

Specificatie attributen gegenereerd voor associaties

Versie	Datum	Auteur	Aanpassingen
1.0	6 sep. 19	Martin Vanbrabant	-
1.1	12 sep. 19	Martin Vanbrabant	Extra gevallen Gestructureerde indeling
1.2.alfa1	27 sep. 19	Martin Vanbrabant	Aanpassingen na meeting 2019-09-25: Herformulering inleiding Extra hoofdstuk Algemene principes Extra specificatie: name(att) Extra specificatie: ap-label-nl(att) Extra specificatie: definition-nl(att); bij attributen in associatieklassen Extra specificatie: uri(att); gevallen met uri op betrokken associatie/rol/klasse Aanpassingen voor cross-package gevallen Aanpassingen voor afdelingen i.c.m. associatieklassen Bevat nog TODO regels – vandaar alfa1

Inleiding

Dit document specificeert te genereren attributen in de output van de OSLO toolchain, in de gevallen waar in de UML associaties voorkomen, al dan niet met bijhorende associatieklassen.

De verschillende gevallen worden onderscheiden en per geval wordt gedefinieerd welke attributen worden gegenereerd en welk eigenschappen deze attributen krijgen.

Om het proza te beperken, worden de eigenschappen in de vorm van 'formules' gespecificeerd. Deze formules zijn een soort pseudo code, waarvan de syntax of notaties hieronder wordt uitgelegd.

Bij dit document hoort een input UML specificatie, in de vorm van enkele Enterprise Architect .EAP bestanden, die als testinput gebruikt worden. De diagrammen en de voorbeelden die onder elke formule worden gegeven zijn ontleend aan deze .EAP bestanden.

Syntax in de formules

+	string concatenatie operator
UMLname (elem)	naam gegeven in UML aan elem. Niet te verwarren met de name tag!
effectiveName (elem)	waarde van de name tag van elem of indien deze niet bestaat, UMLname (elem).
baseURI (elem)	de base URI toegekend aan het package van elem.
xyz (elem)	waarde van tag xyz voor element aangeduid met elem.
tolower1 (tekst)	tekst, maar met eerste karakter omgezet naar kleine letter.
disamb (a, b)	indien nodig omdat de associatie een self-associatie is, voeg ter disambiguatie suffix a toe aan source kant, suffix b aan target kant.

In bovenstaande is elem een van:

- assoc: associatie
- att: attribuut (gegenereerd attribuut in kwestie)
- att_source: attribuut in source klasse of attribuut in associatieklasse, betreffende source klasse
- att_target: attribuut in target klasse of attribuut in associatieklasse, betreffende target klasse
- class: klasse (deelnemende klasse die het betreffende attribuut bevat)
- class_assoc: associatieklasse (die het betreffende attribuut bevat)
- class_source: klasse, aan source kant van associatie
- class_target: klasse, aan target kant van associatie
- rol: rol geplaatst op uiteinde van associatie

Een voorbeeld:

```
name(att) = tolower1(effectiveName(assoc))
```

"De name tag van het attribuut is gelijk aan de name tag van de associatie of als deze niet bestaat, de UML naam van de associatie, waarbij de eerste letter van deze name tag of UML naam werd omgezet naar kleine letter."

Over de voorbeelden

De voorbeelddiagrammen tonen van elk aanwezig element slechts de UML naam ervan (zie ook de functie UMLname (elem) hierboven). Om eenvoudig voorbeelden te kunnen maken van andere eigenschappen van een aanwezig element, werden volgende afspraken gehanteerd. Deze zijn ook geïmplementeerd in de voorbeeld .EAP bestanden.

- De waarde van de altijd aanwezige tag label-nl is aangemaakt op basis van de UML naam, maar bevat ter illustratie minstens één spatie. De eerste letter heeft hetzelfde voorkomen (kleine letter of grote letter) als de UML naam. Ter verduidelijking staan enkelvoudige aanhalingstekens voor en na de waarde.

- De waarde van de optionele tag `ap-label-nl` is gelijk aan de waarde van de tag `label-nl`, gevolgd door ' AP'.
- De waarde van de optionele tag `name` is gelijk aan de UML naam gevolgd door ' _naam '.

Bijvoorbeeld

UML naam	Heeft01
label-nl	'Heeft 01'
ap-label-nl	'Heeft 01 AP'
name	Heeft01_naam

Algemene principes

URI conflicten

In sommige gevallen kan voor meerdere gegenereerde attributen dezelfde URI worden gegenereerd. Een voorbeeld is geval 2, indien de associatie een tag `uri` heeft.

Dit kan aanleiding geven tot een URI conflict. Een URI conflict is het meer dan éénmaal een URI gebruiken, echter met verschillend domein en/of bereik.

In de gevallen waar de toolchain meer dan één keer eenzelfde URI genereert, zal ze een warning geven. Waarom geen error? Omdat dit soms ook niet tot een URI conflict leidt. Dit leidt bijvoorbeeld niet tot een URI conflict in een applicatieprofiel, als het om een afgedaalde associatie zou gaan, die in het vocabularium zou gedefinieerd zijn op een gemeenschappelijke basisklasse van de deelnemende klassen. In geval dit tot een daadwerkelijke URI conflict leidt, kan de gebruiker als alternatief een associatie met rollen gebruiken.

Niet gedefinieerde URI's in een applicatieprofiel

Indien men in een applicatieprofiel een associatie verlegt ("afdaalt") van een basisklasse naar een afgeleide, kan voor het betreffende attribuut in de deelnemende klasse een URI worden gegenereerd, die niet gedefinieerd werd in het vocabularium. Om dit te vermijden kan de gebruiker als alternatief een associatie met rollen gebruiken. Ook voor het linkend attribuut in de eventuele associatieklasse kan dit voorkomen. Om dit te vermijden kan de gebruiker de speciaal daarvoor voorziene tags `name-source-class` en `name-target-class` toekennen aan de associatieklasse. Deze worden beschreven verder in dit document.

