

OSLO LDES: Thematische werkgroep 2

Welkom!

Donderdag 17 februari

Virtuele werkgroep – Microsoft Teams

We starten om 09:05



Doel van vandaag

**Bespreken van de gemaakte aanpassingen aan het data model
en verzamelen van feedback op de nieuwe iteratie**



**Samenvatting van de
eerste thematische
werkgroep**



**Nieuwe aanpassingen
aan het datamodel**



**Capteren van input
a.d.h.v. Mural**

Agenda



Welkom en introductie

09:05 - 09:15



Overzicht: wat hebben we gedaan
in de vorige werkgroep?

09:15 - 09:20



Aanpassingen aan het datamodel

09:25 - 09:35



Versiebeheer en relaties tussen LDES'en

09:35 - 10:25



Navigatie doorheen een LDES

10:35 - 11:25



Brainstorm oefening en volgende stappen

11:25 - 12:00

Praktische zaken

De microfoon van deelnemers staat altijd op **mute**



Steek je **hand** op indien je iets wil zeggen. Interactie wordt aangemoedigd!



Video camera aangemoedigd

Vragen en suggesties kunnen ook steeds gecommuniceerd worden via de **chat** functie.



ja/nee vragen kan je beantwoorden via de chat:

Ja = +1
Neen = - 1
Neutraal = 0

Opname?



Vlaanderen
verbeelding werkt

Welkom en introductie

Mural

Wie is wie?



Link in de chat:

<https://app.mural.co/t/beadvtc7549/m/beadvtc7549/1644223871728/ce828415d3ea921bbfd1f745cab81928966ce9ee?sender=u048a1117151baed084666519>

Samenvatting: Wat hebben we gedaan in de
vorige werkgroep?

Wat hebben we gedaan in de vorige werkgroep?



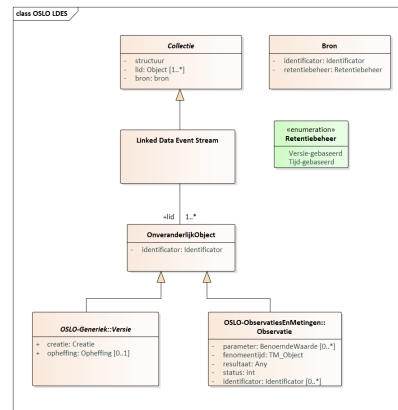
Herhaling business werkgroep

- OSLO Introductie
- Introductie tot LDES
- Brainstorm oefeningen



SEMIC specificatie

- Introductie
- Retentiebeleid
- Versiematerialisatie



Eerste poging tot een datamodel

- Basis-definitie LDES
- Collectie/structuur/lid/bron
- OnveranderlijkObject



Scope van het project

Ontwikkelen van een semantisch framework voor een Linked Data Event Stream

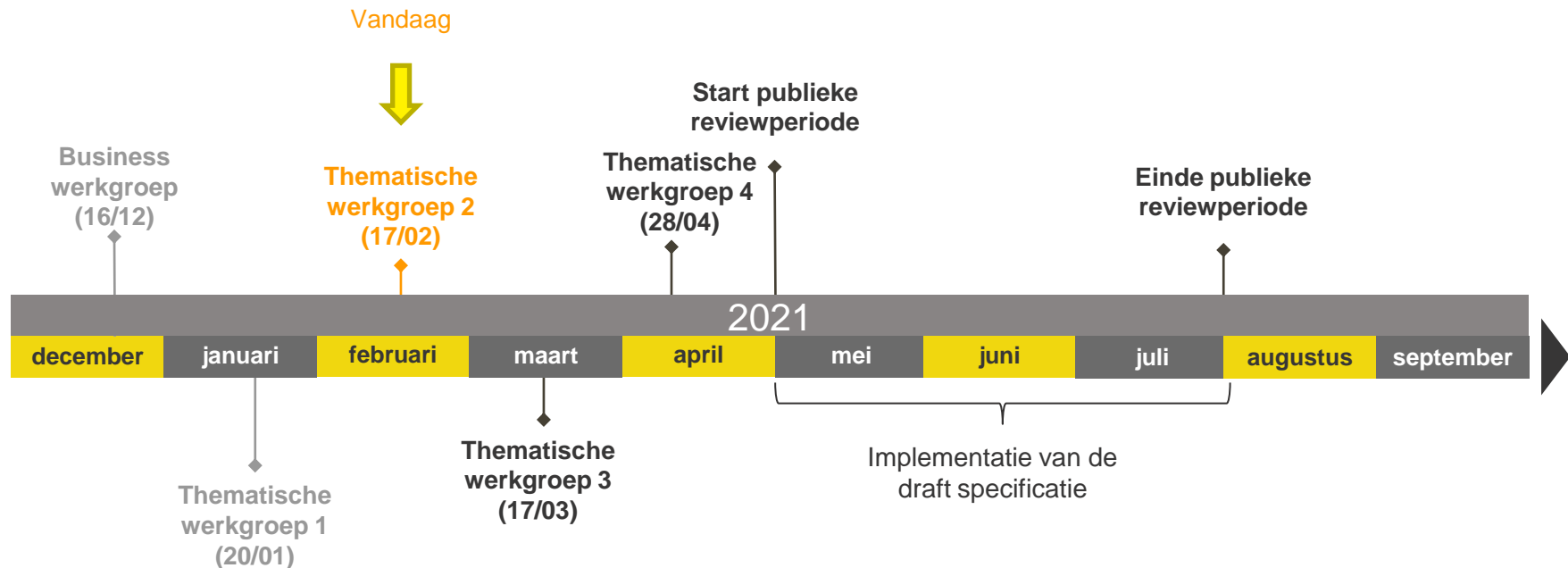
*Omvat een duurzaam **vocabulary** en **applicatieprofiel** voor een Linked Data Event Stream*

We volgen de OSLO methodiek, wat betekent dat:

- We starten van **use cases**
- We zoveel mogelijk aligneren met bestaande standaarden
- We zelf zaken definiëren waar nodig

Overzicht van de planning

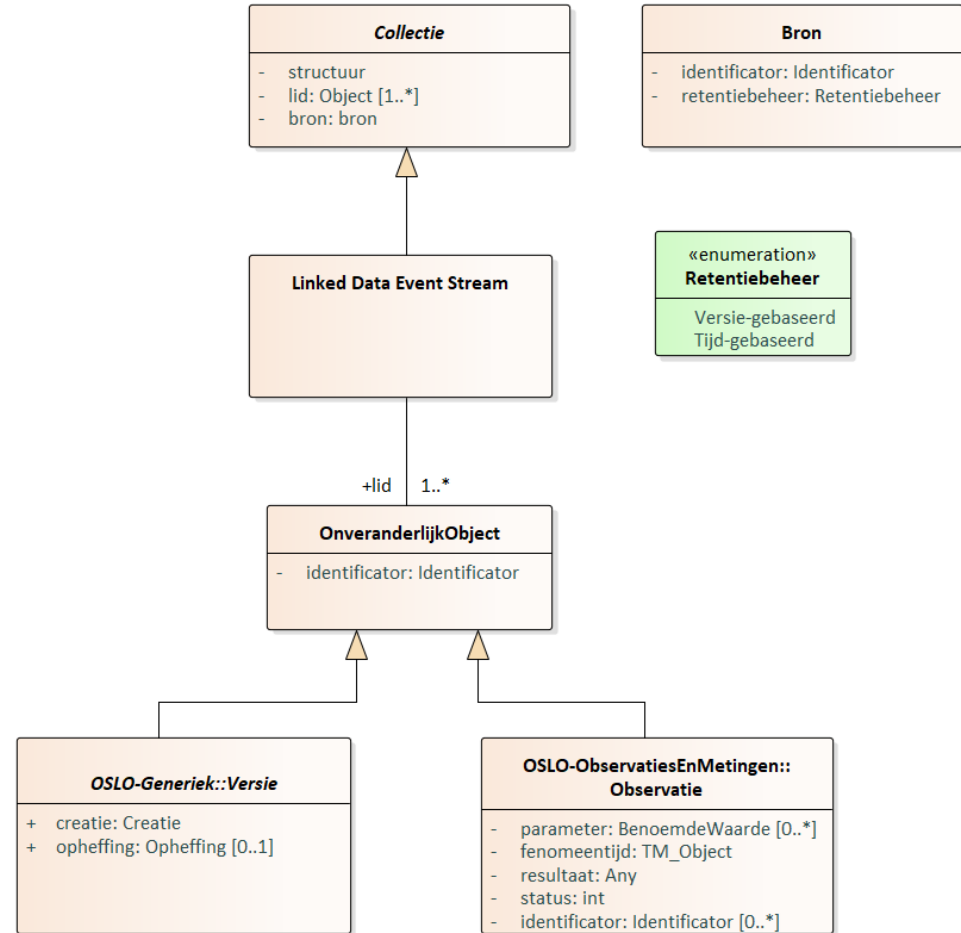
- Startdatum: 16 december 2021
- Duur: 10 maanden



