

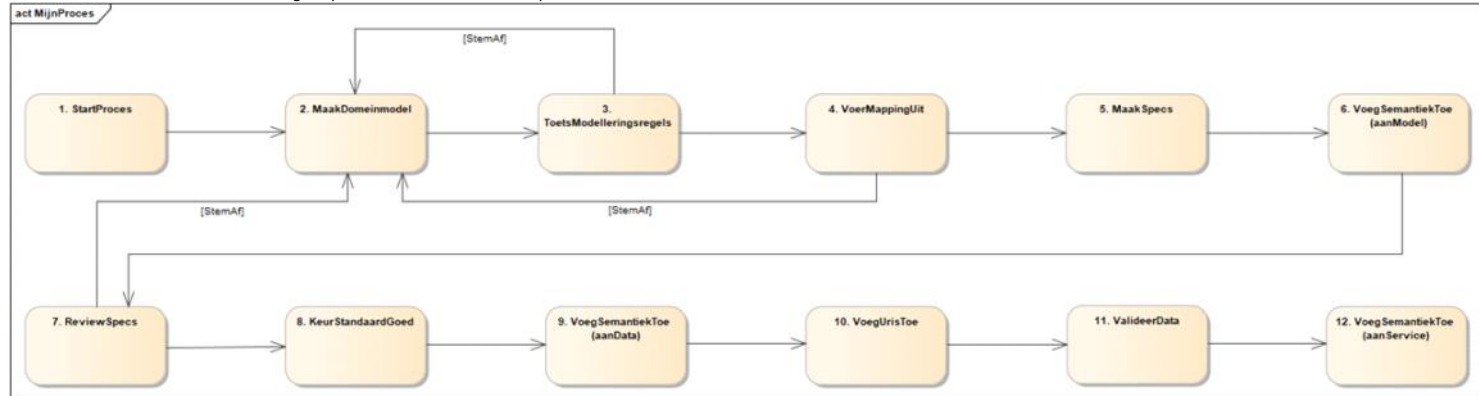
# OSLO-compliance Checklist & Proces

woensdag 4 september 2019 15:26

Checklist OSLO-compliance voor applicaties:

Thema	Item	Check
URI	Nemen de id's vd data de vorm aan ve URI?	
	Is deze URI in overeenstemming met de URI-richtlijn?	
	Zijn de URI's PURI's?	
Aanpak	Werd getoetst aan bestaande standaarden?	
	Werd nieuw VOC/AP ontwikkeld volgens OSLO Proces & Methodologie?	
	Zijn er gepubliceerde specs?	
Datamodel	Is er een domeinmodel?	
	Semantiek toegevoegd aan model?	
	Voldoet het model aan de modelleringsregels?	
Service	Is de data verrijkt met de semantiek?	
	Kan de data gevalideerd worden?	
	Voldoet de service aan de API-richtlijn?	

Om aan deze checklist te voldoen zou volgend proces kunnen worden doorlopen:



Maar dat is voor een scenario waar tbv een nieuwe applicatie ook een nieuwe standaard wordt gemaakt. In praktijk zijn er nog andere scenario's:

<a href="#">Scenario1</a>	Bestaande applicatie	Bestaande OSLO-standaarden
<a href="#">Scenario2</a>	Nieuwe applicatie	Bestaande OSLO-standaarden
<a href="#">Scenario3</a>	Nieuwe applicatie	Nieuwe OSLO-standaard

Maw: het volledige proces slaat op Scenario3.

De andere scenario's kunnen stappen overslaan, sommige items op de compliance-checklist zijn immers niet van toepassing, bv bij Scenario1 is de check "Werd nieuw VOC/AP ontwikkeld volgens OSLO Proces & Methodologie?" overbodig.

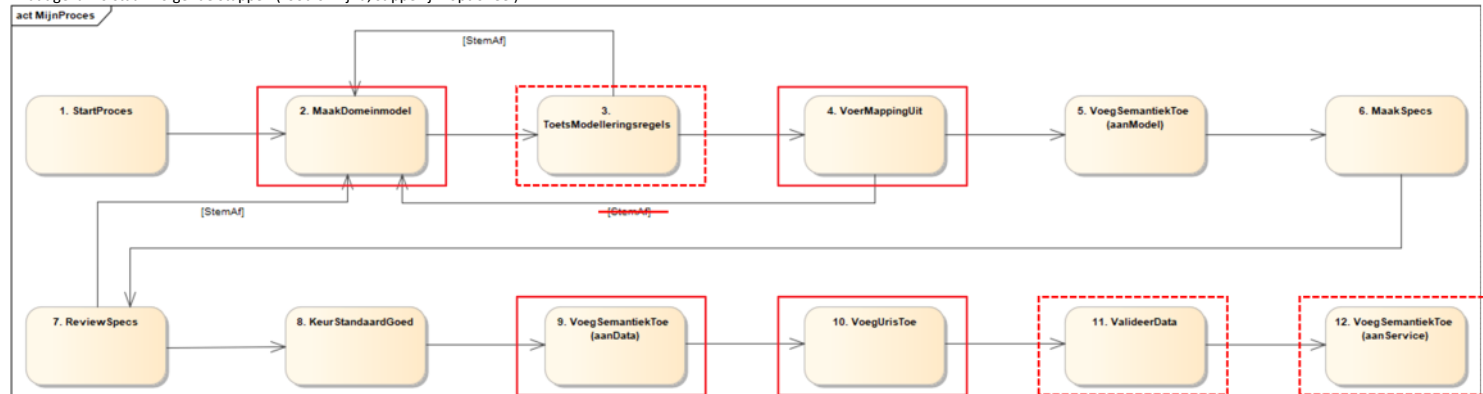
# OSLO-compliance demo Scenario1

dinsdag 3 september 2019 11:27

Demo Scenario1:

Scenario1	Bestaande applicatie	Bestaande OSLO-standaarden
Scenario2	Nieuwe applicatie	Bestaande OSLO-standaarden
Scenario3	Nieuwe applicatie	Nieuwe OSLO-standaard

In dat geval volstaan volgende stappen (rood omlijnd, stippellijn=optioneel):



Voor de demo vertrekken we van een bestaande JSON-service:

- Bv API geeft JSON-output met deze array van museums:

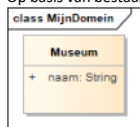
```
[
  {
    "naam": "FOMU"
  },
  {
    "naam": "MIMA"
  }
]
```

- TODO: Echte service maken, zie bv x.

Scenario:

- 2. MaakDomeinmodel

- Op basis van bestaand ImplementatieModel (IM), bv:



- In UML (zie §3.2.1 vd [Modelleringsregels](#)), het domeinmodel kan daardoor makkelijker getoetst worden aan bestaande modellen.

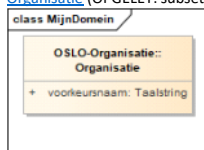
- 3. ToetsModelleringsregels

- OPTIONEEL: Maakt 4.VoerMappingUit makkelijker.

- 4. VoerMappingUit

- Zie [Standaardenregister](#) voor een overzicht vd huidige standaarden..

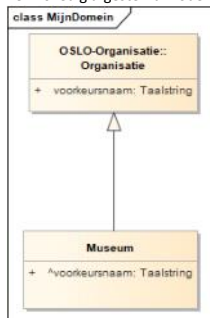
- Museum komt daarin momenteel niet voor, dichtbij komt Organisatie uit [OSLO-Organisatie](#) (OPGELET: subset van werkelijke attributen):



- Mapping (zie x voor template en meer uitleg in §3.1.2 vd [Modelleringsregels](#)):

Element	Aard	Matchtype	Element	Aard	Opmerking
MijnDomein:Museum	klasse	Has a narrow match	OSLO-Organisatie:Organisatie	klasse	
MijnDomein:Museum.naam	attribuut	Has an exact match	OSLO-Organisatie:Organisatie.vorkeursnaam	attribuut	

- Een volledig afgestemd model zou er dus zo uitzien:



(Waarbij het ^-teken aangeeft dat het attribuut voorkeursnaam wordt overgeërfd van OSLO-Organisatie:Organisatie).

- Maar in deze demo gaan we trachten het huidige model te behouden, maw de StemAf control flow van VoerMappingUit naar MaakDomeinmodel valt weg.

- 9. VoegSemantiekToe (aanData)

- We doen dat door JSON om te zetten in JSON-LD (LD staat voor Linked Data).

- Met @type verwijzen we naar de definitie vd objecten als volgt:

```
[
  {
    "@type": "http://www.w3.org/ns/org#Organization",
    "naam": "FOMU"
  },
  {
    "@type": "http://www.w3.org/ns/org#Organization",
```

```

    "naam": "MIMA"
  }
}

```

- OPMERKING: De definities worden in RDF gepubliceerd op het web. De URI (in ons vb <http://www.w3.org/ns/org#Organization>) neemt de vorm aan van een URL.
- OPMERKING: De definitie in het vb is niet door OSLO gecreëerd (de vorm zou dan zijn <https://data.vlaanderen.be/ns/organisatie#Organisatie>), OSLO hergebruikt een bestaande definitie.
- OPMERKING: Gebruikers vd service weten nu alvast dat FOMU en MIMA organisaties zijn. Beter zou zijn dat we mappen op iets als OSLO-Kunst:Museum maar dat bestaat momenteel niet.
- Om de semantiek van naam mee te geven zou dit volstaan:
 

```

[
  {
    "@type": "http://www.w3.org/ns/org#Organization",
    "http://www.w3.org/2004/02/skos/core#prefLabel": "FOMU"
  },
  {
    "@type": "http://www.w3.org/ns/org#Organization",
    "http://www.w3.org/2004/02/skos/core#prefLabel": "MIMA"
  }
]

```

- Maar met @context kan het beknopter:

```

{
  "@context": {
    "Organisatie": "http://www.w3.org/ns/org#Organization",
    "naam": "http://www.w3.org/2004/02/skos/core#prefLabel"
  },
  "@graph": [
    {
      "@type": "Organisatie",
      "naam": "FOMU"
    },
    {
      "@type": "Organisatie",
      "naam": "MIMA"
    }
  ]
}

```

- TIP: Check in [JSON-LD Playground](#) of de [Google Tool voor gestructureerde gegevenstests](#) of de JSON-LD klopt. Bv het bovenstaande geeft in de Google Tool dit:

http://www.w3.org/ns/org#Organization	0 FOUTEN	0 WAARSCHUWINGEN	^
@type	http://www.w3.org/ns/org#Organization		
http://www.w3.org/2004/02/skos/core#prefLabel	FOMU		

http://www.w3.org/ns/org#Organization	0 FOUTEN	0 WAARSCHUWINGEN	^
@type	http://www.w3.org/ns/org#Organization		
http://www.w3.org/2004/02/skos/core#prefLabel	MIMA		

- OPMERKING: We willen hier de bestaande JSON zo min mogelijk wijzigen en behouden dus ook de elementnamen. Zouden we dat niet doen dan zou een verwijzing in @context naar de OSLO-contextfile volstaan:

```

{
  "@context": {
    "https://data.vlaanderen.be/context/organisatie-basis.jsonld",
    "@graph": [
      {
        "@type": "Organisatie",
        "voorkeursnaam": "FOMU"
      },
      {
        "@type": "Organisatie",
        "voorkeursnaam": "MIMA"
      }
    ]
  }
}

```

TODO: Fout in contextfile (Lidmaatschap.isLidVan komt twee keer voor).