Kopiëren van tags

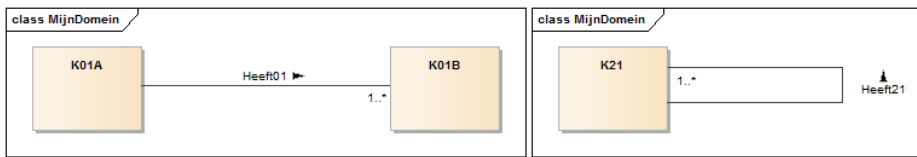
Attributen gegenereerd in deelnemende klassen krijgen naast de in dit document expliciet gedefinieerde tags ook een kopie van elke tag die toegekend was aan de associatie of de rol waarvoor ze werden gegenereerd, behalve de tags die expliciet worden vermeld in de formules.

Attributen gegenereerd in associatieklassen krijgen enkel de in dit document expliciet gedefinieerde tags.

Geval 1: zonder associatieklasse, met richtingspijl, zonder rollen

*Dit is de historische situatie die verder ondersteund wordt omwille van achterwaartse compatibiliteit.
Voor nieuwe modellen: voorkeur geven aan oplossingen zonder richtingspijl.*

UML voorbeeld



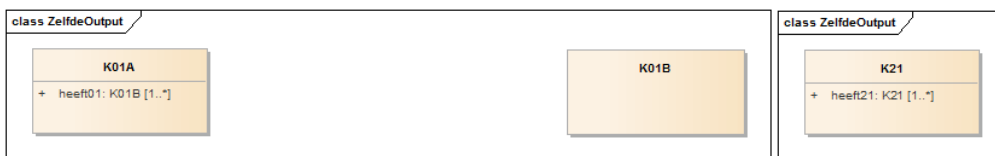
Kenmerken

- Naam van de associatie: aanwezig
- Richtingspijl bij de naam van de associatie: aanwezig
- Rollen: afwezig

Principes voor gegenereerde attributen

- Er wordt een attribuut gegenereerd in de klasse aan de kant van de associatie waaruit de richtingspijl vertrekt.

UML met zelfde output als het voorbeeld



Attributen in klassen

Formule 1.1.1

```
label-nl( att ) = tolower1( label-nl( assoc ) )
```

Voorbeelden

TODO - .EAP aanpassen voor label-nl in lijn met deze voorbeelden (alle formules)
'heeft 01',
'heeft 21'

Formule 1.1.2

Enkel als assoc **een tag** ap-label-nl heeft.

```
ap-label-nl( att ) = tolower1( ap-label-nl( assoc ) )
```

Voorbeelden

TODO - .EAP aanpassen voor ap-label-nl in lijn met deze voorbeelden (alle formules)
'heeft 01 AP',
'heeft 21 AP'

Formule 1.1.3

```
name( att ) = tolower1( effectiveName( assoc ) )
```

Voorbeelden

heeft01, heeft01_naam,
heeft21, heeft21_naam

Formule 1.1.4

Enkel als `assoc` ***een tag*** `uri` ***heeft.***

`uri(att) = uri(assoc)`

Voorbeelden

TODO - voorbeelden

Formule 1.1.5

Enkel als `assoc` ***geen tag*** `uri` ***heeft.***

`uri(att) = baseURI(assoc) + name(att)`

Gewijzigd t.o.v. versie 1.1: `baseURI(assoc)` in plaats van `baseURI(class)`

Voorbeelden

TODO - voorbeelden met `baseURI(assoc)` verschillend van `baseURI(class)`

`https://data.vlaanderen.be/ns/mijndomein#heeft01`

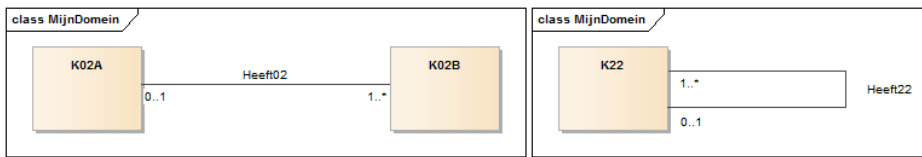
`https://data.vlaanderen.be/ns/mijndomein#heeft01_naam`

`https://data.vlaanderen.be/ns/mijndomein#heeft21`

`https://data.vlaanderen.be/ns/mijndomein#heeft21_naam`

Geval 2: zonder associatieklasse, zonder richtingspijl, zonder rollen

UML voorbeeld



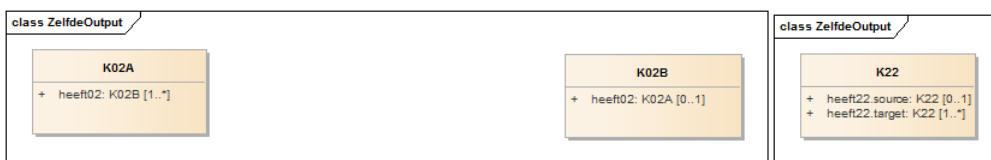
Kenmerken

- Naam van de associatie: aanwezig
- Richtingspijl bij de naam van de associatie: afwezig
- Rollen: afwezig

Principes voor gegenereerde attributen

- Er wordt een attribuut gegenereerd in beide betrokken klassen.
- Waar nodig, worden aan label-nl en URI een suffix toegevoegd ter disambiguatie.

UML met zelfde output als het voorbeeld



Attributen in klassen

Formule 2.1.1

```
label-nl(att) = tolower1(label-nl(assoc)) + disamb(' (source)', ' (target)')
```

Voorbeelden

```
'heeft 02',  
'heeft 22 (source)', 'heeft 22 (target)'
```

Formule 2.1.2

Enkel als assoc *een tag* ap-label-nl *heeft*.

```
label-nl(att) = tolower1(ap-label-nl(assoc)) + disamb(' (source)', ' (target)')
```

Voorbeelden

```
'heeft 02 AP',  
'heeft 22 AP (source)', 'heeft 22 AP (target)'
```

Formule 2.1.3

```
name(att) = effectiveName(class) + '.' + tolower1(effectiveName(assoc)) +  
disamb('.source', '.target')
```

Voorbeelden

```
K02A.heeft02, K02A_naam.heeft02, K02A.heeft02_naam,  
K02B.heeft02, K02B_naam.heeft02, K02B.heeft02_naam,  
K22.heeft22.source, K22_naam.heeft22.source, K22.heeft22_naam.source,  
K22.heeft22.target, K22_naam.heeft22.target, K22.heeft22_naam.target
```

Formule 2.1.4

Enkel als assoc *een tag* uri *heeft*.

```
uri(att) = uri(assoc)
```

Voorbeelden

Formule 2.1.5

Enkel als assoc geen tag uri heeft.