Aanpassingen aan het datamodel o.b.v. feedback

Datamodel TW1

class OSLO LDES



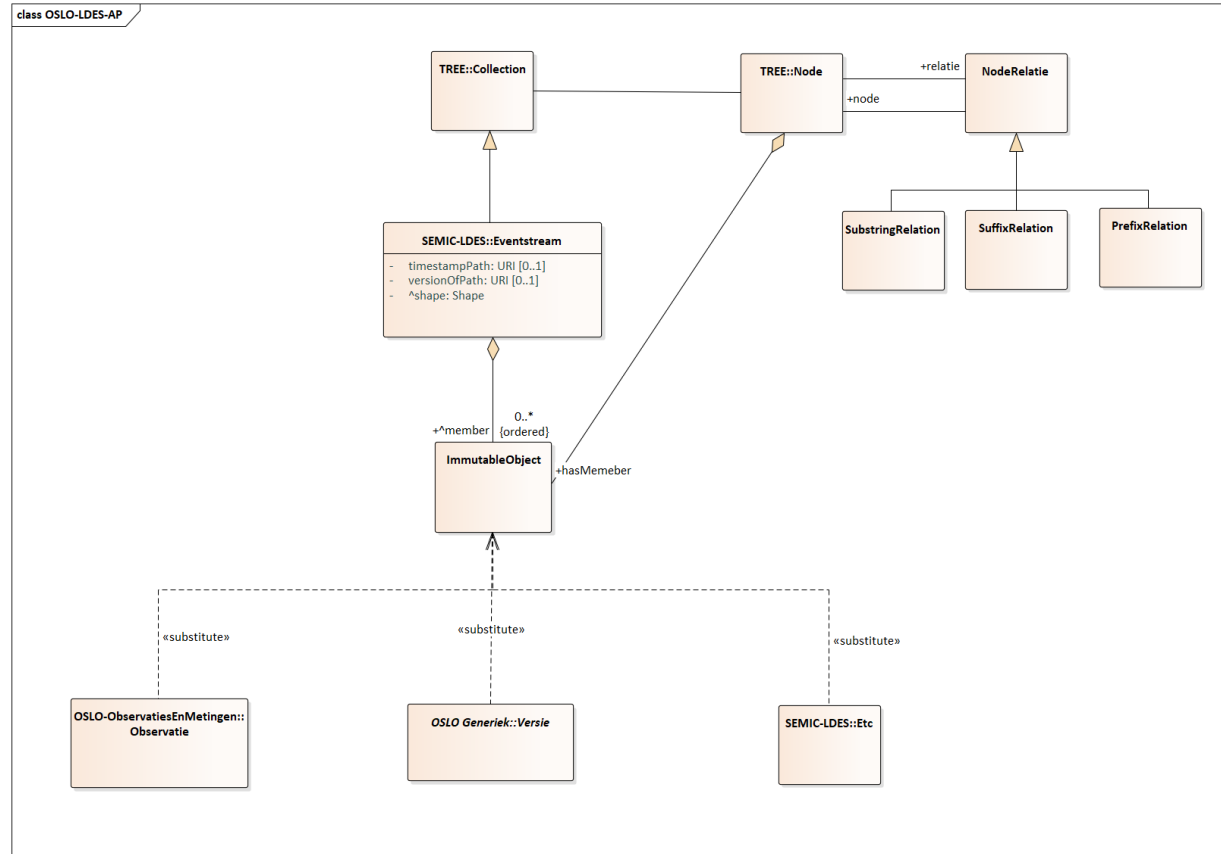
Datamodel TW2

Aanpassingen:

1. Benamingen en definities
2. Observaties
3. Versiebeheer

Toevoegingen:

1. Paginatie
2. Relaties tussen pagina's



1. Benamingen en definities

Benamingen: we zullen werken met de Engelse termen

Definities: volgende definities werden aangepast:

- Basis-definitie LDES
- Shape
- View
- Node
- Relation

OSLO:Observatie & OSLO:Versie

OSLO:Observatie

- Komt uit OSLO-Observaties & Metingen
- Gebaseerd op ISO Observations & Measurements (ISO19156)
- Basis voor OSLO Sensoren & Bemonstering, gebaseerd op W3C Semantic Sensor Network Ontology (W3C SSN/SOSA)
- Beide worden gebruikt in OSLO Bodem & Ondergrond en OSLO Waterkwaliteit
- ISO19156 en W3C SSN/SOSA tevens de basis voor ODALA Air & Water

DEFINITIE

```

classDiagram
    class Object {
        +geobserveerdObject 1
    }
    class Bemonsteringsobject {
        +lid 1..*
        - herkomst: LI_Lineage [0..1]
        - parameter: BenoemdeWaarde [0..*]
        - identifier: Identifier [1..*]
        - type: Bemonsteringsobjecttype [0..1]
    }
    class Kenmerktype {
        +geobserveerdKenmerk 1
    }
    class Observatie {
        - parameter: BenoemdeWaarde [0..*]
        - fenomeentijd: TM_Object
        - resultaatijd: TM_Instant
        - geldigeTijd: TM_Period [0..1]
        - resultaatkwaliteit: DQ_Element [0..*]
        - resultaat: GeneriekDatatype
        - opdracht: Opdracht [0..*]
        - uitgevoerdDoor: Agent [0..*]
        - type: Observatietype [0..*]
        - identifier: Identifier [0..*]
        +gegeven 0..*
        +geassocieerdeObservatie 0..*
    }
    Object "1" -- "1..*" Bemonsteringsobject : +bemonsterdObject
    Object "1" -- "0..*" Kenmerktype : +geobserveerdObject
    Kenmerktype "1" -- "0..*" Observatie : Fenomeen
    Observatie "0..*" -- "0..*" Bemonsteringsobject : Ontwerp
    Observatie "0..*" -- "0..*" Observatie : +geassocieerdeObservatie
    Observatie "0..*" -- "0..*" Observatie : +gegeven
    
```

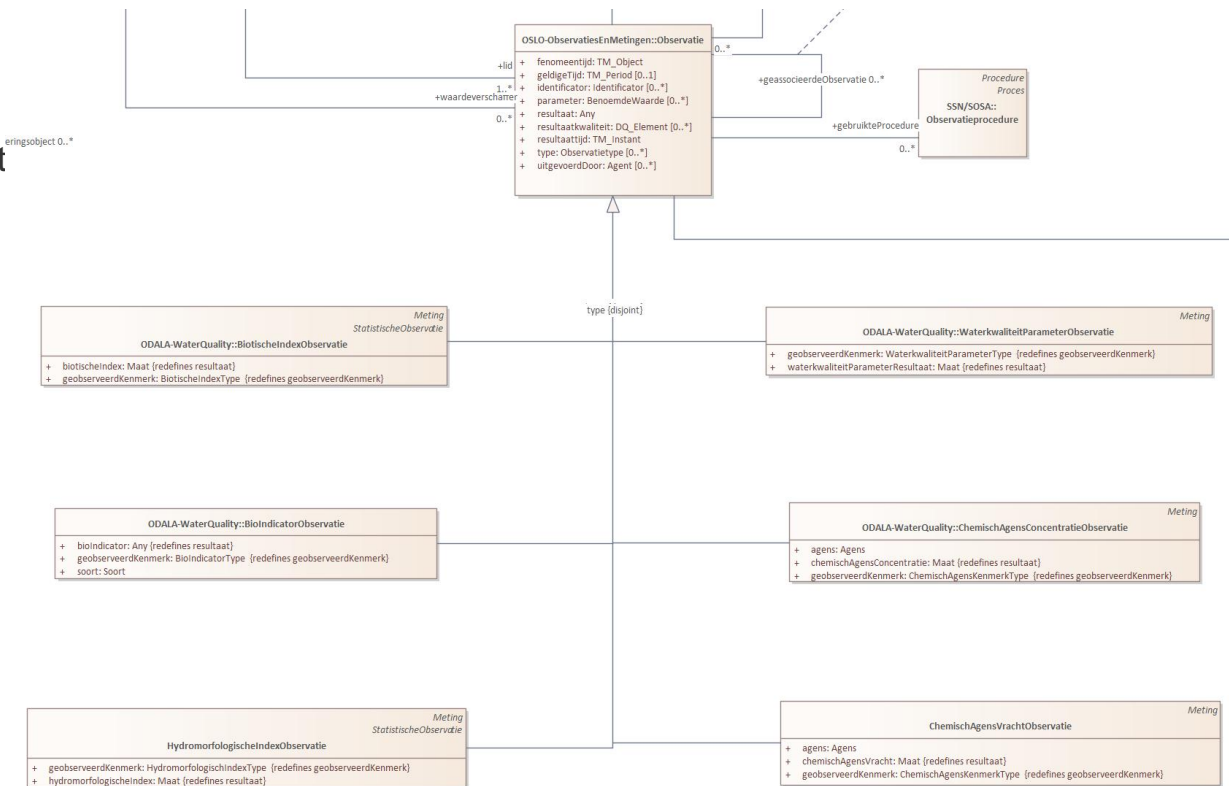
The diagram illustrates the relationships between four main classes: **Object**, **Bemonsteringsobject**, **Kenmerktype**, and **Observatie**.

