**MEMO**

|  |  |
| --- | --- |
| **Onderwerp** | Metamodel MIM/KKG – verzoek: zelf gedefinieerd datatype baseren op ander datatype |
| **Aan** | Paul Janssen, Ellen Debats, Arjan Kloosterboer, Arjan Loeffen |
| **Van** | Lennart van Bergen, Linda van den Brink, Peter Lentjes |
| **Datum** | 2018-03-21 |
| **Status** | Voorstel – stap 5: uitwerking door actiehouder. |

**Inleiding**

#29. <https://github.com/Geonovum/KKG-Metamodel/issues/29>

Op verzoek van Kadaster/IMKAD, Geonovum/BRO: hoe geef je aan dat een zelf gedefinieerd datatype, gebaseerd is op ander datatype?

Bijvoorbeeld:

* primitief datatype AN is gebaseerd op CharacterString (primitief datatype, extern aan MIM)
* NietNegatiefGetal is gebaseerd op Integer (primitief datatype, extern aan MIM)
* Vlak is gebaseerd op GM\_Surface (niet primitief datatype, extern aan MIM)

Bijna altijd is het onderliggende datatype een CharacterString, en dit wordt nu behandeld als de default, maar dit is niet altijd zo. Daarnaast is het netjes om het expliciet aan te geven.

In MIM staat nu:

*Het is ook mogelijk om in het eigen informatiemodel een eigen primitive datatype te definiëren in de vorm van een «Primitief datatype», met als UML metaclass de UML-Primitive datatype.*

***Voorbeelden*** *hiervan, die niet tot KKG behoren, maar ter illustratie zijn opgenomen, zijn:*

* *NietNegatieveInteger: een Integer die alleen de waarde 0 of groter mag hebben. Laat de naam van het primitieve type dan wel terugkomen in de naam (dus niet NietNegatiefGetal).*
* *Een beperking op een Real te specificeren door Decimal op te nemen (een gebroken getal, met (één of meer) cijfers voor de komma en cijfers achter de komma, conform ISO11404).*
* *AN. Deze is gebruikelijk bij een aantal basisregistraties. Datatype met een eigen naam, analoog aan CharacterString, maar met alleen ‘normale’ tekens. Dit zijn alle alfanumerieke tekens (dus inclusief diakrieten), de koppeltekens – en \_ en spaties. De minimale lengte is tenminste 1, de maximale lengte is onbepaald. De 1e positie mag géén spatie bevatten*

Deze uitwerking gaat primair over het specificeren van een zelf-gedefinieerd datatype die gebaseerd is op een ander datatype, die geen MIM stereotype heeft. Maar het is in principe ook toegestaan om structuur over te erven, zoals een specifieke <<referentielijst>>, die de structuur overerft van een generieke <<referentielijst>>, of bij een Bedrag en een Positief Bedrag. Het kan zijn dat dit soort modellering liever niet gezien wordt, en daarom ook (nog) niet wordt opgenomen in het MIM, of met een bepaalde regel of richtlijn.

**De insteek bij de uitwerking die gevolgd is …**

Een zelf-gedefinieerd datatype maken doe je alleen als je een eigen datatype een verbijzondering/ aanscherping is van een ander datatype, waarbij dit andere datatype extern is aan je eigen informatiemodel.

We modelleren we dit als volgt:

* UML-Datatype, met een eigen naam (dit staat al zo in MIM).
* Deze heeft een UML-Generalisatie relatie van het zelf gedefinieerde type naar het datatype waar het een verbijzondering van is.

Merk op dat de definitie van de datatypes in dit geval dus gelijk kan blijven (niet een doel op zichzelf).

**Uitwerking: Generalisatie van datatype naar datatype.**

Voor «Primitief datatype», «Gestructureerd datatype», «Referentielijst» .

Niet voor «Codelist», «Enumeratie ».

Definitie

**MIM Hoofdstuk 2**

**Paragraaf 2.2**

**Generalisatie:** De UML-representatie van een specialisatie, uitgedrukt in een UML-generalization (metaclass).

Een generalisatie kan worden gelegd tussen een objecttypes (bestaande tekst):

***Definitie Generalisatie***

*De typering van het hiërarchische verband tussen een meer generiek object van een objecttype en een meer specifiek object van een ander objecttype waarbij het laatstgenoemde object eigenschappen van het eerstgenoemde object overerft.*

Een generalisatie kan worden gelegd tussen datatypes (nieuw tekst):

**Definitie Generalisatie**

De typering van het hiërarchische verband tussen een meer generieke structuur van data in de vorm van een datatype, en een meer specifieke structuur van data in de vorm van een ander datatype, waarbij het laatstgenoemde datatype de eigenschappen van het eerstgenoemde datatype overerft, én een verbijzondering hierin aanbrengt in de vorm van een meer restrictieve definitie, of een meer restrictief patroon/formeel patroon.

