et utilisant un contexte

@context : URI vers la sémantique qui décrit les attributs utilisés : 'name', 'born', 'spouse'

```
{
    "@context": "https://json-ld.org/contexts/person.jsonld",
    "@id": "http://dbpedia.org/resource/John_Lennon",
    "name": "John Lennon",
    "born": "1940-10-09",
    "spouse": "http://dbpedia.org/resource/Cynthia_Lennon"
}
```

JSON-LD

Sérialisation JSON-LD avec la définition du contexte

Sérialisation JSON-LD avec la définition du contexte

```
{
    "@context": {
        "xsd": "http://www.w3.org/2001/XMLSchema#",
        "name": { "@id": "_:name", "@type": "xsd:string" },
        "born": { "@id": "_:born", "@type": "xsd:date" },
        "spouse": { "@id": "_:spouse", "@type": "@id" }
    },
    "@id": "http://dbpedia.org/resource/John_Lennon",
    "name": "John Lennon",
    "born": "1940-10-09",
    "spouse": "http://dbpedia.org/resource/Cynthia_Lennon"
}
```

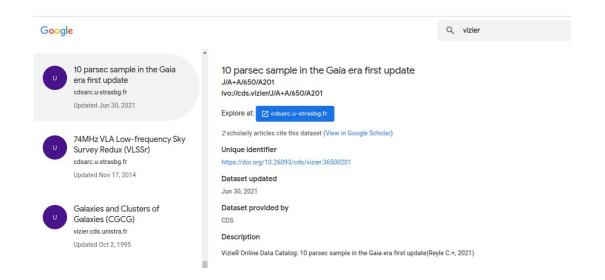
Exemple de serialisation en JSON-LD fournie dans les landing page

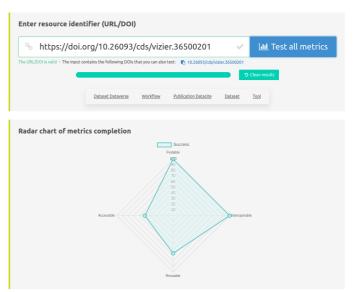
Sémantique : Schema.org https://schema.org/

Schema.org

• Exemple: https://cdsarc.cds.unistra.fr/viz-bin/cat/J/AJ/894/142







https://fair-checker.france-bioinformatique.fr/check

Exemple de serialisation JSON-LD utilisées dans les landing page

- Exemple VizieR: https://vizier.cds.unistra.fr/viz-bin/cat/J/AJ/137/226
- Sémantique : https://schema.org/ Schema.org/



```
Resource rdf décrivant les termes utilisés
                      "@context": "http://schema.org/",
le type de la ressource
                      "@type": "Dataset",
                                                                                         IRI du document
https://schema.org/Dataset
                      "@id": "https://doi.org/10.26093/cds/vizier.51370226",
                      "identifier": "https://doi.org/10.26093/cds/vizier.51370226",
                      "alternateName": [
                           "J/AJ/137/226", "ivo://cds.vizier/J/AJ/137/226"
                      "url": "https://vizier.cds.unistra.fr/viz-bin/cat/J/AJ/137/226",
                      "name": "Differential light curves of TV Tri",
                      "description": "VizieR Online Data Catalog",
                      "rights": "https://cds.unistra.fr/vizier-org/licences vizier.html",
                      "author": [
                           { "givenName": "Yang Y.-G." },
                                                                   Utilisation d'une liste d'auteurs :
                           { "givenName": " Wei J.-Y." }
                                                                    https://schema.org/author
                       "creator": {
                           "@type": "Organization",
                                                                        Objet Person « Organization »
                           "name": "CDS",
                           "url": "https://cds.unistra.fr"
                                                                        https://schema.org/Organisation
                      "datePublished": "26-Oct-2011",
                      "isAccessibleForFree": true,
                      "keywords": [
                                   "Eclipsing binary stars", "Optical astronomy, CCD photometry"
                      "publication": "2009AJ....137..226Y"
```





Exemple d'utilisation de 2 sémantiques existantes dans un contexte

Schema.org



```
{
    "@context": [
        "http://schema.org",
        { "dcterm": "http://purl.org/dc/terms/" }
    "@type": "Dataset",
    "@id": "https://doi.org/10.26093/cds/vizier.1350",
    "identifier":[
        {
            "@type": "PropertyValue",
            "propertyID": "ivoid",
            "value": "ivo://CDS.VizieR/i/350"
    "citation": [
        { "@id": "https://doi.org/10.5270/esa-1ugzkq7",
          "@type": "Dataset" },
        { "@id": "https://ui.adsabs.harvard.edu/abs/2021A+A...649A...1G",
          "@type": "ScholarlyArticle" }
    "dcterm:isReferencedBy": [
        { "@id": "https://doi.org/10.26093/cds/vizier.36510104",
          "@type": "Dataset" }
```