- Object** (top left) has an association with **Bemonsteringsobject** (bottom left) labeled **+bemonsterdObject** with multiplicity 1 at Object and 1..* at Bemonsteringsobject. It also has an association with **Kenmerktype** (top right) labeled **+geobserveerdObject** with multiplicity 1 at Object and 0..* at Kenmerktype.
- Bemonsteringsobject** (bottom left) is a specialization of **Object** (indicated by a solid line with an open triangle). It has attributes: **+lid** (multiplicity 1..*), **- herkomst: LI_Lineage** [0..1], **- parameter: BenoemdeWaarde** [0..*], **- identifier: Identifier** [1..*], and **- type: Bemonsteringsobjecttype** [0..1].
- Kenmerktype** (top right) has an association with **Observatie** (bottom right) labeled **Fenomeen** with multiplicity 1 at Kenmerktype and 0..* at Observatie.
- Observatie** (bottom right) is a specialization of **Kenmerktype** (indicated by a solid line with an open triangle). It has attributes: **- parameter: BenoemdeWaarde** [0..*], **- fenomeentijd: TM_Object**, **- resultaatijd: TM_Instant**, **- geldigeTijd: TM_Period** [0..1], **- resultaatkwaliteit: DQ_Element** [0..*], **- resultaat: GeneriekDatatype**, **- opdracht: Opdracht** [0..*], **- uitgevoerdDoor: Agent** [0..*], **- type: Observatietype** [0..*], and **- identifier: Identifier** [0..*]. It also has associations with **Bemonsteringsobject** labeled **Ontwerp** (multiplicity 0..* at both) and with itself labeled **+geassocieerdeObservatie** (multiplicity 0..* at both). Additionally, it has an association with **Kenmerktype** labeled **+gegeven** (multiplicity 0..* at Observatie).
- Observatie** has a **constraints** section: **(gebovserveerdKenmerk consistent met type geobserveerdObject)** and **(resultaattype & procedure consistent met geobserveerdKenmerk)**.

OSLO:Observatie

SPECIALISATIES

Vb voor OSLO-Waterkwaliteit ^{eringsobject 0..*}



OSLO:Observatie {

VOORBEELD

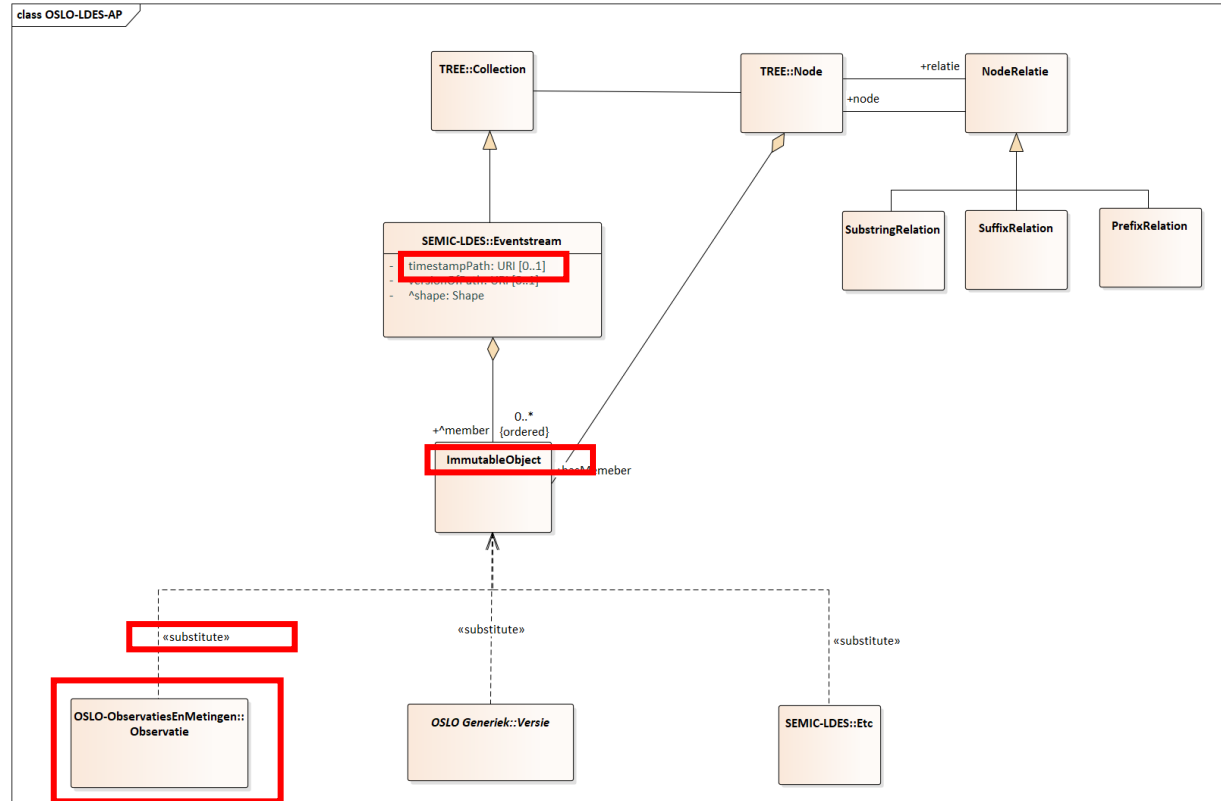
- Observatie van temperatuur ve Bemonsteringspunt in het kanaal Sluis-Brugge met Sensor01
- OSLO:Meting (ISO19103:Measure) = specialisatie van OSLO:Observatie

```
{
  "@context": [
    "ap-observaties-en-metingen.jsonld",
    "ap-sensoren-en-bemonstering.jsonld",
    {
      "qudt-schema": "http://qudt.org/schema/qudt/",
      "qudt-vocab": "http://qudt.org/vocab/quantitykind/",
      "hyf": "https://www.opengis.net/def/schema/hy_features/hyf/"
    }
  ],
  "@id": "ex:Observatie01",
  "@type": "Meting",
  "Observatie.geobserveerdKenmerk": {
    "@id": "qudt-vocab:Temperature",
    "@type": "qudt-schema:QuantityKind"
  },
  "Observatie.geobserveerdObject": {
    "@id": "ex:Station01",
    "@type": "Bemonsteringspunt",
    "RuimtelijkBemonsteringsobject.geometrie": {
      "@type": "Punt",
      "Geometrie.gml": "<gml:Point srsName=<\"http://www.opengis.net/def/crs/EPSSG/0/31370\">
        <gml:pos>33953.76 205257.06</gml:pos></gml:Point>"
    },
    "Bemonsteringsobject.bemonsterdObject": {
      "@id": "http://data.vmm.be/id/kanaal/L107_123",
      "@type": "hyf:HY_Canal"
    }
  },
  "Observatie.fenomeentijd": "2021-01-01-00:00:00Z\\\"^^xsd:dateTime",
  "Meting.resultaat": {
    "@type": "Maat",
    "Maat.maat": {
      "@type": "KwantitatieveWaarde",
      "KwantitatieveWaarde.waarde": "21",
      "KwantitatieveWaarde.standaardEenheid": {
        "@id": "qudt-vocab:DEG_C",
        "@type": "qudt-schema:Unit"
      }
    }
  },
  "Observatie.uitgevoerdMetsensor": {
    "@id": "ex:Sensor01",
    "@type": "Sensor"
  }
}
```

OSLO:Observatie & LDES

DATAMODEL

- OSLO:Observatie is Event & dus immutable
- Substitute ipv generalisatie
- Eventstream.timestampPath = Observatie.fenomeentijd of Observatie.resultaattijd



OSLO:Observatie & LDES

VOORBEELD

```
{
  "@context": [
    "ap-observaties-en-metingen.jsonld",
    {
      "tree": "https://w3id.org/tree#",
      "ldes": "https://w3id.org/ldes#"
    }
  ],
  "@id": "",
  "@type": "ldes:EventStream",
  "tree:shape": "",
  "ldes:timestampPath": "Observatie.fenomeentijd",
  "tree:member": [
    "ex:Observatie01",
    "ex:Observatie02"
  ]
}
```

OSLO:Observatie & LDES

ISSUES/VRAGEN/OPMERKINGEN

- Observatie.fenomeentijd is vh type ISO19108:TM_object (dus niet louter timestamp)
- Alternatieven voor LDES in OSLO-Observaties & Metingen en OSLO-Sensoren & Bemonstering:
 - Observatieverzameling (SOSA:ObservationCollection)
 - Slice (Cube:Slice)
 - TemporeelMomentBereik (ISO1965:CVT_DiscreteTimeInstantCoverage)
- SOSA:Observation.hasSimpleresult toevoegen aan OSLO Observaties & Metingen?
- ...

