Handreiking

Geometrie in model en GML

Geonovum

Linda van den Brink
Paul Janssen
Wilko Quak

datum
1 maart 2012

versie
1.0

Inhoudsopgave

1 Inleiding 4
1.1 Inhoud van de handreiking 4
1.2 Leeswijzer 4
2 Geometrie in Model 5
2.1 Ruimtelijk schema van ISO 19107 5
2.1.1 GM_Object 6
2.2 Geometrische primitieven. 6
2.2.1 GM_Point 6
2.2.2 GM_Curve 7
2.2.3 GM_Surface 7
2.2.4 GM_Solid 7
2.2.5 Interpolatie 8
2.3 Geometrische aggregaties 8
2.3.1 GM_MultiPoint 8
2.3.2 GM_MultiCurve 8
2.3.3 GM_MultiSurface 9
2.3.4 GM_MultiSolid 9
2.4 Definities van 2, 2.5 en 3D 9
2.5 Nederlands profiel op ISO 19107 9
3 Geometrie in GML – Nederlands profiel 10
3.1 Totstandkoming SF2 als Nederlands profiel op GML 10
3.2 GML Simple Features profielen en data complexiteit 11
3.3 Regels voor alle Simple Features profile levels 11
3.3.1 Feature collecties 12
3.3.2 Geometrietypen 13
3.3.3 Coördinaatreferentiesysteem 14
3.4 Simple features profile 0 (SF0) 15
3.5 Simple features profile 1 (SF1) 16
3.5.1 Zelf ontwikkelde datatypen: geneste inhoud. 17
3.6 Simple features profile 2 (SF2) 17
3.7 Schema compliance: data levering – data ontvangen 17
4 GML SF2: Nederlands profiel op GML 3.2 19
4.1 Introductie Schematron 19
4.2 Schematron business rules voor Nederlands profiel 19
5 Basisregels voor transformatie van een SF2 naar een SF0 model. 22
5.1 Inleiding 22
5.2 Verschil SF0 en SF2 22
5.3 Normalisatieregels 22
5.3.1 Meervoudig attribuut 22
5.3.2 Zelfgemaakte datatypen 23
5.3.3 Associaties 23
Bijlage 1 Algemene coderingsregels GML 25
1.1 Van UML naar GML 25
1.2 GML encoding rules 25
Bijlage 2 Uitgewerkt voorbeeld 27
2.1 IMGolf model in UML 27
2.2 IMGolf Feature Catalogus 28
2.3 IMGolf XML applicatie schema 35
2.4 IMGolf GML document 39
Bijlage 3 GML 3.2.1 en GML 3.1.1 43
3.1 Inleiding 43
3.2 Waar iedereen mee te maken krijgt 43
3.3 Andere verschillen met mogelijke gevolgen 44

Hoofdstuk 1

Inleiding

Deze handreiking beschrijft de toepassing van geometrie in informatiemodellering en de implementatie daarvan in GML. Het is daarmee een ondersteuning voor de toepassing van de norm NEN 3610:2011 – Basismodel geo-informatie. NEN 3610 gaat hierin niet verder dan verwijzing naar de relevante geo-informatie (ISO) normen. Voor het werkveld zijn deze normen niet toegankelijk genoeg om te kunnen toepassen. Middels deze handreiking wordt de brug gelegd. Dit document kan als een zelfstandig document worden gelezen. De inhoud is grotendeels gebaseerd op originele normen. Voor de toegankelijkheid is de tekst vertaald naar het Nederlands en bovendien vereenvoudigd. Deze handreiking kan daarom niet de normen vervangen, maar bijvoorbeeld als introductie worden gebruikt. Voor de normatieve referentie wordt in alle gevallen naar de originele documenten verwezen.

1.1 Inhoud van de handreiking

In eerste instantie komt geometrie voor in informatiemodellen in de vorm van geometrietypen populair bekend als punten, lijnen, vlakken. ISO 19107 beschrijft middels een ruimtelijk schema (spatial schema) deze geometrische primitieven, de uitbreidingen daarop en de relaties daartussen. De toepassing in dat ruimtelijk schema is ook bepalend voor toepassing in informatiemodellen. De relatie tussen het ruimtelijk schema en de toepassing in informatiemodellen wordt beschreven.

Het ruimtelijk schema kent geometrietypen met een toenemende complexiteit. Niet alle typen zijn gegeven de stand van techniek even inter-operabel toepasbaar. Er is daarom een keuze gemaakt tot welke typen we ons in Nederland (in de regel) beperken. Deze inperking wordt beschreven in een profiel.

Uitwisseling van geo-informatie (bestanden) gebeurt middels GML. Dat betekent dat de conceptuele informatiemodellen toegepast worden in GML. Beschreven wordt hoe het ruimtelijke schema vertaald wordt naar de GML implementatie omgeving. Ook hier wordt er uitgegaan van een Nederlands profiel op de complete GML implementatie. Afgesproken is dat het Nederlands profiel het GML Simple Features profile level 2 volgt. Dit profiel wordt in de annex door middel van business rules in schematron vastgelegd. Met de schematron regels is een GML bestand te controleren op conformiteit met het Nederlandse profiel.

Voor de implementatie in services is het soms nodig om vanwege de interoperabiliteit de complexiteit van het GML bestand te reduceren tot Simple Features profile 0. Dat betekent dat het informatiemodel ook gesimplificeerd moet worden. In hoofdstuk 5 zijn een aantal basisregels opgenomen voor het transformeren van een SF2 model naar een SF0 model.

1.2 Leeswijzer

Hoofdstuk 2 van dit document bevat een uitleg van het ruimtelijk schema zoals beschreven in ISO 19107 en gehanteerd in de GML standaard.
Hoofdstuk 3 bevat een uitleg van het Simple Features profile for GML.
Hoofdstuk 4 beschrijft de Nederlandse toepassing van GML en het Simple Features profile en de regels die daaruit voortvloeien.
In Hoofdstuk 5 wordt tenslotte uitgelegd hoe, indien nodig, een aan Simple features level 2 voldoend informatiemodel kan worden omgezet naar een vereenvoudigd model dat voldoet aan level 0.
De belangrijkste algemene GML coderingsregels staan beschreven in Appendix 1, een uitgewerkt voorbeeld van de in dit document beschreven regels en werkwijze in appendix 2, en een overzicht van de verschillen tussen GML 3.1.1 en 3.2.1 in appendix 3.

Hoofdstuk 2

Geometrie in Model

Dit hoofdstuk beschrijft de definitie van de geometrietypen zoals die zijn gedefinieerd in ISO 19107 en in informatiemodellen worden toegepast. Alleen het geometrie gedeelte van het ruimtelijke schema is in deze handreiking opgenomen;het topologie gedeelte wordt niet besproken.

2.1 Ruimtelijk schema van ISO 19107

NEN 3610 beschrijft de regels voor het modelleren van geo-informatie. Belangrijk in de definitie van een geografisch object is dat het locatie eigenschappen heeft. Eén van die locatie eigenschappen is de directe locatie die middels coördinaten is beschreven. Door middel van attributen wordt de directe locatie en geometrie door coördinaten gerepresenteerd. De waarden van die attributen zijn coördinaten of coördinaatreeksen die een geometrie representeren. We noemen dit geometrietypen. In ISO 19107 is een klassediagram, het ruimtelijk schema, opgenomen met de geometrietypen en hun onderlinge relaties. De geometrietypen uit dit diagram worden gebruikt om de geometrietypen in geo-informatiemodellen te specificeren.

Voor de geometrietypen zijn er verschillende niveau´s van complexiteit van structuur en dimensionaliteit. Voor de complexiteit van structuur is er een verdeling in de geometrisch primitieven punt, lijn, vlak en volume en de geometrische complexen. Complexen hebben een complexe datastructuur en zijn opgebouwd uit een verzameling geometrische primitieven.