<code>uri(att) = baseURI(assoc) + name(att)</code>
--

Gewijzigd t.o.v. versie 1.1: baseURI(assoc) in plaats van baseURI(class)

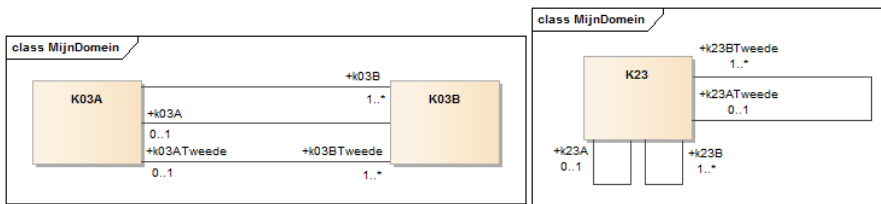
Voorbeelden

TODO - voorbeelden met baseURI(assoc) verschillend van baseURI(class)

```
https://data.vlaanderen.be/ns/mijndomein#K02A.heeft02
https://data.vlaanderen.be/ns/mijndomein#K02A_naam.heeft02
https://data.vlaanderen.be/ns/mijndomein#K02A.heeft02_naam
https://data.vlaanderen.be/ns/mijndomein#K02B.heeft02
https://data.vlaanderen.be/ns/mijndomein#K02B_naam.heeft02
https://data.vlaanderen.be/ns/mijndomein#K02B.heeft02_naam
https://data.vlaanderen.be/ns/mijndomein#K22.heeft22.source
https://data.vlaanderen.be/ns/mijndomein#K22_naam.heeft22.source
https://data.vlaanderen.be/ns/mijndomein#K22.heeft22_naam.source
https://data.vlaanderen.be/ns/mijndomein#K22.heeft22.target
https://data.vlaanderen.be/ns/mijndomein#K22_naam.heeft22.target
https://data.vlaanderen.be/ns/mijndomein#K22.heeft22_naam.target
```


Geval 3: zonder associatieklasse, zonder richtingspijl, met rollen

UML voorbeeld



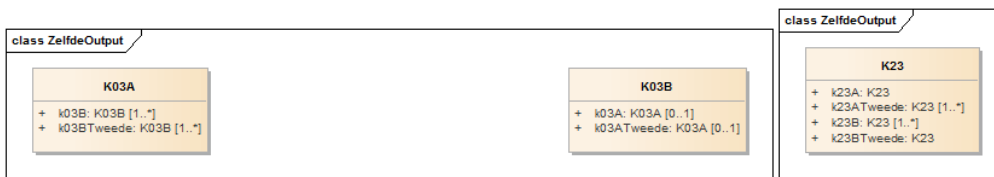
Kenmerken

- Naam van de associatie: afwezig
- Richtingspijl bij de naam van de associatie: afwezig
- Rollen: aanwezig

Principes voor gegenereerde attributen

- Enkel als aan een uiteinde van een associatie een rol is toegekend, wordt in de klasse aan het andere uiteinde van de associatie een attribuut gegenereerd.

UML met zelfde output als het voorbeeld



Attributen in klassen

Formule 3.1.1

`label-nl(att) = label-nl(rol)`

Voorbeelden

TODO - .EAP aanpassen zodat 'rol' voorkomt in `label-nl(rol)`

'k 03A', 'k 03A Tweede',
'k 03B', 'k 03B Tweede',
'k 23A', 'k 23A Tweede',
'k 23B', 'k 23B Tweede'

Formule 3.1.2

Enkel als rol een tag `ap-label-nl heeft`.

`ap-label-nl(att) = ap-label-nl(rol)`

Voorbeelden

TODO - .EAP aanpassen zodat 'rol' voorkomt in `ap-label-nl(rol)`

'k 03A AP', 'k 03A Tweede AP',
'k 03B AP', 'k 03B Tweede AP',
'k 23A AP', 'k 23A Tweede AP',
'k 23B AP', 'k 23B Tweede AP'

Formule 3.1.3

`name(att) = effectiveName(rol)`

Voorbeelden

TODO - .EAP aanpassen zodat 'rol' voorkomt in `effectiveName(rol)`

k03A, k03A_naam, k03ATweede, k03ATweede_naam,
k03B, k03B_naam, k03BTweede, k03BTweede_naam,

k23A, k23A_naam, k23ATweede, k23ATweede_naam,
k23B, k23B_naam, k23BTweede, k23BTweede_naam

Formule 3.1.4

Enkel als rol een tag uri heeft.

$\text{uri(att)} = \text{uri(rol)}$

Voorbeelden

TODO - voorbeelden

Formule 3.1.5

Enkel als rol geen tag uri heeft.

$\text{uri(att)} = \text{baseURI(rol)} + \text{name(att)}$
--

Gewijzigd t.o.v. versie 1.1: baseURI(rol) in plaats van baseURI(class)

Voorbeelden

TODO - voorbeelden met baseURI(rol) verschillend van baseURI(class)