OSLO:Versie

- Komt uit [OSLO-Generiek](#)
- Gebaseerd op [W3C PROV Ontology \(PROV-O\)](#)

OSLO:Versie

DEFINITIE

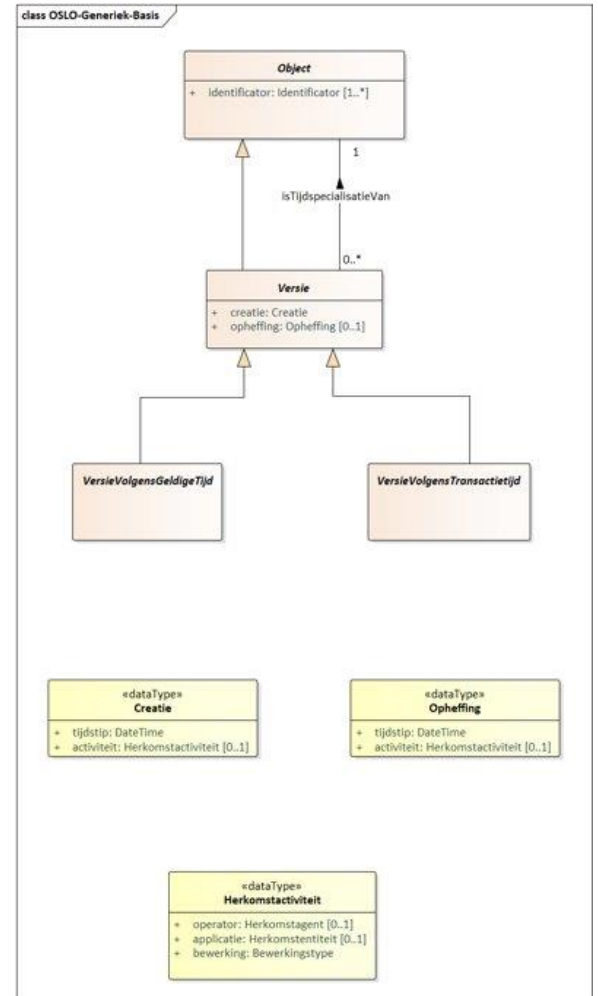
Versie: Stelt een entiteit voor in één specifieke periode.

VersieVolgensGeldigeTijd:

De versie stelt de entiteit voor gedurende een bepaalde periode in de gemodelleerde werkelijkheid.

VersieVolgensTransactieTijd:

De versie stelt de entiteit voor gedurende een bepaalde periode in een databank.



OSLO: Versie

VB ADRES

- Beschrijving ve Versie v/e Adres, Adres wordt bijkomend getypeerd als VersieVolgensTransactietijd (id = versienummer)
- => bijkomende attributen
- Ook info over VersieGeldigeTijd (dus eigenlijk bitemporeel)

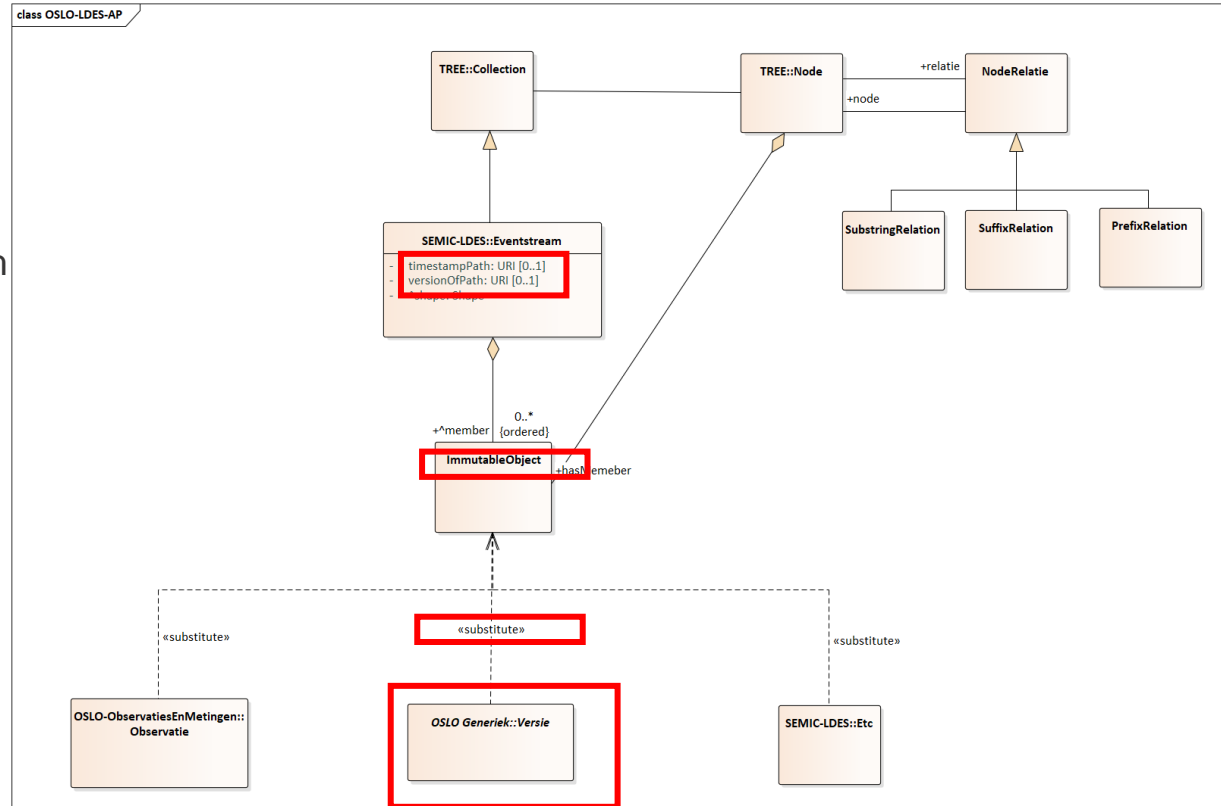
```

"@context": [
  "adresregister.jsonld",
  "generiek-basis.jsonld"
],
"@id": "https://data.vlaanderen.be/id/concept/adres/823dd4a003c1369beef3b31ffea4862ff7d82d2a",
"@type": [
  "Adres",
  "VersieVolgensTransactietijd",
  "versievolgensGeldigeTijd"
],
"Adres.busnummer": "0101",
"Adres.huisnummer": "281",
"heeftStraatnaam": "https://data.vlaanderen.be/id/straatnaam/61036",
"heeftGemeentenaam": {
  "@type": "Gemeentenaam",
  "gemeentenaam": {
    "@language": "nl",
    "@value": "Nieuwpoort"
  }
},
"isAfgeleidVan": "https://data.vlaanderen.be/id/gemeente/38016"
},
"heeftPostinfo": {
  "@type": "Postinfo",
  "postcode": "8620"
},
"isToegekendDoor": "https://data.vlaanderen.be/id/gemeente/38016",
"positie": {
  "@type": "GeografischePositie",
  "GeografischePositie.default": true,
  "GeografischePositie.geometrie": {
    "@type": "Punt",
    "Geometrie.gml": "<gml:Point srsName=<'http://www.opengis.net/def/crs/EPSG/0/31370'><gml:pos>33953.76 205257.06</gml:pos></gml:Point>"
  },
  "GeografischePositie.methode": "https://data.vlaanderen.be/id/conceptscheme/geometriemethode/afgeleidVanObject",
  "GeografischePositie.specificatie": "https://data.vlaanderen.be/id/conceptscheme/geometriespecificatie/gebouwenheid"
},
"officieelToegekend": true,
"status": "https://data.vlaanderen.be/id/concept/adresstatus/inGebruik",
"Versie.isIjdspecialisatieVan": "https://data.vlaanderen.be/id/adres/30012805",
"VersieVolgensTransactietijd.creatie": {
  "@type": "VersieVolgensTransactietijd",
  "Creatie.tijdstip": "\"2021-10-22T16:13:04.032Z\"^^xsd:dateTime"
},
"VersieVolgensGedigeTijd.creatie": {
  "@type": "VersieVolgensGedigeTijd",
  "Creatie.tijdstip": "\"2002-08-13T16:33:18+02:00\"^^xsd:dateTime",
  "Creatie.activiteit": {
    "Herkomstactiviteit.bewerking": "AddressPersistentLocalIdentifierWasAssigned"
  }
}
}
```