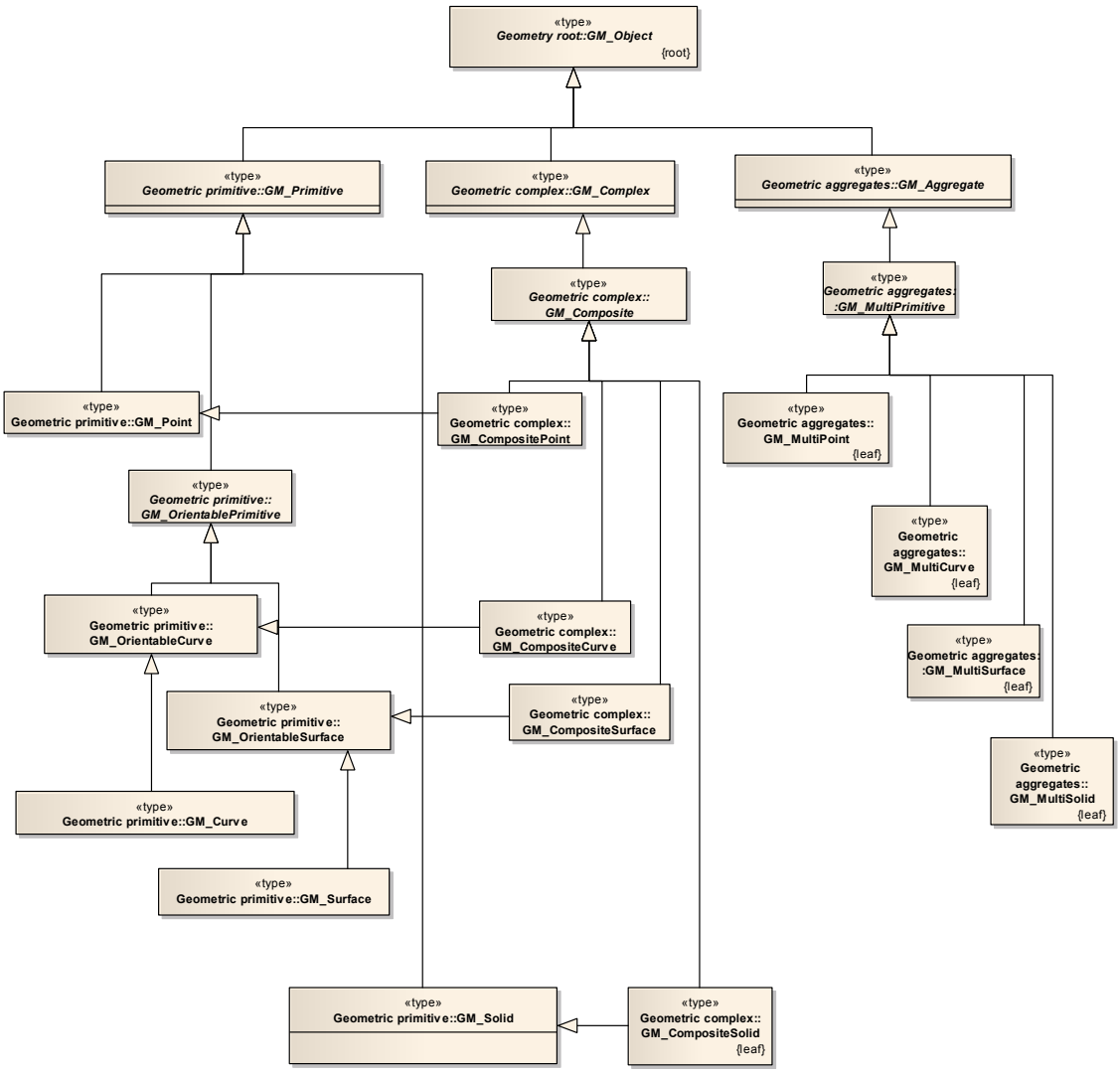
- Geometrische primitieven
- Geometrische complexen

Dimensionaliteit is het tweede criterium bij het definiëren van geometrietypen.

- 0 dimensies
- 1 dimensie
- 2 dimensies
- 3 dimensies

Het ISO 19107 ruimtelijk schema is geconstrueerd rond die twee criteria van data complexiteit en dimensionaliteit. Er is nog een derde criterium, functionaliteit (operaties), dat echter voor de indeling van de geometrietypen in het ruimtelijk schema geen gevolgen heeft. ISO 19125-1 Simple feature access definieert een model voor 2 dimensionale geometrietypen. Zie hiervoor ook het OGC origineel OpenGIS® Implementation Standard for Geographic information - Simple feature access - Part 1: Common architecture.In dit 2D geometriemodel zijn restricties opgenomen die van een algemene geometrie een simpele geometrie maken. De term simple feature staat daarin voor features beperkt tot 2 dimensionale geometrie.

Figuur 1 beeldt het ruimtelijk schema af en is overgenomen uit het INSPIRE consolidated UML model. Dat is weer gebaseerd op het model voor geometrische basistypen uit de ISO 19107.



Figuur 1 Het ruimtelijk schema

Uit dit schema worden alleen de geometrietypen toegelicht die voor deze handreiking van toepassing zijn. De toelichting is beperkt en omvat alleen de informatie die voor de toepassing van dit document relevant is. Voor de volledige beschrijving wordt verwezen naar ISO 19107. Voor simple features (2D) gelden er restricties die apart worden aangegeven.

2.1.1 GM_Object

De superklasse van alle geometrietypen. Deze klasse beschrijft de eigenschappen die alle geometrie types in ISO 19107 delen. Operaties die voor de klasse GM_Object beschreven zijn zullen dus voor ieder geometrietype geïmplementeerd zijn. Als in een model de geometrie van een object beschreven moet worden terwijl er niets bekend is over het mogelijke voorkomen van die geometrie dan kan GM_Object worden gekozen.

Ook als een klasse meerdere geometrietypen toestaat (bijvoorbeeld: een inrichtingselement kan een punt of een lijn zijn) kan GM_Object worden gebruikt, waarbij dit met een constraint wordt ingeperkt tot GM_Point en GM_Curve.

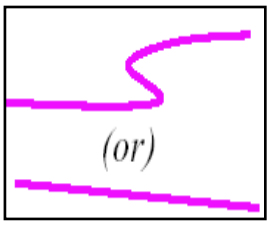
2.2 Geometrische primitieven.

2.2.1 GM_Point

Punt. 0-dimensionale geometrie.

2.2.2 GM_Curve

Lijn. 1-dimensionale geometrie.
Een curve is simple indien er geen zelfintersectie optreedt.

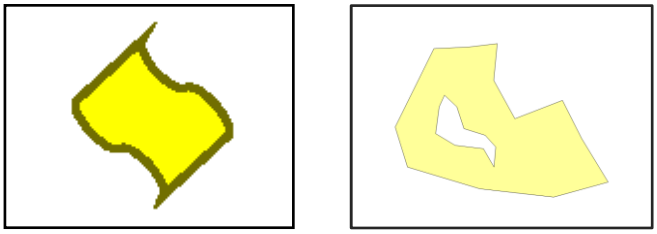


Figuur 2 Lijngeometrie

Lijnen zijn continu en hebben een meetbare lengte in een coördinaten systeem. Lijnen bestaan uit een of meer lijnsegmenten waarbij de lijnsegmenten verschillende interpolatiemethoden kunnen gebruiken. Lijnsegmenten zijn aan elkaar verbonden waarbij het eindpunt van elk segment, behalve de laatste, verbonden is aan het beginpunt van het volgende. Wanneer het begin- en eindpunt van een lijn met elkaar verbonden zijn is de lijn gesloten en spreken we ook wel van een Ring.

2.2.3 GM_Surface

Vlak. 2-dimensionale geometrie.



Figuur 3 Vlakgeometrie en vlakgeometrie met binnengrens

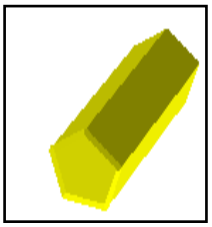
Een vlak heeft een oriëntatie, dat wil zeggen, een boven en onderzijde. De bovenzijde is gedefinieerd als de kant vanaf welke de begrenzing tegen de klok in (anti clock wise) is gedigitaliseerd. Een vlak bestaat uit een of meer 'surface patches'. Indien er meerdere zijn dan vormen die samen een aaneengesloten vlak.

Een vlak is simple indien een vlak bestaat uit één surface patch.

Een vlak heeft normaal gesproken in ieder geval in 2D situaties een buitenkant, de exterior. Daarnaast kan een vlak ook nog 0 of meer interne begrenzingen hebben, de interior. De buitenkant heeft, zoals hierboven beschreven, van bovenaf gezien een richting tegen de klok in. De binnenkant wordt met de klok mee vastgelegd.

2.2.4 GM_Solid

Volume. 3-dimensionaal geometrietype.



Figuur 4 Volume geometrie

De geometrie van een GM_Solid is opgebouwd uit GM_SolidBoundaries (grenzen) die elk weer een GM_Surface zijn. Deze zijn naar buiten toe georiënteerd: de bovenkant van elk vlak is aan de buitenkant van het volume object te zien.

2.2.5 Interpolatie

Lijnen zijn opgebouwd uit lijnsegmenten. Lijnsegmenten kunnen verschillende interpolatiemethoden voor de interpretatie van de coördinaten hebben. In de regel worden in Nederland (zie Nederlands profiel) alleen de lineaire en 'circular arc by three points' gebruikt.

De volgende interpolatiemethoden zijn gespecificeerd. In de lijst hieronder zijn de Engelse ISO 19107 termen weergegeven. Alleen de meest gebruikte typen worden uitgelegd.