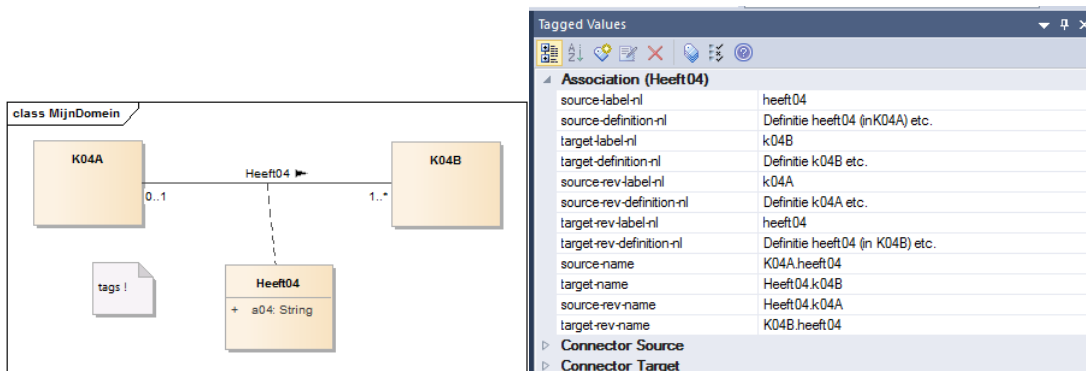
TODO - voorbeelden aanpassen zodra 'rol' voorkomt in name(att)

https://data.vlaanderen.be/ns/mijndomein#k03A
https://data.vlaanderen.be/ns/mijndomein#k03A_naam
https://data.vlaanderen.be/ns/mijndomein#k03ATweede
https://data.vlaanderen.be/ns/mijndomein#k03ATweede_naam
https://data.vlaanderen.be/ns/mijndomein#k03B
https://data.vlaanderen.be/ns/mijndomein#k03B_naam
https://data.vlaanderen.be/ns/mijndomein#k03BTweede
https://data.vlaanderen.be/ns/mijndomein#k03BTweede_naam
https://data.vlaanderen.be/ns/mijndomein#k23A
https://data.vlaanderen.be/ns/mijndomein#k23A_naam
https://data.vlaanderen.be/ns/mijndomein#k23ATweede
https://data.vlaanderen.be/ns/mijndomein#k23ATweede_naam
https://data.vlaanderen.be/ns/mijndomein#k23B
https://data.vlaanderen.be/ns/mijndomein#k23B_naam
https://data.vlaanderen.be/ns/mijndomein#k23BTweede
https://data.vlaanderen.be/ns/mijndomein#k23BTweede_naam

Geval 4: met associatieklasse, specifieke tags op associatieklasse

*Dit is de historische situatie die verder ondersteund wordt omwille van achterwaartse compatibiliteit.
Voor nieuwe modellen: voorkeur geven aan oplossingen zonder specifieke tags en zonder richtingspijl.*

UML voorbeeld



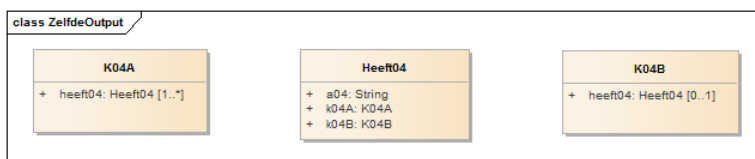
Kenmerken

- Specifieke tags op associatieklasse: aanwezig
- Naam van de associatie: aanwezig
- Richtingspijl bij de naam van de associatie: aanwezig
- Rollen: afwezig

Principes voor gegenereerde attributen

- Het genereren van attributen in bevattende klassen en associatieklassen wordt gestuurd door de tags **source-...**, **source-rev-...**, **target-...** en **target-rev-...** in de in de associatie.
- Het genereren van een attribuut kan worden onderdrukt door een respectievelijke tag **source-ignore**, **source-rev-ignore**, **target-ignore** of **target-rev-ignore** et waarde **true** toe te voegen.
- **Bemerk het ontbreken van een attribuut in de ene betrokken klasse dat verwijst naar de andere betrokken klasse: er wordt bijvoorbeeld in klasse K04A geen attribuut k04B gegenereerd.**

UML met zelfde output als het voorbeeld



Attributen in klassen (behalve associatieklasse)

Voorbeelden

heeft04

<https://data.vlaanderen.be/ns/mijndomein#K04A.heeft04>

<https://data.vlaanderen.be/ns/mijndomein#K04B.heeft04>

Attributen in associatieklasse

Voorbeelden

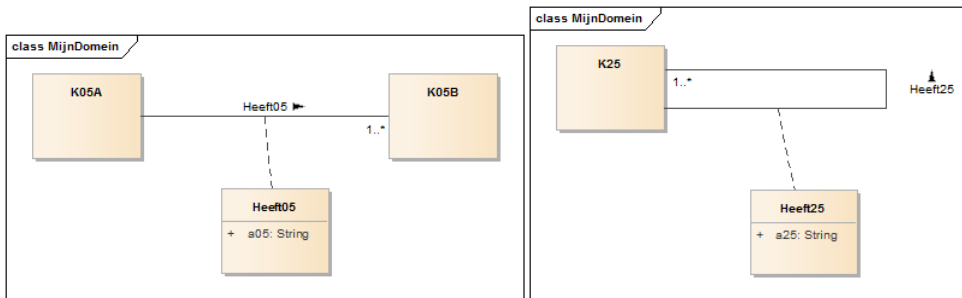
k04A, k04B

<https://data.vlaanderen.be/ns/mijndomein#Heeft04.k04A>
<https://data.vlaanderen.be/ns/mijndomein#Heeft04.k04B>

Geval 5: met associatieklasse, met richtingspijl, zonder rollen

Voor nieuwe modellen: voorkeur geven aan oplossingen zonder richtingspijl.

UML voorbeeld



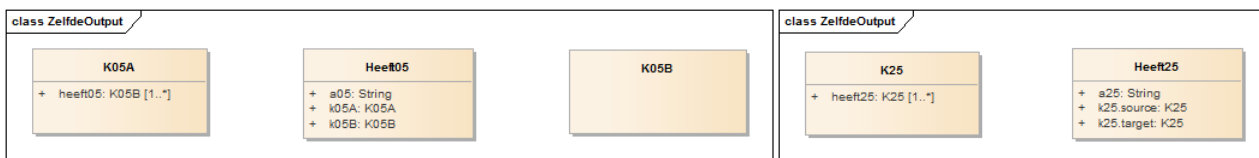
Kenmerken

- Specifieke tags op associatieklasse: afwezig
- Naam van de associatie: aanwezig
- Richtingspijl bij de naam van de associatie: aanwezig
- Rollen: afwezig

Principes voor gegenereerde attributen

- Er wordt een attribuut gegenereerd in de klasse aan de kant van de associatie waaruit de richtingspijl vertrekt.
- Er worden twee attributen gegenereerd in de associatieklasse, elk verwijzend naar een van de betrokken klassen.
- Waar nodig, worden aan label-nl en URI een suffix toegevoegd ter disambiguatie.