OSLO:Versie & LDES

DATAMODEL

- OSLO:Versie als Event & ImmutableObject
- Substitute ipv generalisatie
- Eventstream.timestampPath = Versie.creatie -> Creatie.tijdstip



OSLO:Versie & LDES

VOORBEELD

```
{
  "@context": [
    "generiek-basis.jsonld",
    {
      "tree": "https://w3id.org/tree#",
      "ldes": "https://w3id.org/ldes#"
    }
  ],
  "@id": "",
  "@type": "ldes:EventStream",
  "tree:shape": "",
  "ldes:timestampPath": "Creatie.tijdstip",
  "tree:member": [
    "ex:Adres01-v01",
    "ex:Adres01-v02"
  ]
}
```

OSLO:Versie & LDES

ISSUES/VRAGEN/OPMERKINGEN

- Versie is eigenlijk geen Event maar een Object
- Is Versie immutable?
 - OnvoltooideVersie zeker wel
 - Tot ze voltooid wordt?
- Referenties: naar Object of ook naar Versie?
- Verwerking van gerelateerde Streams?
 - Vb AdresStream en StraatnaamStream
- Kan Object van type veranderen?

OSLO:Versie & LDES

VERGELIJKING

OSLO:Versie	"LDES:Versie"
Unitemporeel, bitemporeel	Default unitemporeel
Default VoltooideVersie	Default OnvoltooideVersie
GeldigeTijd, Transactietijd	Default Transactietijd
(Verschilbestand, snapshot)	EventStream
Ontologie: PROV-O	Configureerbaar
Verbose (meer semantiek)	Niet-verbose (minder semantiek)

OSLO:Versie & LDES

HUIDIGE LDES-IMPLEMENTATIE, VB STRAATNAAM

- Huidige LDES-implementatie Basisregisters
- Object aangevuld met aantal attributen uit PROV-O, DCTERMS & ADMS

```
{  
  "@id":  
    "https://smartdata.dev-vlaanderen.be/base/straatnaam#33428-  
v01",  
  "isVersionOf": "https://data.vlaanderen.be/id/straatnaam/33428",  
  "generatedAtTime": "2000-01-01",  
  "created": "1980-01-01",  
  "eventName": "StreetNameBecameCurrent",  
  "memberOf":  
    "https://smartdata.dev-vlaanderen.be/base/straatnaam",  
  "@type": "Straatnaam",  
  "straatnaam": [  
    {  
      "@value": "Stationsstraat",  
      "@language": "nl"  
    }  
  ],  
  "isToegekendDoor":  
    "https://data.vlaanderen.be/id/gemeente/24038",  
  "status":  
    "https://data.vlaanderen.be/id/concept/straatnaamstatus/inGebruik"  
}
```

OSLO:Versie & LDES

ONVOLTOOIDEVERSIË VS VOLTOOIDEVERSIË

tijd	object-versie	tijdstip	waarde	bewerking
T1	SN123-v1	1/1/1980	Stationsstraat	INSERT
T2	SN123-v2	1/1/2000	Stationstraat	UPDATE
T3	SN123-v3	1/1/2020	Stationstraat	DELETE

OnvoltooideVersies:

- Zonder eindtijdstip
- Invalidatie door volgende bewerking
- Immutable
- 3 Versies, 3 Events

tijd	object-versie	begintijdstip	eindtijdstip	waarde	bewerking
T1	SN123-v1	1/1/1980	NULL	Stationsstraat	INSERT
T2	SN123-v1	1/1/1980	1/1/2000	Stationsstraat	UPDATE
T2	SN123-v2	1/1/2000	NULL	Stationstraat	UPDATE
T3	SN123-v2	1/1/2000	1/1/2020	Stationstraat	DELETE

VoltooideVersies:

- Met eindtijdstip
- Invalidatie door invullen einddtijdstip
- Immutable?
- 2 Versies, 4 Events
- UPDATE = DELETE, gevolgd door INSERT

Relatie(s) tussen verschillende LDES'en

Een 'dataset' wordt een set van event streams

Adressenregister: <https://smartdata.dev-vlaanderen.be/base>

1. LDES van Straatnamen: <https://api.basisregisters.vlaanderen.be/v1/straatnamen>
2. LDES van Postinformatie: <https://smartdata.dev-vlaanderen.be/base/postinfo>
3. LDES van Gemeenten: <https://api.basisregisters.vlaanderen.be/v1/gemeenten>
4. LDES van Adresobjecten: <https://smartdata.dev-vlaanderen.be/base/adres>

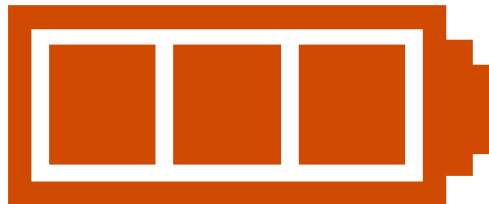
Boodschap:

- Een dataset is een geheel van eventstreams
- Publisher levert geen cohorent product af, maar eventually consistent

Een kleine pauze...



Welkom terug!



Navigatie doorheen een LDES

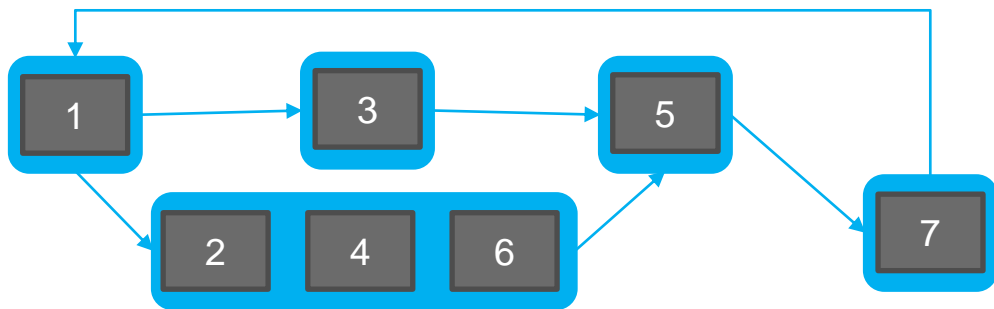




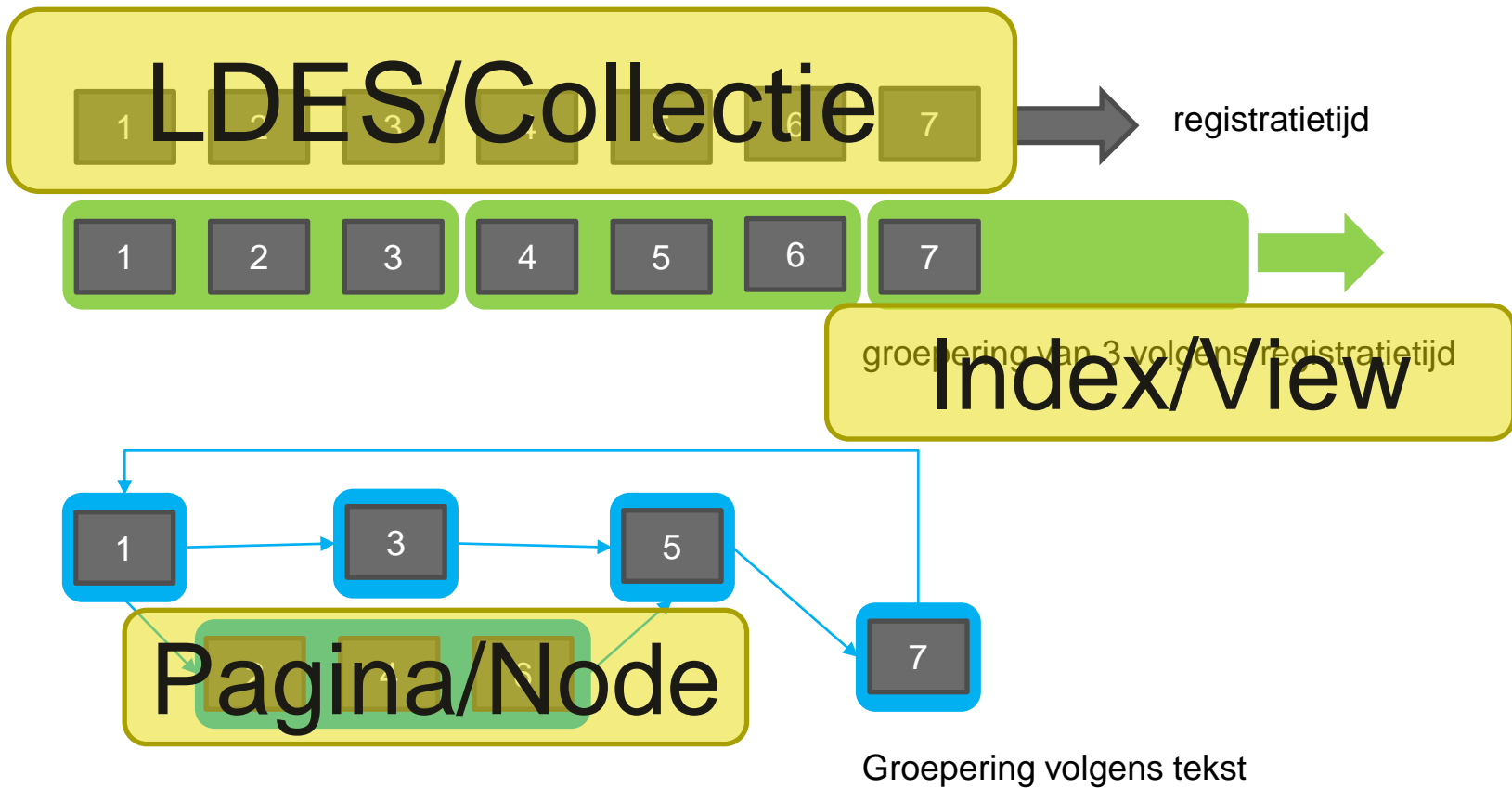
groepering van 3 volgens registratietijd