- Linear: rechte lijn tussen twee opeenvolgende punten
- Geodesic
- Circular arc by three points: elke set van drie punten vormt een cirkelvormige boog (arc) die begint bij het eerste punt, door het tweede punt gaat en eindigt bij het derde punt. Als de drie punten op één lijn liggen ontstaat er een rechte lijn.
- Circular arc by two points
- Circular arc by two points and bulge factor
- Elliptical arc
- Clothoid
- Conic arc
- Polynominal spline
- Cubic spline
- Rational spline

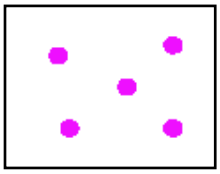
2.3 Geometrische aggregaties

Het is mogelijk om samenstellingen van geometrietypen te aggregeren in nieuwe geometrie typen. Hiervoor is het aggregatiepakket ontwikkeld. Aggregaties hebben verder geen regels over de inhoudelijke structuur; in algemeenheid zijn er geen regels over of de geometrische primitieven elkaar moeten/mogen raken, overlappen enzovoort. Voor simple features (2D) gelden er wel restricties. Deze worden apart aangegeven.

Geometrische complexen oftewel composites zijn ook samenstellingen van geometrische primitieven, maar hierbij gelden wel aanvullende regels. Deze composites worden verder niet besproken. Ze zijn geen onderdeel van het Simple Features profile (zie Hoofdstuk 3).

2.3.1 GM_MultiPoint

Multipunt. Verzameling van punten die gezamenlijk één object vormen. (instanties van GM_Point).
Een multipoint is simple indien er geen punten met dezelfde coördinaten voorkomen.



Figuur 5 Multipunt geometrie

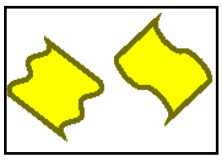
2.3.2 GM_MultiCurve

Multilijn. Verzameling van lijnen die gezamenlijk één object vormen (instanties van GM_Curve).
Een multicurve is simple indien de samenstellende delen simple zijn en de enige intersecties tussen delen zich bevinden in punten aan de buitenkant van die objecten (geen kruising wel vertakking).

Figuur 6 Multilijn geometrie

2.3.3 GM_MultiSurface

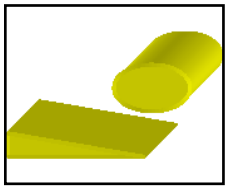
Multivlak. Verzameling van vlakken die gezamenlijk één object vormen (instanties van GM_Surface). Vlakken in een multisurface mogen elkaar niet overlappen. In een simple multisurface mogen vlakken elkaar alleen raken in een eindig aantal punten (wel punten maar geen grenzen gemeenschappelijk)



Figuur 7 Multivlak geometrie

2.3.4 GM_MultiSolid

Multivolume. Verzameling van volumes die gezamenlijk één object vormen (instanties van GM_Solid).



Figuur 8 Multivolume geometrie

2.4 Definities van 2, 2.5 en 3D

Voor het representeren van geometrieën is de dimensionaliteit van groot belang. Voor 2 en 3 dimensionale representatie lijkt voor de hand te liggen wat daar mee bedoeld wordt. In de praktijk wordt er ook nog over 2.5 D gesproken. Het is daarom goed om bruikbare definities te hanteren.

2D: Objecten bestaan uit punten, lijnen of vlakken. Coördinaten zijn een x en een y-waarde.

2,5D: Punten, lijnen of vlakken. Coördinaten zijn een x,y en z waarde. Bij elk x,y-locatie is er maximaal één z-waarde beschikbaar. Dat betekent dat 2D geometrien zich in een 3D ruimte (x,y,z) bevinden maar dat de geometrien zichzelf of elkaar niet mogen overlappen in de horizontale projectie.

3D: Volume eenheden oftewel 'solids' zijn onderdeel van de mogelijke geometrietypen. Bij elke x,y-locatie zijn er meerdere z-waarden mogelijk.

2.5 Nederlands profiel op ISO 19107

Het ruimtelijk schema in al zijn complexiteit is geschikt om zeer gevarieerde representaties van geometrieën te definiëren. Bij het kiezen van geometrietypen ter implementatie in een softwareomgeving is het van belang eisen ten aanzien van interoperabiliteit mee te nemen. In het kader van het Nederlands profiel op ISO 19107 is er gekeken naar GML als implementatieomgeving. Daarin is geëvalueerd welke subset van geometrietypen uit het ruimtelijke schema tot een basisset, een profiel, moeten behoren. Er is daarmee een directe link tussen het Nederlandse profiel op GML en het profiel op ISO 19107.

Hoofdstuk 3

Geometrie in GML – Nederlands profiel

Het Nederlands Basismodel Geo-informatie (NEN 3610) specificeert in overeenstemming met de ISO-geo-informatiestandaarden GML (Geography Markup Language) als uitwisselingsformaat voor geo-informatie. De actuele standaard voor GML wordt gehanteerd. Op dit moment is dat GML 3.2.1, waarbij ook GML 3.1.1 nog ondersteund wordt. Omdat GML 3.2.1 een zeer uitgebreide standaard is wordt er een profiel, een subset gehanteerd.

GML 3.2.1 is een uitgebreide standaard, ontwikkeld door het Open Geospatial Consortium (OGC) en biedt oplossingen voor een groot aantal situaties en variaties voor het uitwisselen van geo-informatie. Variaties zijn er in geometrietypen maar ook in complexiteit van datastructuren. Om verschillende niveaus van toepassing van GML 3.2.1 mogelijk te maken zijn er door OGC zogenaamde profielen gemaakt. De ontwikkeling van de profielen is voortgekomen uit een behoefte van onder meer softwareleveranciers om verschillende niveaus van complexiteit te ondersteunen. Operabiliteit kan daarmee beter gegarandeerd worden. De profielen omvatten elk een subset van de totale GML 3.2.1 standaard. De standaardprofielen zijn Simple Features Profile 0, Simple Features Profile 1 en Simple Features Profile 2 (afgekort tot respectievelijk SF0, SF1, en SF2). Deze profielen hebben een toenemende complexiteit en bieden ook een toenemende functionaliteit. SF0 is dus het eenvoudigst, en SF2 het meest uitgebreid. Het OGC Simple Features profile moet niet verward worden met de ISO 19125 - Simple feature access standaard. De eerste gaat over implementatie van geometrie in GML de tweede over definities van 2 dimensionale geometrietypen.

Een profiel dat meer complexiteit en functionaliteit toestaat, biedt meer mogelijkheden voor datamodellering of geometriebeschrijvingen, maar is ook moeilijker toe te passen in software. Deze complexiteit kan de uitwisseling tussen verschillende softwareplatforms bemoeilijken. Een eenvoudig profiel daarentegen biedt minder mogelijkheden voor datamodellering maar is wel eenvoudiger toe te passen in generieke software. Het is daarom van belang voor- en nadelen tegen elkaar af te wegen.

In Nederland is het SF2 profiel op GML van toepassing verklaard.

3.1 Totstandkoming SF2 als Nederlands profiel op GML

Bij het tot stand komen van het Nederlandse GML profiel zijn een aantal beginselen leidend geweest:

- het profiel moet de eisen van het geo-informatie werkveld weerspiegelen;
- het profiel is gekoppeld aan (toepassingen van) NEN 3610;
- het profiel is een basisprofiel dat basiseisen die door de sectorale informatiemodellen gesteld worden omvat;
- een sectoraal informatiemodel kan uitbreidingen op het basisprofiel formuleren;
- het profiel moet afgestemd zijn op de actuele technische mogelijkheden die software biedt.

Door deze beginselen geeft het profiel voldoende richting voor standaardisering van algemene functionaliteit maar ook flexibiliteit voor toepassingen die specifieke oplossingen nodig hebben. Het profiel geeft richting aan de aanbodkant van geo-informatie en aan de modelleurs van informatiemodellen voor de oplossingen en mogelijkheden voor datamodellering. Voor softwaretoepassingen biedt het profiel een basis waar functionaliteit en toepassingen op afgestemd kunnen worden.

Om een uitspraak te kunnen doen over de eisen die in Nederland gesteld worden aan GML als uitwisselingsstandaard is een enquête gehouden onder de beheerders van aan NEN 3610 gerelateerde informatiemodellen. De volgende informatiemodellen en hun respectievelijke sectoren zijn hier bij betrokken: BAG (adressen en gebouwen), IMGeo (geografie), IMWA, UM Aquo-m, UM Aquo-krw (water), TOP10NL/IMTOP (topografie), GBR (grootschalig weg, water en waterwegen), IMWE (welstand), IMRO

(ruimtelijke ordening), IMKICH (cultuurhistorie), IMKAD (kadaster), IMKL, BMKL (kabels en leidingen), IMBOD (bodem). Op basis van deze eisen is een eerste concept profiel voor GML ontwikkeld.