#### 10. Voeg Uri's Toe

- Wellicht zal de JSON ook ID's bevatten, bv:

```

[
  {
    "id": 123,
    "naam": "FOMU"
  },
  {
    "id": 456,
    "naam": "MIMA"
  }
]

```

- OSLO schrijft voor om daar URI's van te maken, bv

```

[
  {
    "@id": "https://voorbeeld.be/id/organisatie/123",
    "naam": "FOMU"
  },
  {
    "@id": "https://voorbeeld.be/id/organisatie/456",
    "naam": "MIMA"
  }
]

```

- Deze URI's moeten voldoen aan de [OSLO URI-standaard](#) (gebruik de compliance checklist in §5).

- Wordt de data door AIV gepubliceerd dan gelden bijkomende richtlijnen (zie [AIV URI-standaard](#)). De URI's zouden er dan eerder zo uitzien:

```

[
  {
    "@id": "https://data.vlaanderen.be/id/organisatie/123",
    "naam": "FOMU"
  },
  {
    "@id": "https://data.vlaanderen.be/id/organisatie/456",

```

- ```

    "naam": "MIMA"
  }
}

```
- 11. ValideerData
    - OPTIONEEL: Als de OSLO-Standaard van toepassing is op de bestaande applicatie. In dit geval is Organisatie-Basis van toepassing, de museums worden hier beschouwd als organisaties.
    - Bij JSON-LD is validatie mogelijk met behoud van eigen elementnamen dankzij de @context. De standaard OSLO-validator checkt op:
      - Semantiek
      - Datatypes
      - Kardinaliteiten
    - Welke beschreven zijn in een zgn [SHACL-file](#).
    - Valideren we het volgende tegen de OSLO-Standaard Organisatie-Basis:
 

```

{
  "@context": {
    "Organisatie": "http://www.w3.org/ns/org#Organization",
    "naam": "http://www.w3.org/2004/02/skos/core#prefLabel"
  },
  "@graph": [
    {
      "@id": "https://voorbeeld.be/id/organisatie/123",
      "@type": "Organisatie",
      "naam": "FOMU"
    },
    {
      "@id": "https://voorbeeld.be/id/organisatie/456",
      "@type": "Organisatie",
      "naam": "MIMA"
    }
  ]
}

```
  - Dan krijgen we dit:

| Focus Node                              | Result Message                              | Result Path                                   | Result Severity                      | Source Constraint | Source Constraint Component                            | Source Shape | Value |
|-----------------------------------------|---------------------------------------------|-----------------------------------------------|--------------------------------------|-------------------|--------------------------------------------------------|--------------|-------|
| https://voorbeeld.be/id/organisatie/123 | Value does not have datatype rdf:langString | http://www.w3.org/2004/02/skos/core#prefLabel | http://www.w3.org/ns/shacl#Violation | NA                | http://www.w3.org/ns/shacl#DatatypeConstraintComponent | _b0          | FOMU" |
| https://voorbeeld.be/id/organisatie/456 | Value does not have datatype rdf:langString | http://www.w3.org/2004/02/skos/core#prefLabel | http://www.w3.org/ns/shacl#Violation | NA                | http://www.w3.org/ns/shacl#DatatypeConstraintComponent | _b0          | MIMA" |

- Na correctie als volgt:
 

```

{
  "@context": {
    "Organisatie": "http://www.w3.org/ns/org#Organization",
    "naam": "http://www.w3.org/2004/02/skos/core#prefLabel"
  },
  "@graph": [
    {
      "@id": "https://voorbeeld.be/id/organisatie/123",
      "@type": "Organisatie",
      "naam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://voorbeeld.be/id/organisatie/456",
      "@type": "Organisatie",
      "naam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}

```
- Krijgen we geen fouten meer.
- 12. VoegSemantiekToe (aanService)
  - TODO

Uiteindelijk resultaat van dit scenario:

```

{
  "@context": {
    "Organisatie": "http://www.w3.org/ns/org#Organization",
    "naam": "http://www.w3.org/2004/02/skos/core#prefLabel"
  },
  "@graph": [
    {
      "@id": "https://voorbeeld.be/id/organisatie/123",
      "@type": "Organisatie",
      "naam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://voorbeeld.be/id/organisatie/456",
      "@type": "Organisatie",
      "naam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}

```

Worden deze stappen doorlopen dan is als volgt voldaan aan de checklist:

| Thema  | Item                                                             | Check |
|--------|------------------------------------------------------------------|-------|
| URI    | Nemen de id's vd data de vorm aan ve URI?                        | JA    |
|        | Is deze URI in overeenstemming met de URI-richtlijn?             | JA    |
|        | Zijn de URI's PURI's?                                            | NVT   |
| Aanpak | Werd getoetst aan bestaande standaarden?                         | JA    |
|        | Werd nieuw VOC/AP ontwikkeld volgens OSLO Proces & Methodologie? | NVT   |

|           |                                              |     |
|-----------|----------------------------------------------|-----|
|           | Zijn er gepubliceerde specs?                 | NVT |
| Datamodel | Is er een domeinmodel?                       | JA  |
|           | Semantiek toegevoegd aan model?              | NVT |
|           | Voldoet het model aan de modelleringsregels? | OPT |
| Service   | Is de data verrijkt met de semantiek?        | JA  |
|           | Kan de data gevalideerd worden?              | OPT |
|           | Voldoet de service aan de API-richtlijn?     | OPT |