UML met zelfde output als het voorbeeld



Attributen in klassen (behalve associatieklasse)

Formules 5.1.1-5.1.5 zijn dezelfde als 1.1.1-1.1.5 (de voorbeelden verschillen).

Formule 5.1.1

```
label-nl(att) = tolower1(label-nl(assoc))
```

Voorbeelden

```
'heeft 05',
'heeft 25'
```

Formule 5.1.2

Enkel als *assoc een tag* `ap-label-nl heeft`.

```
ap-label-nl(att) = tolower1(ap-label-nl(assoc))
```

Voorbeelden

```
'heeft 05 AP',
'heeft 25 AP'
```

Formule 5.1.3

```
name(att) = tolower1(effectiveName(assoc))
```

Voorbeelden

```
heeft05, heeft05_naam,  
heeft25, heeft25_naam
```

Formule 5.1.4

Enkel als *assoc een tag uri heeft.*

```
uri(att) = uri(assoc)
```

Voorbeelden

TODO - voorbeelden

Formule 5.1.5

```
uri(att) = baseURI(assoc) + name(att)
```

Gewijzigd t.o.v. versie 1.1: baseURI(assoc) in plaats van baseURI(class)

Voorbeelden

TODO - voorbeelden met baseURI(assoc) verschillend van baseURI(class)
https://data.vlaanderen.be/ns/mijndomein#heeft05
https://data.vlaanderen.be/ns/mijndomein#heeft05_naam
https://data.vlaanderen.be/ns/mijndomein#heeft25
https://data.vlaanderen.be/ns/mijndomein#heeft25_naam

Attributen in associatieklasse

Formule 5.2.1.a

```
label-nl(att_source) = tolower1(label-nl(class_source)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 05A',  
'k 25 (source)'
```

Formule 5.2.1.b

```
label-nl(att_target) = tolower1(label-nl(class_target)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 05B',  
'k 25 (target)'
```

Formule 5.2.2.a

Enkel als *class_source een tag ap-label-nl heeft.*

```
ap-label-nl(att_source) = tolower1(ap-label-nl(class_source)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 05A AP',  
'k 25 AP (source)'
```

Formule 5.2.2.b

Enkel als *class_target een tag ap-label-nl heeft.*

```
ap-label-nl(att_target) = tolower1(ap-label-nl(class_target)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 05B AP',  
'k 25 AP (target)'
```

Formule 5.2.3.a

Enkel als *class_assoc een tag name-source-class heeft.*

```
name(att_source) = name-source-class(class_assoc) + disamb('.source', '.target')
```

Voorbeelden

TODO - voorbeelden

Formule 5.2.3.b

Enkel als class_assoc **een tag** name-target-class **heeft.**

```
name(att_target) = name-target-class(class_assoc) + disamb('.source', '.target')
```

Voorbeelden

TODO - voorbeelden

Formule 5.2.4.a

Enkel als class_assoc **geen tag** name-source-class **heeft.**

```
name(att_source) = tolower1(effectiveName(class_source)) + disamb('.source', '.target')
```

Voorbeelden

k05A, k05A_naam

k25.source, k25_naam.source

Formule 5.2.4.b

Enkel als class_assoc **geen tag** name-target-class **heeft.**

```
name(att_target) = tolower1(effectiveName(class_target)) + disamb('.source', '.target')
```

Voorbeelden

k05B, k05B_naam

k25.target, k25_naam.target

Formule 5.2.5.a

Enkel als class_assoc **een tag** uri **heeft.**

```
uri(att_source) = uri(class_assoc) + '.' + name(att_source)
```

Voorbeelden

TODO - voorbeelden

Formule 5.2.5.b

Enkel als class_assoc **een tag** uri **heeft.**

```
uri(att_target) = uri(class_assoc) + '.' + name(att_target)
```

Voorbeelden

TODO - voorbeelden

Formule 5.2.6.a

Enkel als class_assoc **geen tag** uri **heeft.**

```
uri(att_source) = baseURI(class_assoc) + effectiveName(class_assoc) +  
'.' + name(att_source)
```

Voorbeelden

https://data.vlaanderen.be/ns/mijndomein#Heeft05.k05A

https://data.vlaanderen.be/ns/mijndomein#Heeft05_naam.k05A_naam

https://data.vlaanderen.be/ns/mijndomein#Heeft25.k25.source

https://data.vlaanderen.be/ns/mijndomein#Heeft25_naam.k25_naam.source

Formule 5.2.6.b

Enkel als class_assoc **geen tag** uri **heeft.**

```
uri(att_target) = baseURI(class_assoc) + effectiveName(class_assoc) +  
'.' + name(att_target)
```

Voorbeelden

https://data.vlaanderen.be/ns/mijndomein#Heeft05.k05B

https://data.vlaanderen.be/ns/mijndomein#Heeft05_naam.k05B_naam

https://data.vlaanderen.be/ns/mijndomein#Heeft25.k25.target

https://data.vlaanderen.be/ns/mijndomein#Heeft25_naam.k25_naam.target

Formule 5.2.7.a

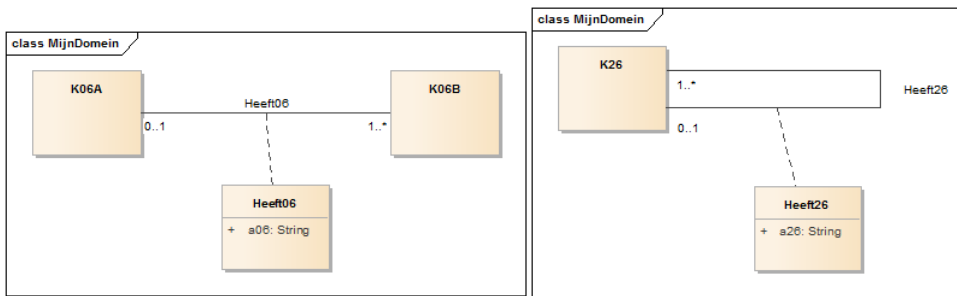
```
definition-nl(att_source) = 'Referentie naar verbonden klasse.'
```