Het conceptprofiel is gepresenteerd aan en voor commentaar uitgebracht naar het hele geo-informatie werkveld. Een belangrijke rol was hierin voor softwareleveranciers en –ontwikkelaars om eisen vanuit de toepassingen in het profiel te verwerken. Het conceptprofiel is gepubliceerd onder de werknaam gml4nl en was op het moment van publicatie gebaseerd op GML 3.1.1.

Bij de ISO publicatie van GML 3.2.1 is ook het Nederlandse profiel vernieuwd. Het Nederlandse profiel bleek daarbij zo weinig van het GML SF2 profiel te verschillen dat besloten is om geen apart Nederlands profiel te publiceren maar SF2 als officieel Nederlandse profiel op GML 3.2.1 te volgen. Belangrijk was daarin dat het huidige SF2 profiel, in tegenstelling tot 3.1.1 SF2, ook cirkels en bogen opgenomen heeft als geometrievormen.

3.2 GML Simple Features profielen en data complexiteit

De GML standaard is een omvangrijke verzameling van XML elementen en attributen waarmee van alles mogelijk is. De GML standaard bevat bijvoorbeeld definities voor dynamische (voortdurend veranderende) objecten, topologie, complexe geometrische typen, en rasters. Dit maakt GML tot een rijke standaard die veel te bieden heeft, maar maakt het ook moeilijker voor software aanbieders om de hele standaard te ondersteunen.

Om dit probleem te adresseren zijn er verschillende profielen op GML gedefinieerd door de OGC. Een GML profiel is een subset van de complete GML set. De subset is wel zodanig dat het profiel niet in tegenspraak is met de complete set. Voor GML 3.2.1 zijn er drie profielen met toenemende complexiteit, de zogenaamde Simple Features profielen. Softwareleveranciers kunnen ervoor kiezen hun ondersteuning van GML te beperken tot een van deze profielen, of uiteraard om de hele standaard te ondersteunen.

Wat is nu een "simple feature"? Een feature is in de context van GML een equivalent van een instantie van een objecttype zoals bijvoorbeeld een gebouw, een boom, een persoon, een ding. Deze instanties hoeven niet noodzakelijkerwijs een geometrie, of locatie-informatie te hebben. Met 'simple' wordt in deze context bedoeld dat de features:

- Een eenvoudige datastructuur hebben.
- Als het ruimtelijke objecten zijn, een eenvoudige geometrische vorm hebben.

De GML standaard is gebaseerd op de algemene XML standaard XML Schema[1]. Een deel van de regels uit de simple features profielen hebben hier betrekking op. SF0 beperkt bijvoorbeeld de datatypen uit XML Schema die gebruikt mogen worden (string, boolean, etc) en definieert daarmee in feite ook een subset van XML Schema. Dit geldt voor de meeste regels die betrekking hebben op de eenvoud van de datastructuur.

Wat er precies wordt beschouwd als 'eenvoudig', oftewel wat er wel en niet is toegestaan, wordt gespecificeerd in de GML Simple Features Profile standaard[2].

3.3 Regels voor alle Simple Features profile levels

De volgende tabel uit het OGC document Geography Markup Language (GML) simple features profile, geeft een overzicht van de inhoud van de verschillende profielen.

---

[1] http://www.w3.org/standards/techs/xmlschema

[2] Geography Markup Language (GML) simple features profile (with technical note). OGC 10-100r3. Open Geospatial Consortium, version 2.0, 2011-05-11.

Tabel 1. Inhoud SF niveau's

	SF-0	**SF-1**	**SF-2**
Beperkte set van ingebouwde niet ruimtelijke typen	String, integer, measurement, date, real, binary, boolean, URI	String, integer, measurement, date, real, binary, boolean, URI	Geen beperking
Beperkte set van geometrietypen[3]	(multi)Punt, (multi)lijn, (multi)vlak,(multi) geometry	(multi)Punt, (multi)lijn, (multi)vlak,(multi) geometry	(multi)Punt, (multi)lijn, (multi)vlak,(multi) geometry
Zelf ontwikkelde datatypen	nee	ja	ja
Gebruik van nillable en xsi:nil	nee	ja	ja
Cardinaliteit(multipliciteit)	0 of 1	0 tot onbeperkt	0 tot onbeperkt
Niet-ruimtelijke referenties	Ja, Alleen by reference (gml:ReferenceType)	Ja By reference (gml:ReferenceType) of inline	Ja By reference en inline in combinatie is mogelijk.
Ruimtelijke referenties	Ja, Alleen by reference (gml:ReferenceType)	Ja Alleen by reference (gml:ReferenceType)	Ja By reference en inline in combinatie is mogelijk.

Het is toegestaan samengestelde (geaggregeerde) geometrie te gebruiken, te weten MultiPoint, MultiCurve, MultiSurface, en MultiGeometry. Hierbinnen geldt dat alleen de enkelvoudige container elementen mogen worden gebruikt. Dus wel: 'gml:pointMember' maar niet: 'gml:pointMember**s**' enzovoort.

Voor **alle niveau's (SF0, SF1 en SF2)** geldt bovendien dat de volgende GML onderdelen in het Simple Features profile **niet** worden ondersteund:

- Rasters (coverages)
- Topologie
- Tijd en dynamische features
- Observaties/metingen

Verder geldt:

- Het gebruik van Gml:metadataProperty is niet toegestaan. Men moet indien nodig zelf metadata elementen definiëren in een GML application schema.
- In het GML application schema moet het compliance level op de voorgeschreven manier (zie paragraaf 3.7) worden aangegeven.
- In het GML application schema moet het volledige GML schema geïmporteerd worden (gml.xsd).
- Een GML application schema mag maximaal één feature collection op de voorgeschreven manier (zie paragraaf 3.3.1) definiëren.

3.3.1 Feature collecties

Een GML document is gedefinieerd als één feature collectie, een verzameling van geo-objecten. Deze collectie krijgt een eigen naam gedefinieerd in het XML-Schema. Een GML document begint met de declaratie van deze collectie in het root element. Er wordt geen gebruik gemaakt van de container gml:featureMembers en de individuele gml:featureMember. Deze elementen zijn komen te vervallen vanaf GML versie 3.2.

[3]Tabel 2. op pagina 16 geeft meer detail over de toegestane geometrietypen.

Voorbeeld XSD fragment:

```
<element name="FeatureCollectionIMRO"
    type="imro:FeatureCollectionIMROType"
    substitutionGroup="gml:AbstractGML"/>
<complexType name="FeatureCollectionIMROType">
    <complexContent>
        <extension base="gml:AbstractFeatureType">
            <sequence minOccurs="0" maxOccurs="unbounded">
                <element name="featureMember">
                    <complexType>
                        <complexContent>
                            <extension base="gml:AbstractFeatureMemberType">
                                <sequence>
                                    <element ref="gml:AbstractFeature"/>
                                </sequence>
                            </extension>
                        </complexContent>
                    </complexType>
                </element>
            </sequence>
        </extension>
    </complexContent>
</complexType>
```

Voorbeeld GML fragment:

```
<imro:FeatureCollectionIMRO gml:id="Collectie"
      xmlns:imro="http://www.geonovum.nl/imro2012"
      xmlns:xlink=http://www.w3.org/1999/xlinkxmlns:gml="http://www.opengis.net/gml/3.2"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.geonovum.nl/imro2012 IMRO2012.xsd">

      <imro:featureMember>
               <imro:Bestemmingsplangebied gml:id="NL.IMRO.0999.BP2008000001-0005">
               ...
               </imro:Bestemmingsplangebied>
      </imro:featureMember>
      <imro:featureMember>
         <imro:Enkelbestemming gml:id="NL.IMRO.135">
               ...
               </imro:Enkelbestemming>
      </imro:featureMember>
   ...
</imro:FeatureCollectionIMRO>
```

3.3.2 Geometrietypen

GML Simple Feature Profile staat een beperkte subset van geometrie typen uit GML toe. Deze subset is voor SF0, SF1, en SF2 gelijk.

Tabel 2. Geometrietypen. Van elk type is ook het equivalent in UML gegeven (Spatial Schema):

In GML	UML equivalent	Restricties	Welk GML Schema document
gml:Point	GM_Point	geen	geometryBasic0d1d.xsd
gml:Curve	GM_Curve	gml:LineString of gml:Curve met gml:LineStringSegment, gml:Arc, gml:Circle of gml:CircleByCenterPoint	
gml:Surface	GM_Surface	ondersteund wordt: gml:Polygon of gml:Surface met gml:PolygonPatch patches. Surface grenzen kunnen beschreven worden met gml:LinearRing of gml:Ring met enkele gml:Curve met gml:LineStringSegment, gml:Arc, gml:Circle of gml:CircleByCenterPoint	geometryBasic0d1d.xsd geometryPrimitives.xsd
gml:Geometry		ondersteund wordt: gml:Point,gml:LineString, gml:Curve, gml:Polygon, gml:Surface, gml:MultiPoint, gml:MultiCurve, gml:MultiSurface	geometryBasic0d1d.xsd
gml:MultiPoint		geen	geometryAggregates.xsd
gml:MultiCurve		zelfde als gml:Curve	geometryAggregates.xsd
gml:MultiSurface		zelfde als gml:Surface	geometryAggregates.xsd
gml:MultiGeometry		zelfde als gml:Geometry	geometryAggregates.xsd

Specificatie van coördinaten:
Voor gml:point en gml:CircleByCenterPoint is dit middels het gml:pos element. Voor alle ander typen is het de gml:posList.