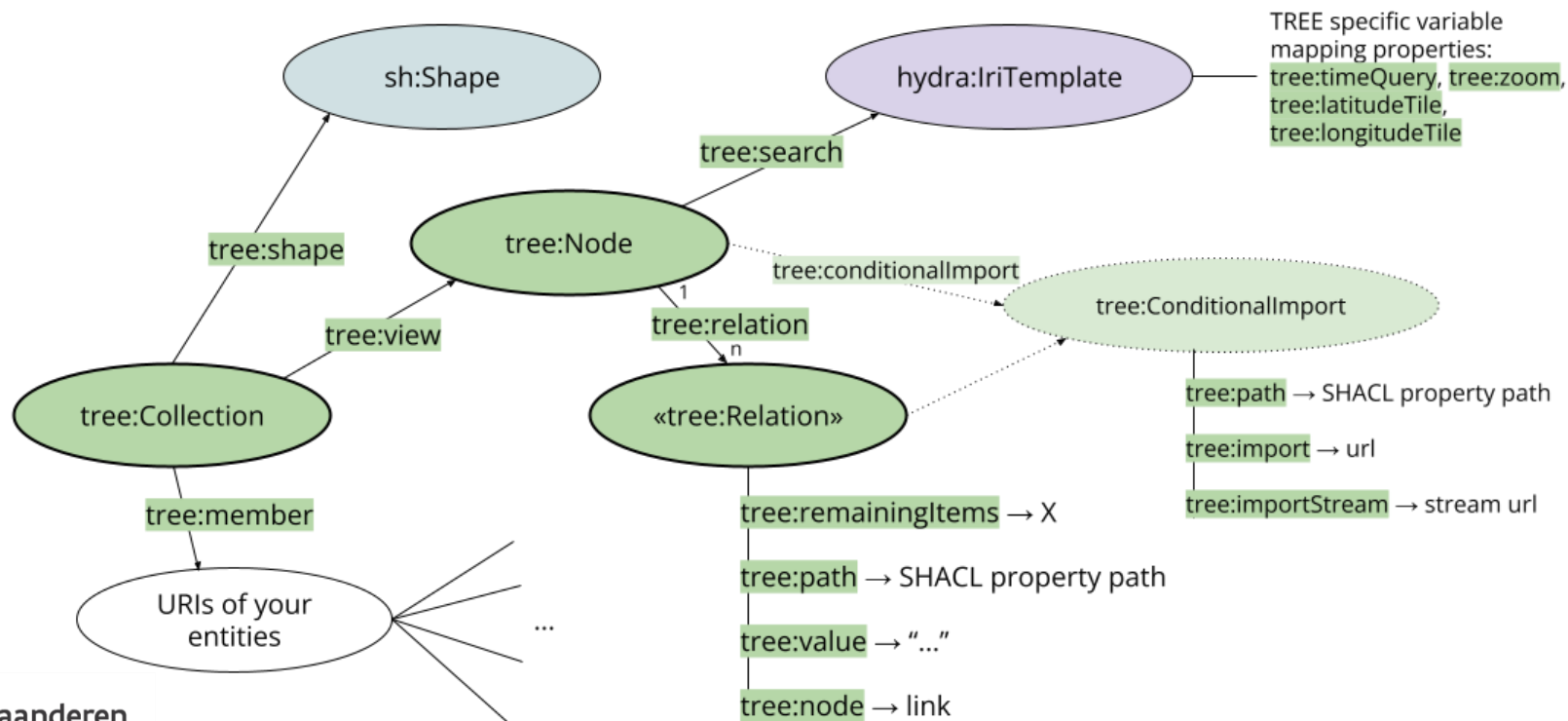
groepering van 3 volgens registratietijd



Groepering volgens azerty



TREE hypermedia specificatie



TREE specificatie definities

<https://w3id.org/tree/specification#introduction>

Collectie (tree:Collection) : een verzameling van elementen die voldoen aan eenzelfde structuur (schema/model/...).

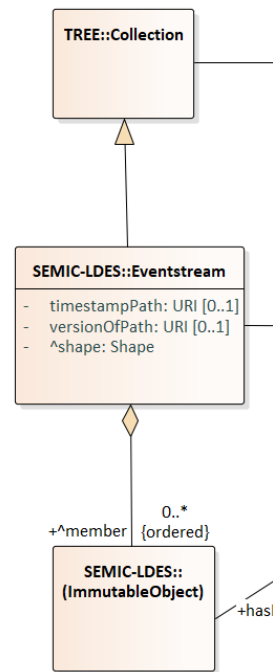
Node (tree:Node) : een pagina die referenties (urls/links) bevat waarmee andere elementen van de collectie gevonden kunnen worden.

Relatie (tree:Relation): een relatie tussen 2 Nodes

LDES specificatie

<https://w3id.org/ldes/specification#introduction>

Eventstream (ldes:Eventstream) : een verzameling van immutable objecten, waarbij de representatievorm van de objecten RDF is.



Disambigueren van View

Definitie 1 : een View is de index/invariant/organisatie waarmee doorheen de collectie (LDES) kan worden genavigeerd

Voorbeeld: View1 is een tekst-index op basis van de labels van de members

Definitie 2: een View is een beginpagina, die zelf al leden kan bevatten, via wiens gelinkte pagina's alle leden kunnen worden gevonden.

Voorbeeld: View1 is de eerste pagina, die zelf al leden kan bevatten, van een hoofdzakelijk op tekst-gebaseerde zoek-structuur



Bijkomende vraag: volledigheid

Moet een index altijd volledig zijn? M.a.w. kan je altijd alle leden van de LDES vinden?

Hoe weten we het verschil tussen volledige en onvolledige indexen?

Bijkomende vraag: volledigheid

Moet een index altijd volledig zijn? M.a.w. kan je altijd alle leden van de LDES vinden?

Hoe weten we het verschil tussen volledige en onvolledige indexen? Moeten we een status-indicatie object hebben?

In LDES is dit deel van de “definitie van een eigenschap view”.

Pagina: elementen

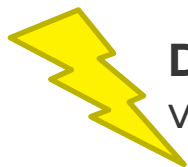
Welke elementen die behoren tot een pagina?

In de TREE-specificatie is er geen expliciete relatie tussen een pagina en zijn elementen: je ziet of een element op de pagina staat door de pagina te downloaden.

Pagina: elementen

Welke elementen die behoren tot een pagina?

In de TREE-specificatie is er geen expliciete relatie tussen een pagina en zijn elementen: je ziet of een element op de pagina staat door de pagina te downloaden.



Definitie 2: een View is een beginpagina, die zelf al leden kan bevatten, via wiens gelinkte pagina's alle leden kunnen worden gevonden.