Formule 5.2.7.b

<code>definition-nl(att_target) = 'Referentie naar verbonden klasse.'</code>
--

Geval 6: met associatieklasse, zonder richtingspijl, zonder rollen

UML voorbeeld



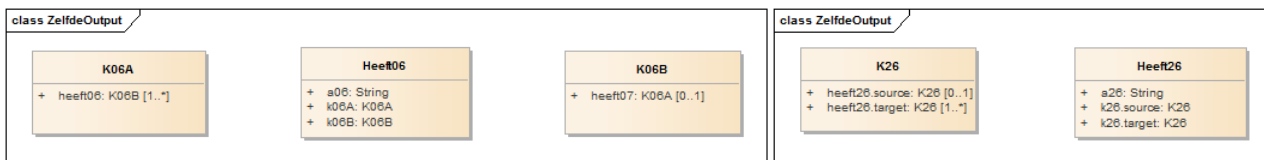
Kenmerken

- Specifieke tags op associatieklasse: afwezig
- Naam van de associatie: aanwezig
- Richtingspijl bij de naam van de associatie: afwezig
- Rollen: afwezig

Principes voor gegenereerde attributen

- Er wordt een attribuut gegenereerd in beide betrokken klassen, telkens verwijzend naar de andere betrokken klasse.
- Er worden twee attributen gegenereerd in de associatieklasse, elk verwijzend naar een van de betrokken klassen.
- Waar nodig, wordt aan eigenschappen een suffix toegevoegd ter disambiguatie.

UML met zelfde output als het voorbeeld



Attributen in klassen (behalve associatieklasse)

Formules 6.1.1-6.1.5 zijn dezelfde als 2.1.1-2.1.5 (de voorbeelden verschillen).

Formule 6.1.1

```
label-nl(att) = tolower1(label-nl(assoc)) + disamb(' (source)', ' (target)')
```

Voorbeelden

```
'heeft 06',  
'heeft 26 (source)', 'heeft 26 (target)'
```

Formule 6.1.2

Enkel als assoc een tag ap-label-nl heeft.

```
label-nl(att) = tolower1(ap-label-nl(assoc)) + disamb(' (source)', ' (target)')
```

Voorbeelden

```
'heeft 06 AP',  
'heeft 26 AP (source)', 'heeft 26 AP (target)'
```

Formule 6.1.3

```
name(att) = effectiveName(class) + '.' + tolower1(effectiveName(assoc)) +  
disamb('.source', '.target')
```

Voorbeelden

```
K06A.heeft06, K06A_naam.heeft06, K06A.heeft06_naam,  
K06B.heeft06, K06B_naam.heeft06, K06B.heeft06_naam,  
K26.heeft26.source, K26_naam.heeft26.source, K26.heeft26_naam.source,  
K26.heeft26.target, K26_naam.heeft26.target, K26.heeft26_naam.target
```

Formule 6.1.4

Enkel als assoc *een tag* uri heeft.

```
uri(att) = uri(assoc)
```

Voorbeelden

TODO - voorbeelden

Formule 6.1.5

Enkel als assoc *geen tag* uri heeft.

```
uri(att) = baseURI(assoc) + name(att)
```

Gewijzigd t.o.v. versie 1.1: baseURI(assoc) in plaats van baseURI(class)

Voorbeelden

TODO - voorbeelden met baseURI(assoc) verschillend van baseURI(class)

```
https://data.vlaanderen.be/ns/mijndomein#K06A.heeft06  
https://data.vlaanderen.be/ns/mijndomein#K06A_naam.heeft06  
https://data.vlaanderen.be/ns/mijndomein#K06A.heeft06_naam  
https://data.vlaanderen.be/ns/mijndomein#K06B.heeft06  
https://data.vlaanderen.be/ns/mijndomein#K06B_naam.heeft06  
https://data.vlaanderen.be/ns/mijndomein#K06B.heeft06_naam  
https://data.vlaanderen.be/ns/mijndomein#K26.heeft26.source  
https://data.vlaanderen.be/ns/mijndomein#K26_naam.heeft26.source  
https://data.vlaanderen.be/ns/mijndomein#K26.heeft26_naam.source  
https://data.vlaanderen.be/ns/mijndomein#K26.heeft26.target  
https://data.vlaanderen.be/ns/mijndomein#K26_naam.heeft26.target  
https://data.vlaanderen.be/ns/mijndomein#K26.heeft26_naam.target
```

Attributen in associatieklasse

Formules 6.2.1-6.2.7 zijn dezelfde als 5.2.1-5.2.7 (de voorbeelden verschillen).

Formule 6.2.1.a

```
label-nl(att_source) = tolower1(label-nl(class_source)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 06A',  
'k 26 (source)'
```

Formule 6.2.1.b

```
label-nl(att_target) = tolower1(label-nl(class_target)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 06B',  
'k 26 (target)'
```

Formule 6.2.2.a

Enkel als class_source *een tag* ap-label-nl heeft.

```
ap-label-nl(att_source) = tolower1(ap-label-nl(class_source)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 06A AP',  
'k 26 AP (source)'
```

Formule 6.2.2.b

Enkel als `class_target` **een tag** `ap-label-nl` **heeft.**

```
ap-label-nl(att_target) = tolower1(ap-label-nl(class_target)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

```
'k 06B AP',  
'k 26 AP (target)'
```

Formule 6.2.3.a

Enkel als `class_assoc` **een tag** `name-source-class` **heeft.**

```
name(att_source) = name-source-class(class_assoc) + disamb('.source', '.target')
```

Voorbeelden

TODO - voorbeelden

Formule 6.2.3.b

Enkel als `class_assoc` **een tag** `name-target-class` **heeft.**