3.3.3 Coördinaatreferentiesysteem

In het Simple Features profiel wordt aanbevolen om het coördinaatreferentiesysteem (CRS) van de in een GML document aanwezige coördinaten in het GML document op te nemen. In clausule 9 van NEN 3610:2011 wordt dit zelfs verplicht gesteld.

In het GML bestand dient het coördinaatreferentiesysteem opgenomen te worden als waarde van het in GML gedefinieerde attribuut **srsName**. Aangeraden wordt om dit zo generiek mogelijk te doen. Als alle coördinaten in het document hetzelfde CRS hebben, zou het srsName attribuut op het niveau van de feature collection moeten worden opgenomen. Anders kan srsName worden opgenomen op feature member niveau of op geometrie niveau.

Voor het definiëren van het coördinaatreferentiesysteem kunnen EPSG[4]-codes worden gebruikt.

```
<wfs:FeatureCollection>
    <gml:boundedBy>
        <gml:Envelope srsName="urn:opengis:def:crs:EPSG::28992">...</gml:Envelope>
    </gml:boundedBy>
    <!-- feature instances go here -->
    <wfs:member>
        <myns:MyFeature>
            <gml:boundedBy>
                <gml:Envelope srsName="...">...</gml:Envelope>
            </gml:boundedBy>
            <!-- zero or more property values go here -->
            <myns:geomProperty>
                <gml:Polygon srsName="...">...</gml:Polygon>
            </myns:geomProperty>
            <!-- zero or more property values go here -->
        </myns:MyFeature>
    </wfs:member>
    <!-- more feature instances go here -->
</wfs:FeatureCollection>
```

Container elementen voor samengestelde geometrietypen
Geometrietypen die zijn samengesteld fungeren als een container (aggregatie) voor de verschillende individuele geometrie elementen. De volgende container elementen zijn toegestaan.

Tabel 3. Samengestelde geometrietypen

Samengesteld geometrie type	Container element
gml:MultiPoint	gml:pointMember
gml:MultiCurve	gml:curveMember
gml:MultiSurface	gml:surfaceMember
gml:MultiGeometry	gml:geometryMember

Lijninterpolaties
Het type lijninterpolatie beschrijft hoe de verbinding tussen twee coördinaatparen (punten) is beschreven. In de GML standaard is hiervoor een uitgebreid aantal mogelijkheden. Het Simple Features profiel ondersteunt hiervan maar twee:lineaire interpolatie en interpolatie door middel van cirkelbogen. Deze laatste niet lineaire interpolatie was geen onderdeel van het SF profiel van de vorige GML versie (3.1). Vanwege het toegenomen gebruik in applicaties zijn ze nu wel toegevoegd.

In de tabel in paragraaf 3.3.2 is precies te zien welke typen lijninterpolatie zijn toegestaan door te kijken naar de restricties bij gml:Curve en gml:Surface. Toegestaan voor het opnemen van rechte lijnen zijn alleen gml:LineString en gml:Curve met gml:lineStringSegments. Voor het opnemen van bogen zijn alleen gml:Arc, gml:Circle en gml:CircleByCenterPoint toegestaan. gml:Arc is de GML implementatie van de cirulaire boog met drie punten (beginpunt, eindpunt en punt waar de boog doorheen loopt).

De overige typen lijninterpolatie, zoals gml:CubicSpline, gml:ArcByBulge, etc, zijn niet toegestaan.

3.4 Simple features profile 0 (SF0)

SF0 is het meest eenvoudige profiel op GML, m.a.w. het profiel met de meeste restricties op GML en XML Schema.

[4] European Petroleum Survey Group

Regels:

- Waarden van een eigenschap mogen inline (genest) of door verwijzing worden opgenomen. Er moet wel een keuze gemaakt worden voor één van de twee manieren: het is niet toegestaan om in een SF0 GML document de ene keer een waarde inline op te nemen en een andere keer te verwijzen. Verwijzingen mogen alleen opgenomen worden met gml:ReferenceType. Het gml:ReferenceType is de simpele GML implementatie van in UML als associaties gemodelleerde koppelingen tussen objecttypen. De naam van de gekoppelde bron (objecttype) wordt opgenomen in een appinfo element, zodat te zien is naar wat voor soort ding er gerefereerd wordt. De daadwerkelijke verwijzing naar een object instantie wordt met xlink:href gedaan.
- Beperkte set van ingebouwde niet-geometrische typen. Andere eenvoudige of samengestelde typen uit XML Schema of zelf gedefinieerd, zijn niet toegestaan, alleen de volgende:
    1. String:
        a. Een element dat een alfanumerieke waarde heeft moet zijn van het XML Schema type string of een restrictie daarvan. De maxLength of length mogen worden gebruikt om de string verder in te perken.
        b. Een element dat een alfanumerieke waarde heeft waarvan de taal moet worden aangegeven, moet hiervoor gebruik maken van het xml:lang attribuut waarin de taalcode is opgegeven.
    2. Integer: Een element met een integer waarde moet zijn van het XML Schema-type integer of een restrictie daarvan. In het totalDigits element kan worden aangegeven hoe lang het getal mag zijn.
    3. Measurement: Een element dat een meting als waarde heeft moet van het type gml:MeasureType zijn of zijn afgeleid.
    4. Date: Een element dat een datum als waarde heeft, moet zijn van een van de XML Schema typen date of dateTime of een restrictie daarvan.
    5. Real: Een element dat een real waarde moet bevatten, moet zijn van een van de XML Schema typen double of decimal of een restrictie daarvan. Bij een decimal mogen de facetten totalDigits en fractionDigits worden opgegeven, bij een double niet. Meestal zullen elementen met real waarden overigens metingen zijn en dus met een gml:MeasureType worden opgenomen.
    6. binary: Een element dat een binaire inhoud heeft, moet dit ofwel inline in het GML bestand coderen met het XML Schema type base64Binary of hexBinary; of ernaar verwijzen met een URL. In elk geval is het opnemen van de mimeType verplicht.
    7. boolean: Een element dat een boolean (ja/nee) waarde heeft, moet van het XML Schema type boolean zijn.
    8. URI: een element dat een URI als waarde heeft, moet van het XML Schema type anyURI zijn.
- De niet geometrische typen mogen qua waarde verder alleen ingeperkt worden met de XML Schema facetten minInclusive, minExclusive, maxInclusive, maxExclusive, en enumeration. Hiermee kan men de minimale en maximale waarden van getallen inperken en enumeraties definiëren.
- Een element eigenschap mag niet meerdere keren voorkomen; de kardinaliteit die mag worden opgegeven is beperkt tot 0 of 1.
- Het XML Schema attribuut nillable mag niet gebruikt worden.
- Het aantal toegestane GML geometrie typen is ingeperkt. Dit is dezelfde subset voor SF0, SF1, en SF2; zie3.3.2.
- Features mogen meerdere geometrische eigenschappen hebben.

3.5 Simple features profile 1 (SF1)

SF1 kent minder restricties dan SF0 en is dan ook iets complexer.

SF1 neemt de regels van SF0 over, met uitzondering van het volgende:

- Het is toegestaan om eigen datatypen te maken met een eenvoudige waarde, maar wel gebaseerd op de set van toegestane niet-geometrische typen uit SF0.

- Het is toegestaan om samengestelde datatypen te maken die een eigen, geneste structuur hebben.
- Eigenschappen mogen meerdere keren voorkomen.
- Het XML Schema attribuut nillable mag gebruikt worden.

3.5.1 Zelf ontwikkelde datatypen: geneste inhoud.

In SF1 en SF2 kan een geneste structuur van attributen toegepast worden. Attributen kunnen daarbij samengesteld worden uit een verzameling attributen. We noemen deze constructie het samengesteld attribuut.

Een voorbeeld is een attribuut met de naam 'identificatie' en een datatype 'identifier'. Het datatype bestaat vervolgens uit drie attributen 'identificatiecode', 'organisatie', 'versie'. Het attribuut 'identificatie' is in dit geval samengesteld uit drie andere attributen, verenigd in het zelfgemaakte datatype 'identifier'.