Discussiepunten:

- ()

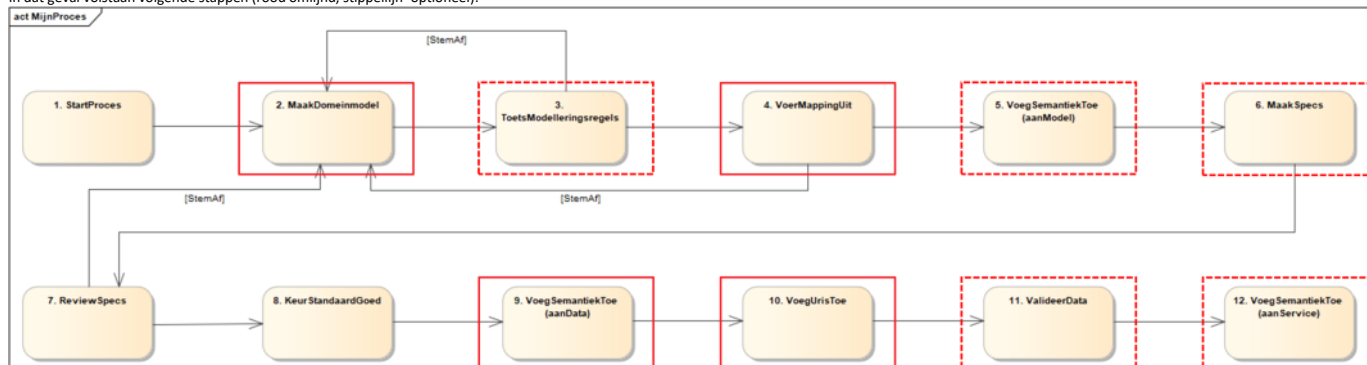
# OSLO-compliance demo Scenario2

donderdag 5 september 2019 11:06

Demo Scenario1:

|           |                      |                            |
|-----------|----------------------|----------------------------|
| Scenario1 | Bestaande applicatie | Bestaande OSLO-standaarden |
| Scenario2 | Nieuwe applicatie    | Bestaande OSLO-standaarden |
| Scenario3 | Nieuwe applicatie    | Nieuwe OSLO-standaard      |

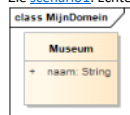
In dat geval volstaan volgende stappen (rood omlijnd, stippellijn=optioneel):



Voor de demo gaan we ervan uit dat we een service willen ontwikkelen als in [scenario1](#), maar dan wel afgestemd op OSLO.

Scenario:

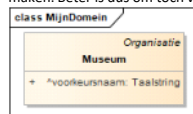
- 2. MaakDomeinmodel
  - o Zie [scenario1](#).
- 3. ToetsModelleringsregels
  - o OPTIONEEL: Zie [scenario1](#).
- 4. VoerMappingUit
  - o Zie [scenario1](#). Echter gaan we nu niet het oorspronkelijk model behouden, dus niet:



o Maar wel:

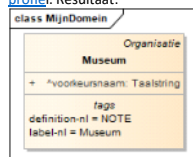


- o In dat geval hoeven we geen eigen @context toe te voegen, we kunnen direct valideren tegen OSLO-Organisatie.
- o MAAR: Dit is enkel ok voor applicaties die Museums niet als Museums maar als Organisaties beschouwen. Bovendien: willen we bv attributen specifiek voor Museums (bv openingsuren) dan zouden we een eigen versie van OSLO-Organisatie moeten maken. Beter is dus om toch voor het resultaat vd mapping in [scenario1](#) te gaan:

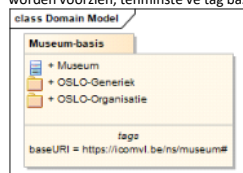


- o Als we hiervan specs maken en/of een SHACL-file (om data tegen te valideren), kunnen we dit model gebruiken voor toepassingen in de eigen context (bv aannemers moeten data aanleveren in bepaald formaat, zie use case SB250 van AWV).
- o Later kan er nog een OSLO-standaard van worden gemaakt (zie [scenario3](#)).

- 5. VoegSemantiekToe (aanModel)
  - o OPTIONEEL: Dit is nodig om specs/SHACL te maken.
  - o We gebruiken het standaard mechanisme van UML om de metadata bij het model op te slaan: valued tags (zie §3.2.19 vd [Modelleringsregels](#)).
  - o Het toevoegen van tags kan handmatig of gebruik het door OSLO verstrekte [UML-profiel](#). Resultaat:



- o OPMERKING: NOTE als waarde vd tag te groot is (in EA), waarde van definition-nl is eigenlijk: Instelling, zonder winstbejag, ten dienste van de gemeenschap en van haar ontwikkeling, toegankelijk voor het publiek, die de materiële en immateriële getuigenissen van de mens en zijn omgeving verzamelt, bewaart, onderzoekt en tentoonstelt en hierover informatie verstrekt voor studie, educatie en recreatie.
- o OPMERKING: Dit is de huidige definitie van het begrip Museum volgens de [International Council of Museums](#) (ICOM).
- o Ook het model (in UML vertegenwoordigd door een package) moet van metadata worden voorzien, tenminste ve tag baseURI. Resultaat:



- o OPMERKING: We doen hier alsof de specs (en de service) ontwikkeld worden door ICOMVL, vandaar de baseURI <https://icomvl.be/ns/museum#>.
- o OPGELET: Ook de elementen uit de OSLO-Standaarden (in Dit geval [OSLO-Organisatie](#) voor de klasse Organisatie en [OSLO-Generiek](#) voor het datatype Taalstring) en hun packages moeten voorzien zijn vd nodige tags.