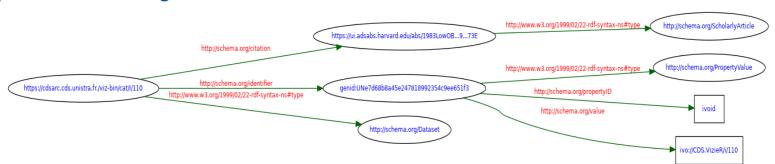
Le coin du développeur

API en Python, Java, C, TypeScript, Rust, ...

https://json-ld.org/#developers

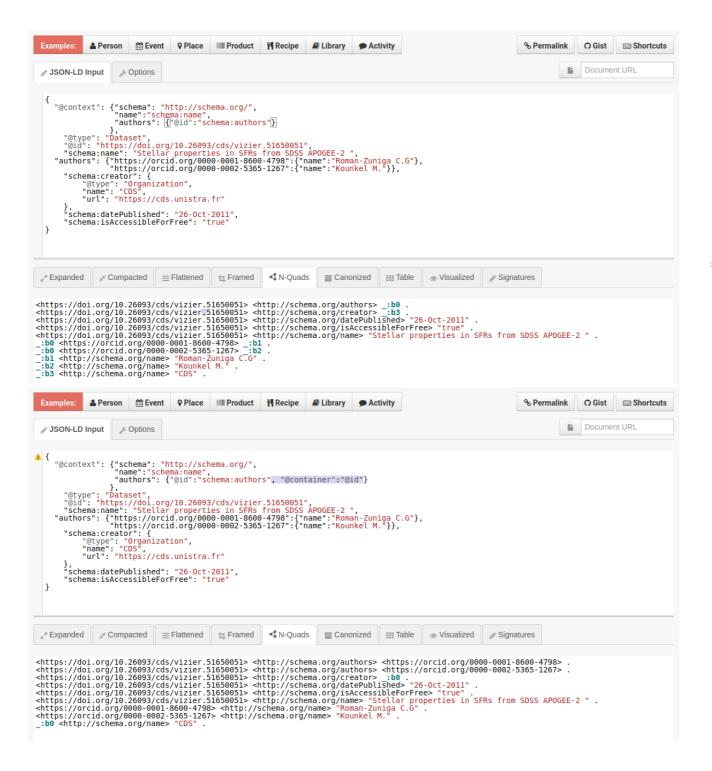
Possibilité de convertir une sérialisation RDF en une autre..

- Exemples :https://json-ld.org/
- (tentatives) de Validateurs
 - https://www.jsonschemavalidator.net/
 - https://json-ld.org/playground/
- RDF validateur :
 - https://www.w3.org/RDF/Validator/
 - https://www.w3.org/2015/03/ShExValidata/



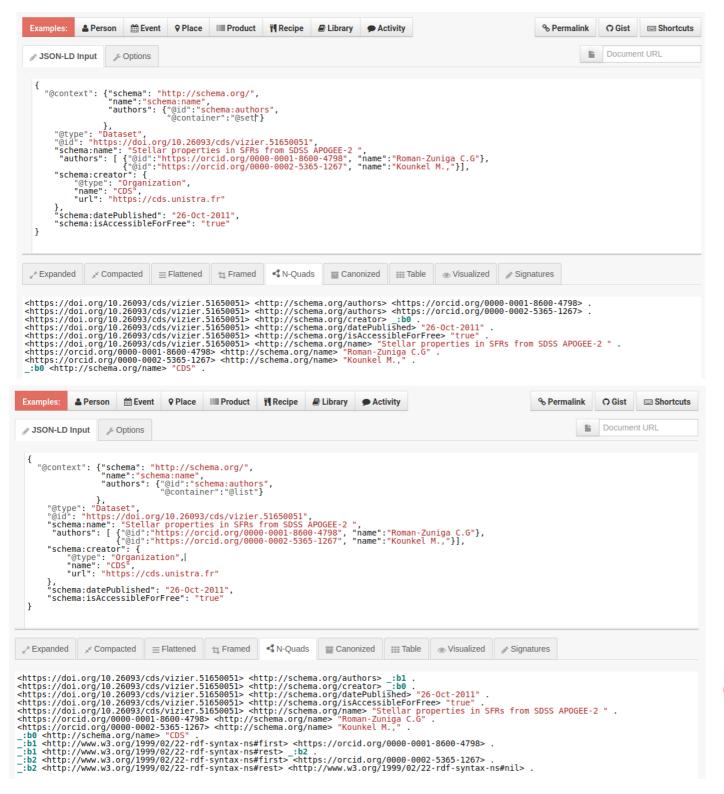
from rdflib import Graph

g = Graph()
result = g.parse("test_rdflib.json", format="json-ld")
result.serialize(format="xml")





@container n'est pas defini⇒ empty node





@container avec un set

@container avec une liste



- Un standard W3C
- Un candidat pour fournir du JSON dans le cadre du VO
- Autres alternatives :
 - JSON schema
 - JSON api



{ json:api }