In een GML application schema ziet die constructie als volgt uit:

```
<complexType name="IdentifierPropertyType">
    <sequence>
        <element ref="fw:Identifier"/>
    </sequence>
</complexType>
<element name="Identifier">
    <complexType>
        <sequence>
            <element name="identificatiecode" type="string"/>
            <element name="organisatie" type="string" minOccurs="0" maxOccurs="1"/>
            <element name="versie" type="string" minOccurs="0" maxOccurs="1"/>
        </sequence>
    </complexType>
</element>
```

3.6 Simple features profile 2 (SF2)

SF2 is het minst restrictieve niveau in het profiel en is geschikt voor geavanceerde informatiemodellen. Vergeleken met SF1 vervallen in SF2 alle regels over het definiëren van niet-geometrische eigenschappen. D.w.z. alle typen uit de XML Schema standaard mogen gebruikt worden, eigen eenvoudige en samengestelde typen mogen worden gedefinieerd, er zijn geen restricties op kardinaliteit, etc.

Waarden van een eigenschap mogen in SF2 inline (genest) of door verwijzing worden opgenomen. Een mix van beide mag gebruikt worden en gebruik van gml:ReferenceType is niet verplicht.

Wel blijft de restrictie op de toegestane geometrietypen gelden, net als de regel over het definiëren van feature collecties.

3.7 Schema compliance: data levering – data ontvangen

Het Simple Features profile van GML 3.1 was in technisch opzicht een gestripte versie van de volledige GML XSD. Dit bleek echter bij veel validatietools problemen op te leveren. De gestripte XSD had dezelfde namespace als de volledige GML XSD waardoor conflicten konden optreden.

Bij het Simple Features profile van GML 3.2 is daarom een andere werkwijze gekozen. Een GML applicatieschema dat voldoet aan een bepaald level van het Simple Features profile, moet hoewel de GML standaard niet volledig hoeft te worden ondersteund, de volledige GML XSD en de gmlsfLevels.xsd importeren en in de header aangeven welk SF level is toegepast. Dit dient te gebeuren door een annotation op te nemen binnen het schema element:

```
<schema xmlns="http://www.w3.org/2001/XMLSchema"
    xmlns:gmlsf="http://www.opengis.net/gmlsf/2.0">
    <import namespace="http://www.opengis.net/gml/3.2"
    schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
    <import namespace="http://www.opengis.net/gmlsf/2.0"
        schemaLocation="http://schemas.opengis.net/gmlsfProfile/2.0/gmlsfLevels.xsd"/>
    <annotation>
        <appinfo
            source="http://schemas.opengis.net/gmlsfProfile/2.0/gmlsfLevels.xsd">
            <!--hier een 0, 1 of 2 invullen-->
             <gmlsf:ComplianceLevel>2</gmlsf:ComplianceLevel>
        </appinfo>
    </annotation>
    ...
</schema>
```

In plaats van een absolute URL naar de locatie van de schema's in de schema repository van de OGC, is het ook toegestaan een absolute of relatieve URL naar een andere locatie te gebruiken, zolang het schema maar opgehaald kan worden.

Het is toegestaan om andere application schema's te importeren, maar alleen als die voldoen aan hetzelfde Simple Features profile niveau of een lager niveau (en dus minder complex zijn) en dit op de voorgeschreven manier aangeven in de header.

Om te controleren of een GML bestand voldoet aan de regels van het Simple Features profile, kan de GML xsd niet worden gebruikt omdat er geen gestripte versie meer beschikbaar is. In plaats daarvan wordt aangeraden om een Schematron[5] validatie te gebruiken. Voor SF2 is een document met Schematron regels beschikbaar in Annex E van het Simple Features profile for GML. Deze regels zijn beschreven in het volgende hoofdstuk, en zijn als validatie ook beschikbaar in de schema repository van de OGC[6] en de online validator van Geonovum[7].

[5] Zie http://www.schematron.com/
[6] http://schemas.opengis.net/gmlsfProfile/2.0/gmlsfL2.sch
[7] http://www.geonovum.nl/validatie

GML SF2: Nederlands profiel op GML 3.2

Vanwege de voordelen van voldoen aan internationale conventies en omdat het GML Simple Features profile level 2(SF2) bijna voldoet aan de eisen die er in Nederland gesteld worden is er besloten om SF2 als Nederlandse standaard voor de implementatie van GML 3.2.1 te nemen. In het vorige hoofdstuk is de inhoud van dat profiel toegelicht. In dit hoofdstuk wordt het profiel beschreven middels businessrules die van toepassing zijn op een GML document dat voldoet aan SF 2. De businessrules zijn opgesteld in Schematron, een taal ontwikkeld voor het valideren van XML documenten.

4.1 Introductie Schematron

Schematron gaat in tegenstelling tot XML Schema niet uit van een definitie van de complete XML structuur van een set documenten, maar stelt in staat om regels op te stellen voor specifieke onderdelen van XML documenten. Schematron kan daardoor gebruikt worden om heel gericht enkele regels te controleren. Voor het opstellen van de regels zijn er veel mogelijkheden, waaronder het toetsen van onderlinge afhankelijkheden tussen onderdelen van de XML structuur, wat met XML Schema niet mogelijk is. Terwijl XML Schema te vergelijken is met UML, is Schematron verwant aan de Object Constraint Language (OCL). Beide zijn de implementatie van hun meer conceptuele equivalent.

De in Schematron opgestelde regels kunnen voor validatie van XML (GML) documenten worden gebruikt. Een deel van de bekende XML editors kan validaties tegen Schematron regels uitvoeren of de Schematron controle kan worden uitgevoerd als XSLT transformatie. Validatie tegen de Schematron implementatie van SF2 kan ook via de Geonovum website worden uitgevoerd[8].

De nu volgende paragraaf geeft de Schematron regels die voor het valideren tegen GML SF2 zijn opgesteld. Door de toelichting bij elke regel wordt duidelijk gemaakt hoe op dat bepaalde onderwerp SF2 verschilt van de complete GML standaard.

4.2 Schematron business rules voor Nederlands profiel

De hieronder weergegeven business rules omvatten alleen de SF2 regels. De controle beperkt zich tot met name de restricties met betrekking tot geometrietypen. De restricties op XML Schema, die voor SF0 en SF1 gelden, zijn niet opgenomen. Deze Schematron regels kunnen echter wel gebruikt worden om tenminste de geometrie restricties voor een SF0 of SF1 bestand te valideren. Deze zijn immers voor SF0, SF1 en SF2 gelijk.

Deze schematron regels zijn ontleend aan het OGC Simple Feature Profile for GML. De nieuwste versie is te vinden op http://schemas.opengis.net/gmlsfProfile/2.0/.

Geen gebruik van gml:metaDataProperty.

```
<sch:assert test="not(gml:metaDataProperty)">This profile prohibits use of
gml:metaDataProperty elements for referencing metadata in instance documents.</sch:assert>
```

Geen topologie elementen

Geen voorkomens van Node, Edge, Face, TopoSolid, TopoPoint, TopoCurve, TopoSurface, TopoVolume, or TopoComplex.

```
<sch:assert test="not(self::gml:Node|self::gml:Edge|self::gml:Face|self::gml:TopoSolid
|self::gml:TopoPoint|self::gml:TopoCurve|self::gml:TopoSurface
|self::gml:TopoVolume|self::gml:TopoComplex)">Spatial properties are limited to the set of
geometric types consisting of point, curve with linear and/or circular arc interpolation,
planar surface, or aggregates thereof. Spatial topology is excluded.</sch:assert>
```

[8] http://validatie-dataspecificaties.geostandaarden.nl/genericvalidator/content/standard/157

Interpolatie van curve segmenten
Toegestaan binnen een gml:Curve zijn LineStringSegment, Arc, Circle, or CircleByCenterPoint

```
<sch:assert test="not(self::gml:Curve)
orself::gml:Curve/gml:segments[gml:LineStringSegment|gml:Arc|gml:Circle|gml:CircleByCenter
Point]">Curves (standalone or within surfaces) must have linear and/or circular arc
interpolation (LineString, Curve with Arc, Circle or CircleByCenterpoint
segments)</sch:assert>
```

Surface is Polygon of Surface.
Geen voorkomens van OrientableSurface, CompositeSurface, PolyhedralSurface, Tin, or TriangulatedSurface.

```
<sch:asserttest="not(self::gml:OrientableSurface|
self::gml:CompositeSurface|self::gml:PolyhedralSurface|self::gml:Tin|self::gml:Triangulate
dSurface)">Planar surface types are restricted to Polygon or Surface
elements.</sch:assert>
```

Beperkt aantal geometrytypen.
Geen Solid, MultiSolid, CompositeSolid, CompositeCurve of Grid.

```
<sch:asserttest="not(self::gml:Solid|self::gml:MultiSolid|
self::gml:CompositeSolid|self::gml:CompositeCurve|self::gml:Grid)">
Supported geometry types are restricted to point, curve with linear and/or circular arc
interpolation, planar surface, or aggregates thereof.
</sch:assert>
```