```
name(att_target) = name-target-class(class_assoc) + disamb('.source', '.target')
```

Voorbeelden

TODO - voorbeelden

Formule 6.2.4.a

Enkel als `class_assoc` **geen tag** `name-source-class` **heeft.**

```
name(att_source) = tolower1(effectiveName(class_source)) + disamb('.source', '.target')
```

Voorbeelden

```
k06A, k06A_naam  
k26.source, k26_naam.source
```

Formule 6.2.4.b

Enkel als `class_assoc` **geen tag** `name-target-class` **heeft.**

```
name(att_target) = tolower1(effectiveName(class_target)) + disamb('.source', '.target')
```

Voorbeelden

```
k06B, k06B_naam  
k26.target, k26_naam.target
```

Formule 6.2.5.a

Enkel als `class_assoc` **een tag** `uri` **heeft.**

```
uri(att_source) = uri(class_assoc) + '.' + name(att_source)
```

Voorbeelden

TODO - voorbeelden

Formule 6.2.5.b

Enkel als `class_assoc` **een tag** `uri` **heeft.**

```
uri(att_target) = uri(class_assoc) + '.' + name(att_target)
```

Voorbeelden

TODO - voorbeelden

Formule 6.2.6.a

Enkel als `class_assoc` **geen tag** `uri` **heeft.**

```
uri(att_source) = baseURI(class_assoc) + effectiveName(class_assoc) +  
'.' + name(att_source)
```

Voorbeelden

```
https://data.vlaanderen.be/ns/mijndomein#Heeft06.k06A  
https://data.vlaanderen.be/ns/mijndomein#Heeft06_naam.k06A_naam  
https://data.vlaanderen.be/ns/mijndomein#Heeft26.k26.source  
https://data.vlaanderen.be/ns/mijndomein#Heeft26_naam.k26_naam.source
```

Formule 6.2.6.b

Enkel als class_assoc *geen tag* uri *heeft*.

```
uri(att_target) = baseURI(class_assoc) + effectiveName(class_assoc) +  
'.' + name(att_target)
```

Voorbeelden

```
https://data.vlaanderen.be/ns/mijndomein#Heeft06.k06B  
https://data.vlaanderen.be/ns/mijndomein#Heeft06_naam.k06B_naam  
https://data.vlaanderen.be/ns/mijndomein#Heeft26.k26.target  
https://data.vlaanderen.be/ns/mijndomein#Heeft26_naam.k26_naam.target
```

Formule 6.2.7.a

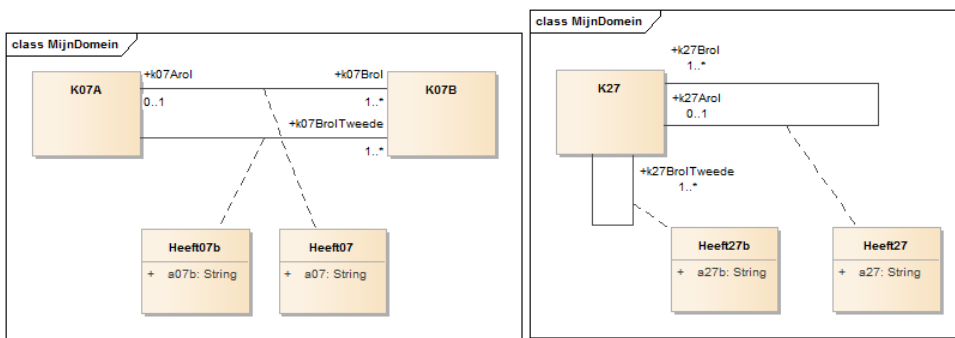
```
definition-nl(att_source) = 'Referentie naar verbonden klasse.'
```

Formule 6.2.7.b

```
definition-nl(att_target) = 'Referentie naar verbonden klasse.'
```

Geval 7: met associatieklasse, zonder richtingspijl, met rollen

UML voorbeeld



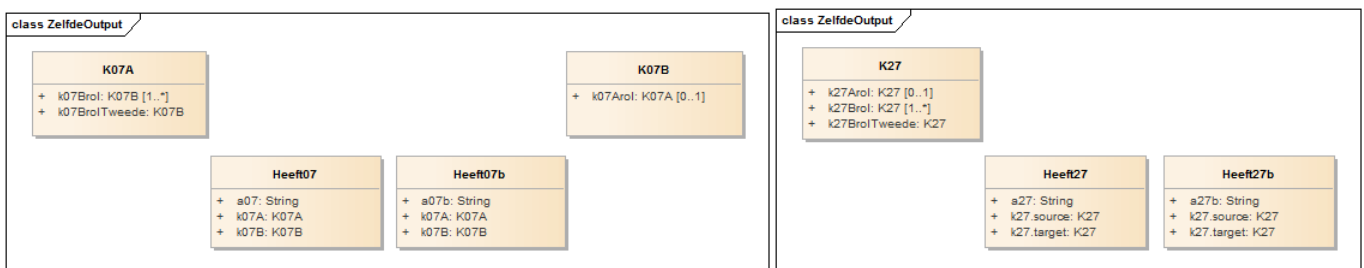
Kenmerken

- Specifieke tags op associatieklasse: afwezig
- Naam van de associatie: afwezig
- Richtingspijl bij de naam van de associatie: afwezig
- Rollen: aanwezig

Principes voor gegenereerde attributen

- Enkel als aan een uiteinde van een associatie een rol is toegekend, wordt in de klasse aan het andere uiteinde van de associatie een attribuut gegenereerd.
- Er worden twee attributen gegenereerd in de associatieklasse, elk verwijzend naar een van de betrokken klassen.
- Waar nodig, wordt aan eigenschappen een suffix toegevoegd ter disambiguatie.

UML met zelfde output als het voorbeeld



Attributen in klassen (behalve associatieklasse)

Formules 7.1.1-7.1.5 zijn dezelfde als 3.1.1-3.1.5 (de voorbeelden verschillen).

Formule 7.1.1

```
label-nl(att) = label-nl(rol)
```

Voorbeelden

```
'k 07A rol', 'k 07B rol',
'k 07B rol Tweede',
'k 27A rol', 'k 27B rol',
'k 27B rol Tweede'
```

Formule 7.1.2

Enkel als rol een tag ap-label-nl **heeft.**

```
ap-label-nl(att) = ap-label-nl(rol)
```

Voorbeelden

'k 07A rol AP', 'k 07B rol AP', 'k 07B rol Tweede AP',
'k 27A rol AP', 'k 27B rol AP', 'k 27B rol Tweede AP'

Formule 7.1.3

```
name(att) = effectiveName(rol)
```

Voorbeelden

k07Arol, k07Arol_naam, k07Brol, k07Brol_naam, k07BrolTweede, k07BrolTweede_naam,
k27Arol, k27Arol_naam, k27Brol, k27Brol_naam, k27BrolTweede, k27BrolTweede_naam

Formule 7.1.4

Enkel als rol een tag uri heeft.