Toelichting:

* het andere datatype is bijvoorbeeld een CharacterString, Integer, GM Surface of DMO en dient als basis voor een zelf te definiëren datatype (zie 3.1.2.).
* deze generalisatie is van toepassing op de volgende datatypes: «Primitief datatype», «Gestructureerd datatype», «Referentielijst».

Keuze maken: afsplitsen nieuw stereotype <<generalisatie datatype>> of het doen met <<generalisatie>>. Als afsplitsen, dan ook splitsen voor Union (?).

**Paragraaf 2.1**

Generalisatie toevoegen aan 2.1.2.

**Paragraaf 2.3**

**Specificatie voor «Generalisatie»**

De generalisaties worden naar het volgende aspect gespecificeerd:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Aspect** | **Kardi**  **naliteit** | **Toelichting** | **In UML 2.5** | **In EA** |
| **Naam** | 0..1 | De naam van de generalisatie.  Standaard ‘is specialisatie van’. | *name van de metaclass Named element* | *Name* |
| **Datatype** | 1 | Het datatype dat een specialisatie is van een (ander) datatype. | */source: related*  *Element bij Relationship* *Element* | *Source* |
| **Gerelateerd datatype** | 1 | Het datatype dat de generalisatie is van een (ander) datatype. | */target: related*  *Element bij Relationship* *Element* | *Target* |

**H3**

**3.1.2. Primitief datatype zelf definiëren** 🡪 herformuleren naar **3.1.2. Datatype zelf definiëren**

Het is ook mogelijk om in het eigen informatiemodel een eigen datatype te definiëren in de vorm van een «Primitief datatype», «Gestructureerd datatype», of «Referentielijst». Zelf gedefinieerde datatypes hebben altijd een eigen definitie en optioneel een eigen patroon of formeel patroon.

**V*oorbeelden*** hiervan, die niet tot KKG behoren, maar ter illustratie zijn opgenomen, zijn:

* NietNegatieveInteger: een Integer die alleen de waarde 0 of groter mag hebben. Laat de naam van het primitieve type dan wel terugkomen in de naam (dus niet NietNegatiefGetal).
* Een beperking op een Real te specificeren door Decimal op te nemen (een gebroken getal, met (één of meer) cijfers voor de komma en cijfers achter de komma, conform ISO11404).
* AN. Deze is gebruikelijk bij een aantal basisregistraties. Datatype met een eigen naam, analoog aan CharacterString, maar met alleen ‘normale’ tekens. Dit zijn alle alfanumerieke tekens (dus inclusief diakrieten), de koppeltekens – en \_ en spaties. De minimale lengte is tenminste 1, de maximale lengte is onbepaald. De 1e positie mag géén spatie bevatten.
* Een Vlak: een verbijzondering van een GM Surface, met een eigen definitie.



De gele datatypes zijn extern aan het model.

Het type modelelement (stereotype) veranderd niet door de generalisatie. Een zelf gedefinieerd primitief datatype zal een generalisatie hebben met een ander primitief datatype. Een zelf gedefinieerd gestructureerd datatype zal een generalisatie hebben met een ander gestructureerd datatype.

Het komt voor dat het zelf gedefinieerde datatype een generalisatie heeft naar een extern gedefinieerd datatype, waarvan het modelelement (stereotype) niet is gespecificeerd. Maak dan zelf een inschatting.

Let hierbij op bij een «Gestructureerd datatype». Deze hoort altijd twee of meer data elementen te hebben. Een GM Surface heeft zelf geen data elementen, waardoor een Vlak niet als een «Gestructureerd datatype» mag worden bestempeld. GM Surface wordt in deze gezien als een «Primitief datatype»

**MIM Hoofdstuk 3**

**Bijlage: standaard werkwijze afhandelen punten**

1. bepalen wel of niet in scope bepalen   
   Ja.
2. bepalen of het hoort bij conceptueel en/of logisch, of in een extensie  
   Conceptueel en logisch, MIM zelf.
3. Richting/alternatieven bespreken  
   Eerste stap gezet, zie hierboven. Inventariseren ideeën.
4. Actiehouder bepalen  
   Lennart en Linda.
5. Voorstel uitwerken door actiehouder  
   Lennart.
6. Ter review rondsturen door actiehouder en review verwerken.
7. Vaststellen door kernteam:   
     
   Als niet OK: n.v.t.
   1. discussie overleg inplannen
   2. beslissen, indien mogelijk
   3. bijzonder overleg, indien nodig
8. Voorstel delen met een aantal mensen uit de doelgroep in de nabije periferie.
9. Bij besluit: verwerken in standaard – document én profiel, met een achteraf controle of het nu goed is.
10. Vaststellen nieuwe versie van de standaard.