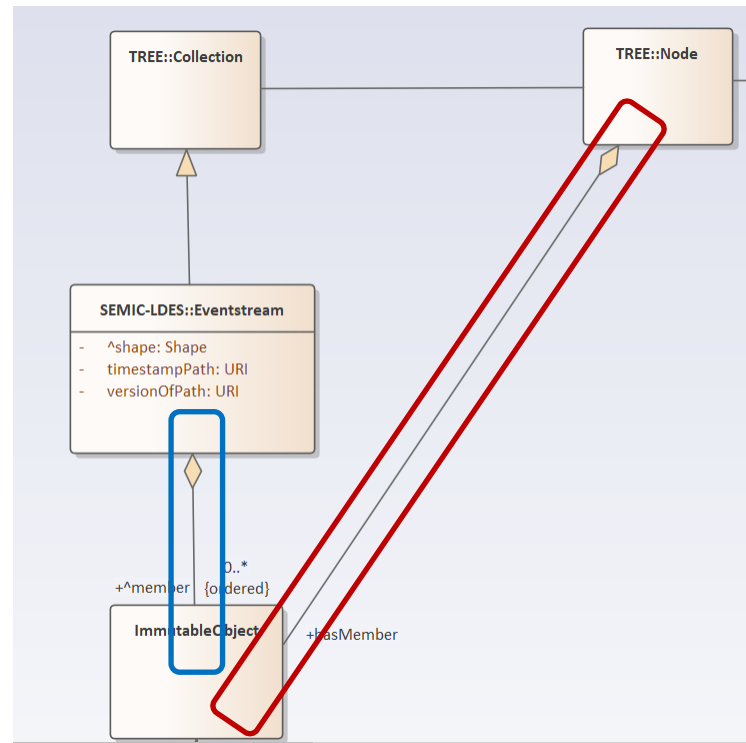
Pagina: elementen

hasMember is een subproperty van *member*.

Maar is deze relatie nodig?

- Het is niet zo dat de lijst van leden voor een bepaalde node vast staat in de tijd.
- Leden kunnen verdwijnen van de pagina.

voorbeeld: Index = bedrijven volgens NACEcode.



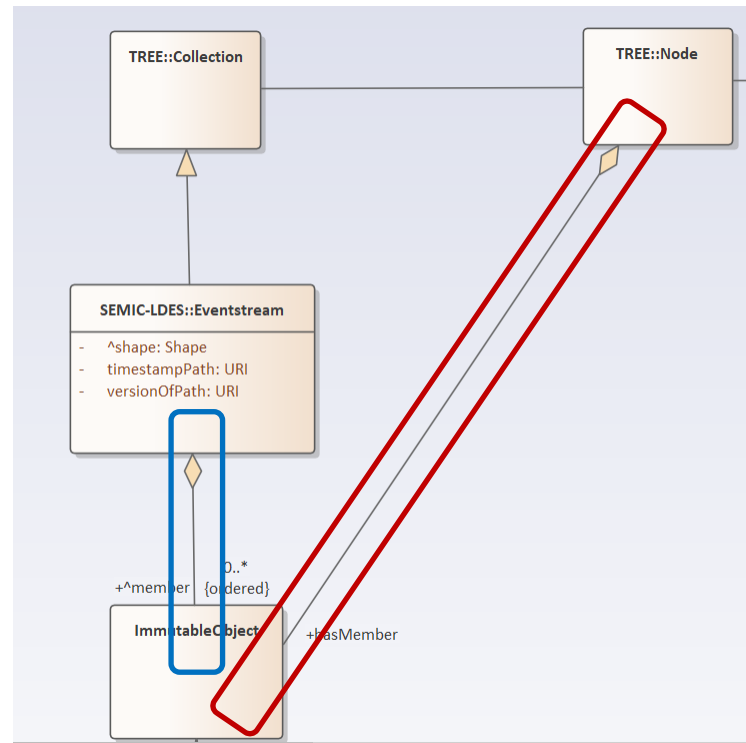
Pagina: elementen

GET <https://ex.org/view1>

```
<https://ex.org/MyEventStream> a ldes:EventStream ;  
  tree:view <https://ex.org/view1> ;  
  tree:member <m1> .
```

```
<https://ex.org/view1> tree:relation [  
  A tree:SubstringRelation ;  
  tree:value "m" ;  
  tree:path foaf:name ;  
  tree:node <https://ex.org/view2>  
] ;
```

redundant want je zit op deze pagina
ex:hasMember <m1> .



Pagina: elementen

Vragen:

Invoeren van een eigenschap hasMember voor een pagina?

Met gebruiksnota: *Deze eigenschap is ingevoerd om de definitie van de pagina te ondersteunen. In een LDES implementatie is het niet verplicht om deze relatie op te nemen de API respons. Dit omdat in een API respons van een pagina steeds de relatie met de eventstream bevat.*

Verschillende index-organisaties

Hoe drukken we uit of de index is georganiseerd volgens

- *Pagina's van 3 elementen volgens de registratietijd*
- *Pagina's van elementen volgens de eerste letter van de naam*
- ...

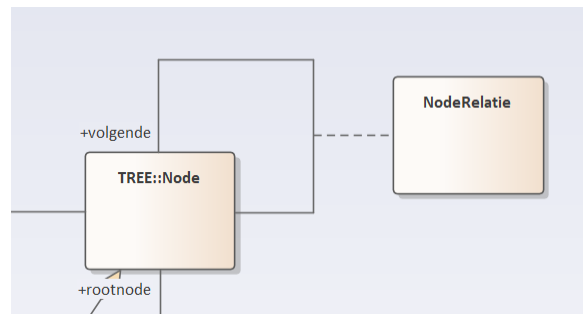
De *volgende* pagina

View2 bevat members met "m" in de naam, uitgedrukt volgens TREE:

```
<https://ex.org/view1> tree:relation [  
    a tree:SubstringRelation ;  
    tree:value "m" ;  
    tree:path foaf:name ;  
    tree:node <https://ex.org/view2>  
] ;
```

De *volgende* pagina

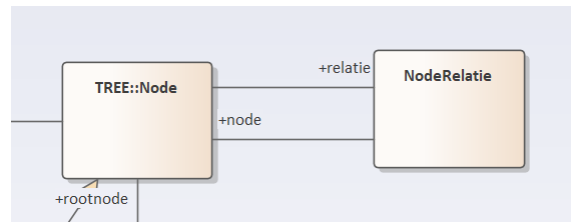
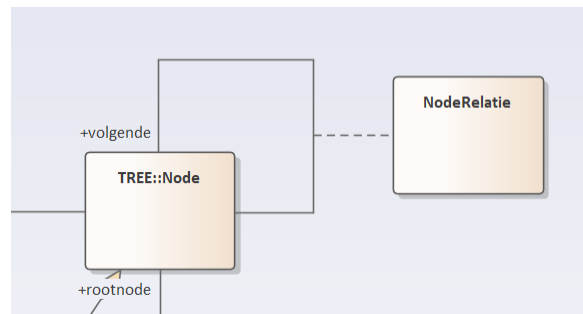
Een *pagina* (Node) is verbonden met 1 of meerdere *volgende* Node(s) volgende de eigenschap beschreven in *NodeRelatie*.



De *volgende* pagina

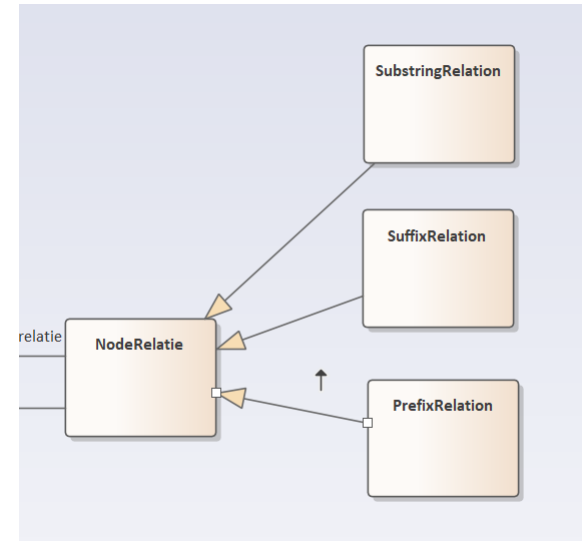
Een *pagina* (Node) is verbonden met 1 of meerdere *volgende* Node(s) volgende de eigenschap beschreven in *NodeRelatie*.

Vanuit implementatie perspectief is de kennis van de NodeRelatie onontbeerlijk. Daarom heeft TREE de associatieklasse expliciet gemaakt.