- Makkelijkste manier om dit te bereiken is de packages importeren. Overbodige elementen mogen daarbij uit het package worden verwijderd (die vh VOC zijn immers elders al gedefinieerd en voor het AP behouden we enkel wat noodzakelijk is voor de applicatie):



- 6. MaakSpecs
  - OPTIONEEL: Laat toe om het model als specificatie te publiceren of uit te wisselen. Produceert ook een contextfile (nuttig voor stap 9 VoegsemantiekToe (aanData)) en een SHACL-file (nuttig voor stap 11 ValideerData).
  - Gebruik hiervoor de OSLO-Toolchain (TODO: Definitieve link?).
  - Men kan de toolchain lokaal of online draaien, we demonstreren hier de online manier (VRAAG: Officiële AIV-dienstverlening?).
  - Maak eerst een eigen github repository om het model + configuratie - en andere noodzakelijke bestanden op te slaan, bv [deze repository](#):



- Met volgende structuur/bestanden:

| GeertThijs Add files via upload |                      | Latest commit 3480732 now |
|---------------------------------|----------------------|---------------------------|
| config                          | Delete dummy         | 18 minutes ago            |
| site-skeleton                   | Delete dummy         | 17 minutes ago            |
| templates                       | Add files via upload | 16 minutes ago            |
| README.md                       | Create README.md     | 42 minutes ago            |
| museum-basis.EAP                | Add files via upload | 5 minutes ago             |
| museum.EAP                      | Add files via upload | 5 minutes ago             |
| stakeholders.csv                | Add files via upload | now                       |

- Waarbij:
  - De toolchain werkt met EAP-bestanden, hier dus museum.EAP voor het vocabularium (VOC) en museum-basis.EAP voor het applicatieprofiel (AP).
  - Het bestand stakeholders bevat de auteurlijst vh model, hier:  
 Voornaam;Naam;Affiliatie;E-mail;Website;Museum  
 Geert;Thijs;Informatie  
 Vlaanderen;geert.thijs@kb.vlaanderen.be;https://overheid.vlaanderen.be/informatie-vlaanderen;A
  - De folder config bevat een configuratiebestand in JSON voor het VOC en voor het AP. Bv voor het VOC:
 

```

[
  {
    "name": "museum",
    "type": "voc",
    "eap": "museum.EAP",
    "diagram": "Museum",
    "template": "museum-voc.j2",
    "title": "Museum",
    "publication-state": "zonderStatus",
    "publication-date": "17-09-2019",
    "license": "",
    "contributors-file": "stakeholders.csv",
    "contributors-column": "Museum",
    "site": "site-skeleton",
    "feedbackurl": "",
    "standaardregisterurl": ""
  }
]
          
```
  - De folder site-skeleton bevat typisch het UML-diagram voor het AP.
  - In templates staan uitbreidingen op de basistemplate vd specificatie. Bv voor het VOC:
 

```

{% extends "voc2.j2" %}
{% block summary %}
<p>Het ICOMVL-Museum vocabularium bevat termen die betrekking hebben op musea.</p>
{% endblock %}
{% block status %}
<p>Dit document werd gepubliceerd door ICOM-Vlaanderen.</p>
{% endblock %}
          
```

 (TODO: Welk formaat is dit eigenlijk?)
- De toolchain runt automatisch vanuit de github repository [data.vlaanderen.be](#).

momenteel nog vanuit de branch [test-feature-checkout](#) (TODO: Definitieve link?).  
 In configuratiefile [publication.json](#) staat een lijst met aan te maken specs. Daaraan voegen we onze modellen toe:

```
{
  "urlref":
  "/doc/vocabulary/museum/zonderstatus/20190917",
  "repository": "https://github.com/GeertThijs/demo-
  MaakSpecs",
  "branchtag" : "master",
  "name" : "museum",
  "filename" : "config/museum.json",
  "dummy" : "dummy001",
  "navigation" : {
    "next" : "/ns/museum"
  }
},
{
  "urlref": "/doc/applicatieprofiel/museum-
  basis/zonderstatus/20190917",
  "repository": "https://github.com/GeertThijs/demo-
  MaakSpecs",
  "branchtag" : "master",
  "name" : "museum-basis",
  "filename" : "config/museum-basis.json",
  "dummy" : "dummy001",
  "navigation" : {
    "next" : "/doc/applicatieprofiel/museum-basis"
  }
}
```

- De toolchain schrijft zijn resultaat in github directory [OSLO-Generated](#), nu nog in de branch [test-feature-checkout](#) (TODO: Definitieve link?).
- De specs in ruwe vorm zijn daar te vinden op de in de configuratiefile opgegeven relatieve paden bv voor het VOC [https://github.com/informatievlaanderen/OSLO-Generated/tree/test-feature-checkout/doc/vocabulary/museum/zonderstatus/20190917](#).
- Waarna ze automatisch online worden gezet momenteel op [otl-test.data.vlaanderen](#) (TODO: Definitieve link?)
- Resultaat is onze eigen spec voor museums:
  - VOC: [https://otl-test.data.vlaanderen.be/doc/vocabulary/museum/zonderstatus/20190917](#)
  - AP: [https://otl-test.data.vlaanderen.be/doc/applicatieprofiel/museum-basis/zonderstatus/20190917](#)

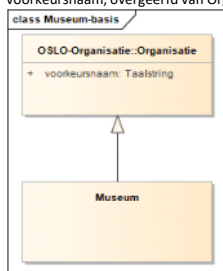
- In het VOC vinden we onze klasse Museum terug, gedefinieerd als subklasse van OSLO-Organisatie:
 