```
uri(att) = uri(rol)
```

Voorbeelden

TODO - voorbeelden

Formule 7.1.5

Enkel als rol geen tag uri heeft.

```
uri(att) = baseURI(rol) + name(att)
```

Gewijzigd t.o.v. versie 1.1: baseURI(rol) in plaats van baseURI(class)

Voorbeelden

TODO - voorbeelden met baseURI(rol) verschillend van baseURI(class)
https://data.vlaanderen.be/ns/mijndomein#k07Arol
https://data.vlaanderen.be/ns/mijndomein#k07Arol_naam
https://data.vlaanderen.be/ns/mijndomein#k07Brol
https://data.vlaanderen.be/ns/mijndomein#k07Brol_naam
https://data.vlaanderen.be/ns/mijndomein#k07BrolTweede
https://data.vlaanderen.be/ns/mijndomein#k07BrolTweede_naam
https://data.vlaanderen.be/ns/mijndomein#k27Arol
https://data.vlaanderen.be/ns/mijndomein#k27Arol_naam
https://data.vlaanderen.be/ns/mijndomein#k27Brol
https://data.vlaanderen.be/ns/mijndomein#k27Brol_naam
https://data.vlaanderen.be/ns/mijndomein#k27BrolTweede
https://data.vlaanderen.be/ns/mijndomein#k27BrolTweede_naam

Attributen in associatieklasse

Formules 7.2.1-7.2.7 zijn dezelfde als 5.2.1-5.2.7 (de voorbeelden verschillen).

Formule 7.2.1.a

```
label-nl(att_source) = tolower1(label-nl(class_source)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

'k 07A',
'k 27 (source)'

Formule 7.2.1.b

```
label-nl(att_target) = tolower1(label-nl(class_target)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

'k 07B',
'k 27 (target)'

Formule 7.2.2.a

Enkel als class_source een tag ap-label-nl heeft.

```
ap-label-nl(att_source) = tolower1(ap-label-nl(class_source)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

'k 07A AP',

'k 27 AP (source)'

Formule 7.2.2.b

Enkel als `class_target` **een tag** `ap-label-nl` **heeft.**

```
ap-label-nl(att_target) = tolower1(ap-label-nl(class_target)) +  
disamb(' (source)', ' (target)')
```

Voorbeelden

'k 07B AP',
'k 27 AP (target)'

Formule 7.2.3.a

Enkel als `class_assoc` **een tag** `name-source-class` **heeft.**

```
name(att_source) = name-source-class(class_assoc) + disamb('.source', '.target')
```

Voorbeelden

TODO - voorbeelden

Formule 7.2.3.b

Enkel als `class_assoc` **een tag** `name-target-class` **heeft.**

```
name(att_target) = name-target-class(class_assoc) + disamb('.source', '.target')
```

Voorbeelden

TODO - voorbeelden

Formule 7.2.4.a

Enkel als `class_assoc` **geen tag** `name-source-class` **heeft.**

```
name(att_source) = tolower1(effectiveName(class_source)) + disamb('.source', '.target')
```

Voorbeelden

k07A, k07A_naam
k27.source, k27_naam.source

Formule 7.2.4.b

Enkel als `class_assoc` **geen tag** `name-target-class` **heeft.**

```
name(att_target) = tolower1(effectiveName(class_target)) + disamb('.source', '.target')
```

Voorbeelden

k07B, k07B_naam
k27.target, k27_naam.target

Formule 7.2.5.a

Enkel als `class_assoc` **een tag** `uri` **heeft.**

```
uri(att_source) = uri(class_assoc) + '.' + name(att_source)
```

Voorbeelden

TODO - voorbeelden

Formule 7.2.5.b

Enkel als `class_assoc` **een tag** `uri` **heeft.**

```
uri(att_target) = uri(class_assoc) + '.' + name(att_target)
```

Voorbeelden

TODO - voorbeelden

Formule 7.2.6.a

Enkel als `class_assoc` **geen tag** `uri` **heeft.**

```
uri(att_source) = baseURI(class_assoc) + effectiveName(class_assoc) +  
'.' + name(att_source)
```

Voorbeelden

https://data.vlaanderen.be/ns/mijndomein#Heeft07.k07A
https://data.vlaanderen.be/ns/mijndomein#Heeft07_naam.k07A_naam
https://data.vlaanderen.be/ns/mijndomein#Heeft27.k27.source

`https://data.vlaanderen.be/ns/mijndomein#Heeft27_naam.k27_naam.source`

Formule 7.2.6.b

Enkel als class_assoc geen tag uri heeft.

```
uri(att_target) = baseURI(class_assoc) + effectiveName(class_assoc) +  
'.' + name(att_target)
```

Voorbeelden

```
https://data.vlaanderen.be/ns/mijndomein#Heeft07.k07B  
https://data.vlaanderen.be/ns/mijndomein#Heeft07_naam.k07B_naam  
https://data.vlaanderen.be/ns/mijndomein#Heeft27.k27.target  
https://data.vlaanderen.be/ns/mijndomein#Heeft27_naam.k27_naam.target
```

Formule 5.2.7.a

```
definition-nl(att_source) = 'Referentie naar verbonden klasse.'
```

Formule 5.2.7.b

```
definition-nl(att_target) = 'Referentie naar verbonden klasse.'
```