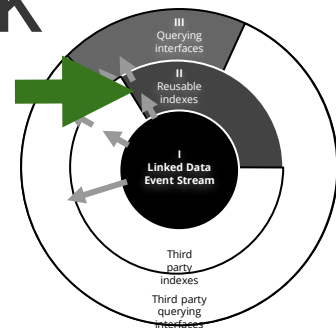


Bestaande types in TREE

- PrefixRelation
- SubstringRelation
- SuffixRelation
- GreaterThanRelation
- GreaterThanOrEqualRelation
- LessThanRelation
- LessThanOrEqualRelation
- EqualToRelation
- GeospatiallyContainsRelation



Types van NodeRelaties: aanpak



- Opnemen zoals een codelijst in OSLO
 - Voorzien een publicatie methode
 - Voorzien een discussie forum voor uitbreidingen
- Onderdeel van vervolg traject
- In scope hier: afspraken rond de methode en aanpak.

Afgeleide datasets

Afgeleide datasets (1/2)

Collecties die afgeleid zijn van een LDES:

Snapshots

= Een dump van een LDES tot een bepaald moment en eventueel verwijderen van members die niet meer geldig zijn

Versie-materialisaties

= een bepaalde versie waar geen versie-identificatoren meer in te vinden zijn

Tijdreeksen en aggregaties

= de members van de afgeleide LDES worden compacte stukjes time-windows met een evolutie van een bepaalde property

Combinatie

= Een merge-join van verschillende LDES-en in 1 LDES

Afgeleide datasets (2/2)

Schema mappings en alignering

= Data-model vertalen naar ander model

Anonimisering

= aggregeren + k-anonymity en/of differential privacy

Reconciliation

= Extra links toevoegen naar andere bronnen op basis van beschrijving

Brainstorm zo dadelijk

- Welke andere types van afgeleide datasets kennen jullie? Niet-deterministisch ook?
- Zou deze lijst exhaustief kunnen zijn?

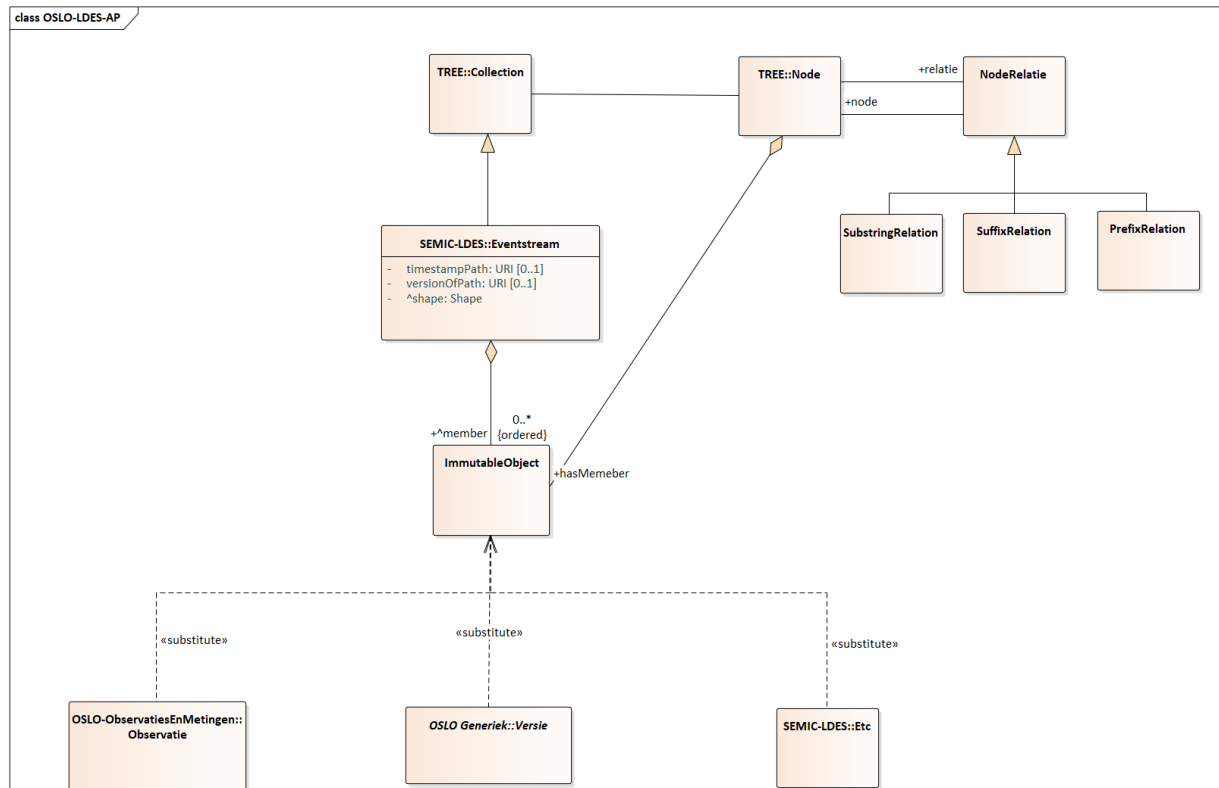
Hoe we deze exact beschrijven volgt in een latere workshop

Brainstorm oefening

Datamodel TW2

Drie vragen:

1. Algemene feedback op het datamodel
2. Andere soorten retentiebeleiden?
3. Welke types van afgeleide datasets kent u?



Brainstorm oefeningen



Link in de chat:

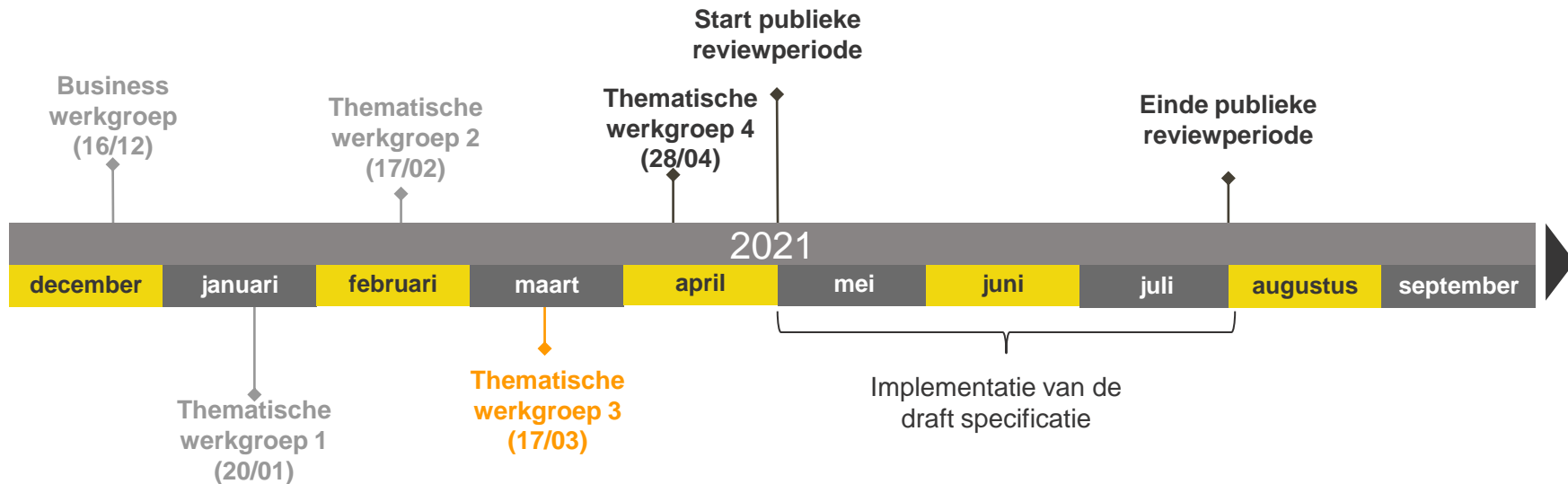
<https://app.mural.co/t/beadvtc7549/m/beadvtc7549/1644223871728/ce828415d3ea921bbfd1f745cab81928966ce9ee?sender=u048a1117151baed084666519>

Volgende stappen

Volgende stappen

Thematische werkgroep op **17 maart** (9h00-12h00)

Vergeet niet in te schrijven! <https://overheid.vlaanderen.be/informatie-vlaanderen/agenda/thematische-werkgroep-3-oslo-ldes>



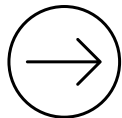
Volgende stappen – in de tussentijd...



Capteren van feedback en herzien van het model



Indienen van feedback/vragen via mail of GitHub.



Publiceren van het applicatieprofiel op de testomgeving



Vlaanderen
verbeelding werkt

Feedback & samenwerking



Feedback kan per e-mail worden
gegeven aan de volgende personen:

- oslo@vlaanderen.be



Feedback/input kan gegeven
worden via GitHub:

<https://github.com/Informatievlaanderen/OSLOthema-Ides/issues>

Via het aanmaken van **issues**

Vragen?



Vlaanderen
verbeelding werkt

Bedankt!