**Klasse *Museum***

| Type              | Klasse                                                                                                                                                                                                                                                                                                                 |
|-------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| URI               | <a href="https://icoml.be/ns/museum#Museum">https://icoml.be/ns/museum#Museum</a>                                                                                                                                                                                                                                      |
| Specialisatie van | <a href="http://www.w3.org/ns/org#Organization">http://www.w3.org/ns/org#Organization</a>                                                                                                                                                                                                                              |
| Definitie         | instelling, zonder winstbejag, ten dienste van de gemeenschap en van haar ontwikkeling, toegankelijk voor het publiek, die de materiële en immateriële getuigenissen van de mens en zijn omgeving verzamelt, bewaart, onderzoekt en tentoontstelt en hierover informatie verstrekt voor studie, educatie en recreatie. |

- En in het AP wordt uitgelegd hoe de klasse Museum moet worden gebruikt, ttz als subklasse van OSLO-Organisatie:
 

Organisatie en verplicht met het attribuut voorkeursnaam, overgeërfd van Organisatie:



## Museum

### Beschrijving

Instelling, zonder winstbejag, ten dienste van de gemeenschap en van haar ontwikkeling, toegankelijk voor het publiek, die de materiële en immateriële getuigenissen van de mens en zijn omgeving verzamelt, bewaart, onderzoekt en tentoontstelt en hierover informatie verstrekt voor studie, educatie en recreatie.

### Subklasse van

[Organisatie](#)

### Eigenschappen

Voor deze entiteit zijn geen eigenschappen gedefinieerd.

## Organisatie

### Beschrijving

Een aantal mensen georganiseerd in een gemeenschap of andere sociale, commerciële of politieke structuur. De groep heeft een gemeenschappelijk doel of bestaansreden die de individuele leden ervan overstijgt en kan handelen als Agent. Een organisatie heeft dikwijls een hiërarchische structuur.

### Eigenschappen

Voor deze entiteit zijn de volgende eigenschappen gedefinieerd: [voorkeursnaam](#).

| Eigenschap                    | Verwacht Type              | Kardinaliteit | Beschrijving                                              | Gebruik | Codelijst |
|-------------------------------|----------------------------|---------------|-----------------------------------------------------------|---------|-----------|
| <a href="#">voorkeursnaam</a> | <a href="#">Taalstring</a> | 1             | Naam waarmee de organisatie bij voorkeur wordt aangeduid. |         |           |

- 9. VoegSemantiekToe (aanData)
  - In [scenario1](#) deden we dit:
 

```
{
```



```

"@context": {
  "Organisatie": "http://www.w3.org/ns/org#Organization",
  "naam": "http://www.w3.org/2004/02/skos/core#prefLabel"
},
"@graph": [
  {
    "@id": "https://voorbeeld.be/id/organisatie/123",
    "@type": "Organisatie",
    "naam": {
      "@language": "nl",
      "@value": "FOMU"
    }
  },
  {
    "@id": "https://voorbeeld.be/id/organisatie/456",
    "@type": "Organisatie",
    "naam": {
      "@language": "nl",
      "@value": "MIMA"
    }
  }
]
}

```

- o MAAR: in dit scenario stemmen we af op bestaande OSLO-standaarden.
- o In [scenario1](#) werd gesproken over het aanpassen van veldnamen zodat naar de contextfile van OSLO-Organisatie kan verwezen worden:

```

{
  "@context":
    "https://data.vlaanderen.be/context/organisatie-
    basis.jsonld",
  "@graph": [
    {
      "@id": "https://voorbeeld.be/id/organisatie/123",
      "@type": "Organisatie",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://voorbeeld.be/id/organisatie/456",
      "@type": "Organisatie",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}

```

- o MAAR: We maakten nu eigen specs en nevenproduct daarvan is oa een [contextfile](#) met daarin:

```

{
  "@context": {
    "Museum": "https://icomvl.be/ns/museum#Museum",
    "Organisatie": "http://www.w3.org/ns/org#Organization",
    "voorkeursnaam": {
      "@id":
        "http://www.w3.org/2004/02/skos/core#prefLabel",
      "@type": "http://www.w3.org/1999/02/22-rdf-syntax-
        ns#langString"
    }
  }
}

```

- o Maw, we kunnen de oorspronkelijke JSON nu verrijken met de semantiek van onze eigen specs:

```

{
  "@context": "https://otl-
  test.data.vlaanderen.be/doc/applicatieprofiel/museum-
  basis/zonderstatus/20190917/context/museum-basis.jsonld",
  "@graph": [
    {
      "@id": "https://voorbeeld.be/id/organisatie/123",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://voorbeeld.be/id/organisatie/456",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}

```

#### 10. VoegUrisToe

- o Gebeurde al in stap 9 VoegSemantiekToe (aanData) op de manier beschreven in [scenario1](#).
- o MAAR: in de veronderstelling dat deze specs door ICOMVL worden ontwikkeld doen we beter dit:

```

{
  "@context": "https://otl-
  test.data.vlaanderen.be/doc/applicatieprofiel/museum-
  basis/zonderstatus/20190917/context/museum-basis.jsonld",
  "@graph": [
    {
      "@id": "https://icomvl.be/id/museum/123",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://icomvl.be/id/museum/456",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}

```

#### 11. ValideerData

- OPTIONEEL: Maar wel zo handig om 1) de output vd service te ontwikkelen (analist) OF 2) de output te testen (ontwikkelaar), om 3) aangeleverde data te valideren (leveranciers).
- In [scenario1](#) valideerde we tov een bestaande OSLO-standaard, hier gaan we valideren tov onze eigen spec.
- Om die reden kunnen we de online OSLO-Validator niet gebruiken, die is louter voor de OSLO-standaarden geconfigureerd. We gaan daarom de OSLO-Validator lokaal installeren en configureren.
- Code vd webapp maar ook een Docker-container is beschikbaar in de github repository [OSLO-Validator](#).