Punt coordinaten met gml:pos element.
Binnen een gml:Point moeten de coordinaten zijn opgenomen in een gml:pos element.

```
<sch:assert test="count(self::gml:Point/gml:pos) = count(self::gml:Point/*)">Geometry
coordinates shall only be specified using the gml:pos element for gml:Point.</sch:assert>
```

Gml:CircleByCenterPoint gebruikt gml:pos element.
Binnen een gml:CircleByCenterPoint moeten de coordinaten zijn opgenomen in een gml:pos element.

```
<sch:assert test="count(self::gml:CircleByCenterPoint/gml:pos|
self::gml:CircleByCenterPoint/gml:radius)
 = count(self::gml:CircleByCenterPoint/*)">Geometry coordinates shall only be specified
using the gml:pos element for gml:CircleByCenterPoint.
</sch:assert>
```

Gml:posList element voor coordinaten in andere dan punt geometrien.
Gml:LineString:
Binnen een gml:LineString moeten de coordinaten zijn opgenomen in een gml:posList element.

```
<sch:assert test="count(self::gml:LineStringSegment/gml:posList)
=count(self::gml:LineStringSegment/*)">Geometry coordinates shall only be specified using
the gml:posList element for gml:LineStringSegment.
</sch:assert>
```

Gml:LinearRing
Binnen een gml:LinearRing moeten de coordinaten zijn opgenomen in een gml:posList element.

```
<sch:assert test="count(self::gml:LinearRing/gml:posList) =
count(self::gml:LinearRing/*)">Geometry coordinates shall only be specified using the
gml:posList element for gml:LinearRing.</sch:assert>
```

Gml:Arc
Binnen een gml:Arc moeten de coordinaten zijn opgenomen in een gml:posList element.

```
<sch:assert test="count(self::gml:Arc/gml:posList) =
count(self::gml:Arc/*)">Geometry coordinates shall only be specified using the gml:posList
element for gml:Arc.</sch:assert>
```

gml:Circle

Binnen een gml:Circle moeten de coordinaten zijn opgenomen in een gml:posList element.

```
<sch:assert test="count(self::gml:Circle/gml:posList) =
count(self::gml:Circle/*)">Geometry coordinates shall only be specified using the
gml:posList element for gml:Circle.</sch:assert>
```

Toegestane container elementen:

gml:pointMember

```
<sch:assert test="not(self::gml:MultiPoint/gml:pointMembers)">This profile restricts
instance documents to using the property container gml:pointMember for the MultiPoint
geometry type.</sch:assert>
```

gml:curveMember

```
<sch:assert test="not(self::gml:MultiCurve/gml:curveMembers)">This profile restricts
instance documents to using the property container gml:curveMember for the MultiCurve
geometry type.</sch:assert>.
```

gml:surfaceMember

```
<sch:assert test="not(self::gml:MultiSurface/gml:surfaceMembers)">This profile restricts
instance documents to using the property container gml:surfaceMember for the MultiSurface
geometry type.</sch:assert>
```

gml:geometryMember

```
<sch:assert test="not(self::gml:MultiGeometry/gml:geometryMembers)">This profile restricts
instance documents to using the property container gml:geometryMember for the
MultiGeometry geometry type.</sch:assert>
```

Surface elementen zijn altijd gml:PolygonPatch

```
<sch:assert test="count(self::gml:Surface/gml:patches/gml:PolygonPatch) =
count(self::gml:Surface/gml:patches/*)">The content of gml:Surface elements is restricted
to gml:PolygonPatch patches.</sch:assert>
```

De dimensie van het coordinaatreferentiesysteem is 1, 2 of 3

```
<sch:assert test="not(self::*/@srsDimension > 3)">coordinate reference systems may have 1,
2 or 3 dimensions</sch:assert>
```

Basisregels voor transformatie van een SF2 naar een SF0 model.

SF2 is de standaard voor informatiemodellen in Nederland. Toch is het in sommige gevallen gewenst om te werken met een eenvoudiger informatiemodel, dat met name voor software die de complexere zaken in SF2 niet ondersteunt, te begrijpen is.

5.1 Inleiding

De beperkingen die SF0 oplegt aan de GML structuur op het gebied van datastructurering en datatypen maken dit niveau het meest verwant aan een gangbare databasestructuur. Dit kan bijvoorbeeld het importeren van een GML bestand of het serveren van GML data in een web service sterk vereenvoudigen.

Als het voor een implementatie nodig is dat de GML bestanden aan SF0 voldoen in plaats van het in Nederland algemeen geldende SF2, kan er naast het conceptuele informatiemodel, dat voldoet aan SF2, een afgeleide versie worden gemaakt die voldoet aan SF0. Dit brengt een aantal wijzigingen in het informatiemodel met zich mee, die in dit hoofdstuk besproken worden.

De in dit hoofdstuk opgenomen voorbeelden zijn fictief en hebben geen relatie met een actueel model.

5.2 Verschil SF0 en SF2

Wie een informatiemodel moet omzetten dat voldoet aan SF2 naar een variant die voldoet aan het strengere SF0, moet met de volgende aspecten rekening houden:

- Van de XML Schema typen mogen alleen string, integer, date, real, binary, boolean, en URI gebruikt worden, plus het GML type measurement.
- Eigen, eenvoudige of samengestelde datatypen zijn niet toegestaan.
- Eigenschappen mogen geen meervoudige kardinaliteit hebben.
- Verwijzingen moeten met gml:ReferenceType worden vastgelegd.
- XML Schema 'nillable' functionaliteit mag niet gebruikt worden.

5.3 Normalisatieregels

Om een informatiemodel volgens SF0 te implementeren op basis van een SF2 informatiemodel kunnen de volgende normalisatieregels gevolgd worden.

5.3.1 Meervoudig attribuut

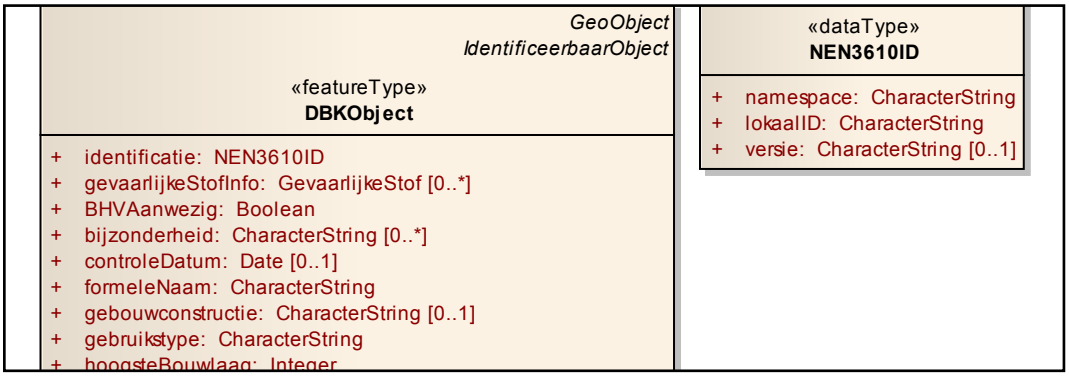
In SF2 is het toegestaan om attributen (eigenschappen van features) meerdere keren te laten voorkomen. In SF0 mag elk attribuut maar één keer voorkomen.

Er zijn twee oplossingen waaruit gekozen kan worden:

1. Beperk het aantal toegestane voorkomens van het attribuut tot één, en neem indien er meerdere waarden aanwezig zijn, deze op in het attribuut, met een scheidingsteken (bijvoorbeeld ';') tussen elke twee waarden.
   Het voordeel hiervan is dat men onbeperkt waarden kan opnemen. Het nadeel is dat er geen enumeratie of ander datatype anders dan een (alfanumerieke) string kan worden toegekend aan het attribuut. Ook is er extra moeite nodig om eventueel de individuele waarden weer uit te lezen.
2. Neem voor het attribuut zoveel voorkomens op als redelijkerwijs nodig is. Als een attribuut bijvoorbeeld in de praktijk niet meer dan 3 keer voorkomt, kan men hiervoor drie attributen opnemen in het informatiemodel (bijvoorbeeld: naam1, naam2 en naam3). De attributen moeten altijd beginnen met dezelfde inhoudelijke naam en eindigen met een cijferaanduiding.

5.3.2 Zelfgemaakte datatypen

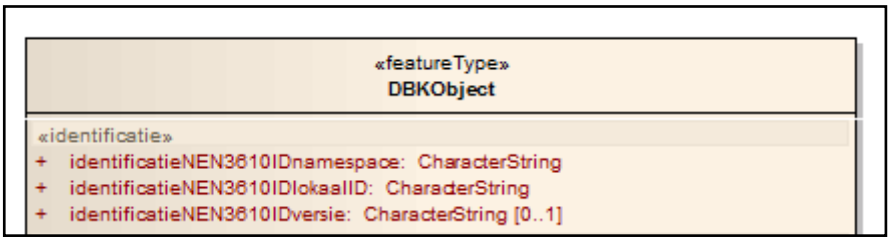
Complexe datatypen, oftewel samengestelde attributen, met eigen geneste structuur, zijn in SF2 wel toegestaan, maar in SF0 niet.