- Check eerst of Docker geïnstalleerd is als volgt:

```
C:\Users\thijsge>docker -v
Docker version 18.09.2, build 6247962
```

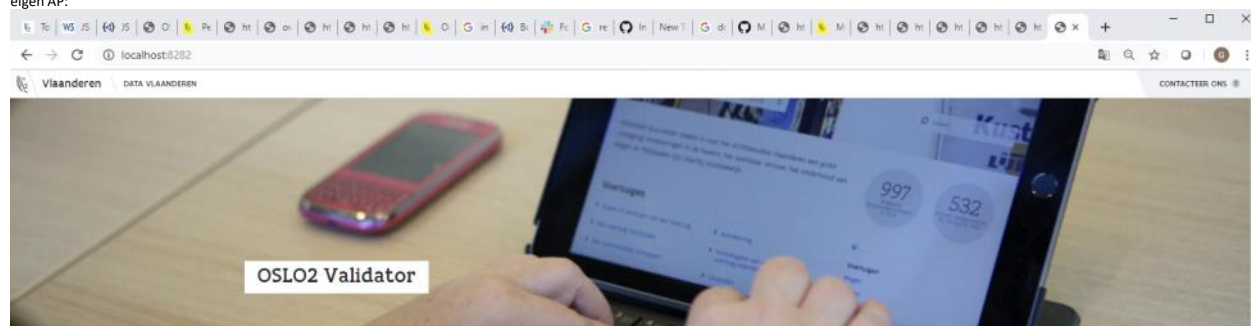
- Parameter vd container is een configuratiefile, <https://raw.githubusercontent.com/GeertThijs/MyFiles/master/demo-ValideerData-config.json> bv zie eruit als volgt:

```
{
  "Museum-basis": {
    "location": "https://raw.githack.com/Informatievlaanderen/OSLO-Generated/test-feature-checkout/doc/applicatieprofiel/museum-basis/zonderstatus/20190917/shacl/museum-basis-SHACL.ttl",
    "dependencies": [
      "https://raw.githack.com/Informatievlaanderen/OSLO-Generated/test-feature-checkout/doc/vocabulary/museum/zonderstatus/20190917/voc/museum.ttl"
    ]
  }
}
```

- Waarbij:

- "location" wijst naar de SHACL-file.
- "dependencies" wijst naar de RDF-file vh VOC.

- OPMERKING: Beide files zijn nevenproducten van stap 6 MaakSpecs.
- OPMERKING: De oorspronkelijke URL is in feite <https://raw.githubusercontent.com> etc maar wordt vervangen door <https://raw.githack.com> omdat anders de extensie ttl niet gelezen wordt en het MIME type tekst/plain ipv tekst/turtle wordt toegepast.
- OPMERKING: De noodzaak om in de config file niet enkel naar de SHACL-file te verwijzen maar ook naar de RDF-file vh VOC (via het keyword "dependencies") wordt duidelijk verderop.
- Run nu de Docker-container als volgt:
- ```
C:\Users\thijsge>docker run -p 8282:8080 -e AP_CONFIG=https://raw.githubusercontent.com/GeertThijs/MyFiles/master/demo-ValideerData.json informatievlaanderen/oslo-shacl-validator:latest
```
- ```
(docker run -p 8282:8080 -e AP_CONFIG=https://raw.githubusercontent.com/GeertThijs/MyFiles/master/demo-ValideerData-config.json informatievlaanderen/oslo-shacl-validator:latest)
```
- De OSLO-validator is nu te vinden op <http://localhost:8282/> met in het valmenu ons eigen AP:



Valideer via opladen van bestand | Valideer via URL

## Valideer via opladen van bestand

### OSLO<sup>2</sup> applicatie profiel

Selecteer hieronder het OSLO<sup>2</sup> applicatie profiel waartegen u uw data wil valideren.

Museum-basis

### Data

Selecteer hieronder het bestand dat u wil valideren.

[Bijlage toevoegen](#)

### Valideer

Klik hier om te valideren

generiek.rdf

Show all

- Als we volgend bestand valideren:

```
{
  "@context": "https://otl-test.data.vlaanderen.be/doc/applicatieprofiel/museum-basis/zonderstatus/20190917/context/museum-basis.jsonld",
  "@graph": [
    {
      "@id": "https://icomv1.be/id/museum/123",
      "@type": "Museum"
    },
    {
      "@id": "https://icomv1.be/id/museum/456",
      "@type": "Museum"
    }
  ]
}
```

- Krijgen we volgende fouten:

| Focus Node                                                                    | Result Message     | Result Path                                                                                               | Result Severity                                                                         | Source Constraint | Source Constraint Component                                                                                                 | Source Shape | Value |
|-------------------------------------------------------------------------------|--------------------|-----------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------|-------------------|-----------------------------------------------------------------------------------------------------------------------------|--------------|-------|
| <a href="https://icomvl.be/id/museum/123">https://icomvl.be/id/museum/123</a> | Less than 1 values | <a href="http://www.w3.org/2004/02/skos/core#prefLabel">http://www.w3.org/2004/02/skos/core#prefLabel</a> | <a href="http://www.w3.org/ns/shacl#Violation">http://www.w3.org/ns/shacl#Violation</a> | NA                | <a href="http://www.w3.org/ns/shacl#MinCountConstraintComponent">http://www.w3.org/ns/shacl#MinCountConstraintComponent</a> | _b0          | NA    |
| <a href="https://icomvl.be/id/museum/456">https://icomvl.be/id/museum/456</a> | Less than 1 values | <a href="http://www.w3.org/2004/02/skos/core#prefLabel">http://www.w3.org/2004/02/skos/core#prefLabel</a> | <a href="http://www.w3.org/ns/shacl#Violation">http://www.w3.org/ns/shacl#Violation</a> | NA                | <a href="http://www.w3.org/ns/shacl#MinCountConstraintComponent">http://www.w3.org/ns/shacl#MinCountConstraintComponent</a> | _b0          | NA    |

Reden: de van OSLO-Organisatie:Organisatie overgeërfde voorkeursnaam ontbreekt.

- Trachten we daarentegen dit te valideren:

```
{
  "@context": "https://otl-test.data.vlaanderen.be/doc/applicatieprofiel/museum-basis/zonderstatus/20190917/context/museum-basis.jsonld",
  "@graph": [
    {
      "@id": "https://icomvl.be/id/museum/123",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://icomvl.be/id/museum/456",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}
```

Dan zijn er 0 fouten:

| Focus Node                 | Result Message | Result Path | Result Severity | Source Constraint | Source Constraint Component | Source Shape | Value |
|----------------------------|----------------|-------------|-----------------|-------------------|-----------------------------|--------------|-------|
| No data available in table |                |             |                 |                   |                             |              |       |

- OPMERKING: De configuratiefile voor de validator bevat niet enkel een verwijzing naar de SHACL-file maar ook naar het VOC (via het keyword "dependencies"), in dit vb verwijzen we naar <https://raw.githubusercontent.com/Informatievlaanderen/OSLO-Generated/test-feature-checkout/doc/vocabulary/museum/zonderstatus/20190917/voc/museum.ttl>.
- Reden hiervoor is dat de SHACL-file informatie geen info mbt overerving bevat, die staat immers al in het VOC.
- Zouden we volgende configuratiefile gebruiken:

```
{
  "Museum-basis": {
    "location": "https://raw.githubusercontent.com/Informatievlaanderen/OSLO-Generated/test-feature-checkout/doc/applicatieprofiel/museum-basis/zonderstatus/20190917/shacl/museum-basis-SHACL.ttl",
    "dependencies": [ ]
  }
}
```

- Dan zou dit bestand perfect valideren:

```
{
  "@context": "https://otl-test.data.vlaanderen.be/doc/applicatieprofiel/museum-basis/zonderstatus/20190917/context/museum-basis.jsonld",
  "@graph": [
    {
      "@id": "https://icomvl.be/id/museum/123",
      "@type": "Museum"
    },
    {
      "@id": "https://icomvl.be/id/museum/456",
      "@type": "Museum"
    }
  ]
}
```

- Reden: de validator dedecteert niet dat een Museum een subklasse is van Organisatie en dus het verplicht veld voorkeursnaam overerft.

- 12. VoegSemantiekToe (aanService)
  - TODO

Uiteindelijk resultaat van dit scenario:

```
{
  "@context": "https://otl-test.data.vlaanderen.be/doc/applicatieprofiel/museum-basis/zonderstatus/20190917/context/museum-basis.jsonld",
  "@graph": [
    {
      "@id": "https://icomvl.be/id/museum/123",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    },
    {
      "@id": "https://icomvl.be/id/museum/456",
      "@type": "Museum",
      "voorkeursnaam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}
```

Ter vergelijking het resultaat van [scenario1](#):

```
{
  "@context": {
    "Organisatie": "http://www.w3.org/ns/org#Organization",
    "naam": "http://www.w3.org/2004/02/skos/core#prefLabel"
  },
  "@graph": [
    {
      "@id": "https://voorbeeld.be/id/organisatie/123",
      "@type": "Organisatie",
      "naam": {
        "@language": "nl",
        "@value": "FOMU"
      }
    }
  ]
}
```

```

    },
    {
      "@id": "https://voorbeeld.be/id/organisatie/456",
      "@type": "Organisatie",
      "naam": {
        "@language": "nl",
        "@value": "MIMA"
      }
    }
  ]
}

```

Belangrijkste verschil: @type is nu Museum ipv Organisatie.

Worden deze stappen doorlopen dan is als volgt voldaan aan de checklist (verschillen tov [scenario1](#) in fluo):

| Thema     | Item                                                             | Check |
|-----------|------------------------------------------------------------------|-------|
| URI       | Nemen de id's vd data de vorm aan ve URI?                        | JA    |
|           | Is deze URI in overeenstemming met de URI-richtlijn?             | JA    |
|           | Zijn de URI's PURI's?                                            | NVT   |
| Aanpak    | Werd getoetst aan bestaande standaarden?                         | JA    |
|           | Werd nieuw VOC/AP ontwikkeld volgens OSLO Proces & Methodologie? | NVT   |
|           | Zijn er gepubliceerde specs?                                     | JA    |
| Datamodel | Is er een domeinmodel?                                           | JA    |
|           | Semantiek toegevoegd aan model?                                  | JA    |
|           | Voldoet het model aan de modelleringsregels?                     | OPT   |
| Service   | Is de data verrijkt met de semantiek?                            | JA    |
|           | Kan de data gevalideerd worden?                                  | OPT   |
|           | Voldoet de service aan de API-richtlijn?                         | OPT   |

Discussiepunten:

- ()

# OSLO-compliance demo Scenario3

donderdag 5 september 2019 13:50