Figuur 9 Klasse met zelfgemaakt, complex datatype 'identificatie' in SF2 model

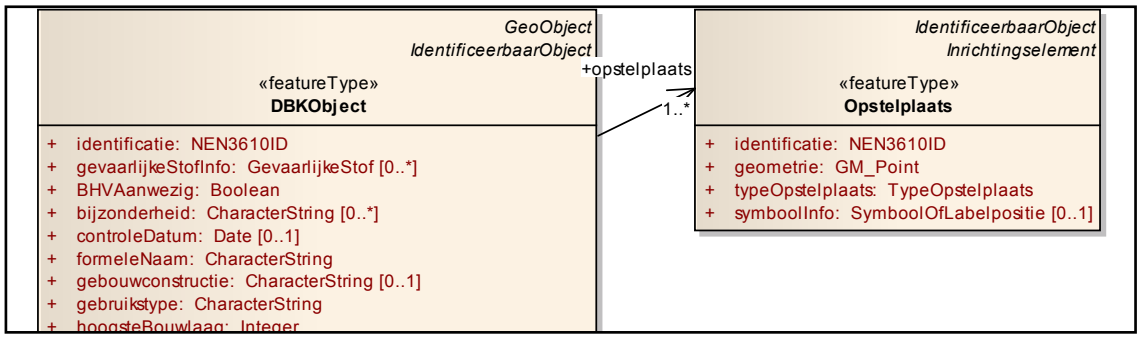
Samengestelde attributen worden in een aan SF0 voldoend informatiemodel opgenomen als losse attributen. Met een gelijknamig stereotype dat bij elk van de attributen die bij het samengestelde attribuut horen wordt opgenomen, wordt in het UML informatiemodel aangegeven dat de attributen bij elkaar horen. Het stereotype wordt in de UML klasse weergegeven boven de attributen die het gemeenschappelijk hebben tussen <<>>.

Daarnaast krijgen deze attributen een gemeenschappelijke prefix, die aangeeft dat ze een groep vormen. Zo zijn ze in een GML bestand ook te herkennen als bij elkaar horende groep.



Figuur 10 Klasse met samengesteld attribuut in SF0 model

5.3.3 Associaties



Figuur 11 Associatie in een SF2 UML model

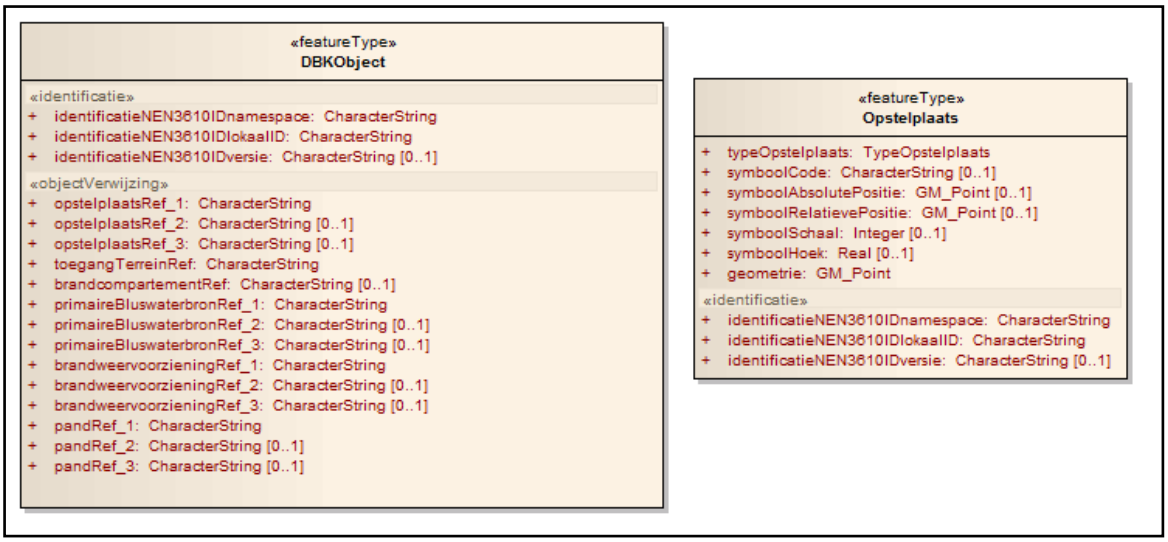
Ook associaties zijn in een SF2 informatiemodel wel toegestaan, maar in een SF0 model niet. Associaties worden in een GML bestand gecodeerd als ofwel een geneste structuur, ofwel een link. In SF0 zijn geneste structuren niet toegestaan (zie de vorige paragraaf) en mogelijk links/verwijzingen alleen worden uitgedrukt met behulp van het gml:ReferenceType.

Om het zo eenvoudig mogelijk te houden geldt voor de vertaling van associaties uit een SF2 model naar een SF0 model dat deze in plaats van als associaties tussen klassen in het model worden opgenomen als attributen:

- met het stereotype <<objectVerwijzing>>,
- en met een suffix 'Ref'.

Als de associatie meerdere keren mag voorkomen, wordt dit op de zelfde manier opgelost als bij meervoudige attributen.

In het attribuut wordt de identificatie opgenomen van het object waarnaar verwezen wordt.



Figuur 12 Klasse in SF0 model met objectVerwijzingen

Bijlage 1 Algemene coderingsregels GML

Hoe uit een UML informatiemodel uiteindelijk een XML Schema wordt gemaakt is aan regels gebonden die in ISO 19118 en in annex E van de OGC standaard GML beschreven zijn. Dit hoofdstuk geeft een overzicht van de belangrijkste van deze regels.

1.1 Van UML naar GML

In ISO/TC 211 en OGC en daarom ook in NEN3610, wordt het semantisch model in UML vastgelegd. Vervolgens wordt via een geautomatiseerde procedure een XML/GML Schema uit het UML model gegenereerd. Op deze wijze wordt geborgd dat er geen verschillen ontstaan tussen het semantisch model in UML en het technisch model in het uitwisselformaat XML/GML. Het is ook toegestaan om handmatig een GML-conform XML Schema te maken.

Het semantisch model, vastgelegd in UML, wordt een UML Application Schema genoemd. Het bijbehorende XML Schema wordt een GML Application Schema genoemd. De term 'application' duidt er daarbij op dat het om een *toepassing* van GML gaat voor een specifiek domein/sector.

Het informatiemodel wordt gemodelleerd middels één of meer UML klassendiagrammen. De klassen worden gegroepeerd in UML packages, één voor elk te genereren schema bestand. Aan de packages en klassen kan extra informatie worden toegekend die gebruikt wordt om het genereren van het schema te sturen. Dit wordt gedaan door een stereotype toe te kennen aan het package of de klasse. GML definieert een standaard set aan stereotypen (GML 3.2.1 Annex E).

Bij een package kan bijvoorbeeld worden aangegeven dat het van het stereotype 'ApplicationSchema' is waarna kan worden aangegeven wat de gewenste target namespace, namespace prefix en bestandsnaam zijn van het te genereren XML Schema. Bij een klasse kan men aangeven dat het een 'FeatureType' is (geo-object) waarna bij de klasse metadata specifiek voor FeatureTypes kan worden ingevuld. Een klasse kan echter ook bijvoorbeeld van het stereotype 'Union' zijn.

Een klasse waarvan is aangegeven dat het een FeatureType is, zal volgens de GML encoding rules voor Features worden opgenomen in het te genereren XML Schema als een globaal XML element in de substitutionGroup van gml:AbstractFeature met een globaal complexType dat een extensie is van gml:AbstractFeatureType. Een klasse waarvan is aangegeven dat het een Union is, zal heel anders naar het XML Schema vertaald worden, namelijk als een complexType met een choice.

Voor het genereren van het XML Schema wordt een tool[9] gebruikt die op basis van een UML model een of meer GML Application Schema's genereert volgens de door de GML standaard gestelde eisen. Wie deze tool gebruikt voor het genereren van het GML application schema hoeft zich dus geen zorgen meer te maken over de GML encoding rules.

1.2 GML encoding rules

Het XML Schema moet voldoen aan de eisen die GML eraan stelt (GML 3.2.1 clause 21 en delen van clause 7), en, indien uit UML gegenereerd, aan de eisen die GML stelt aan het vertalen van UML model naar XML Schema (GML 3.2.1 Annex E, ISO 19118).

Het GML XML Schema bevat abstracte typen, die in een GML Application Schema worden uitgebreid, en concrete typen die direct kunnen worden gebruikt.

De belangrijkste eisen aan GML Application schema's zijn:

---

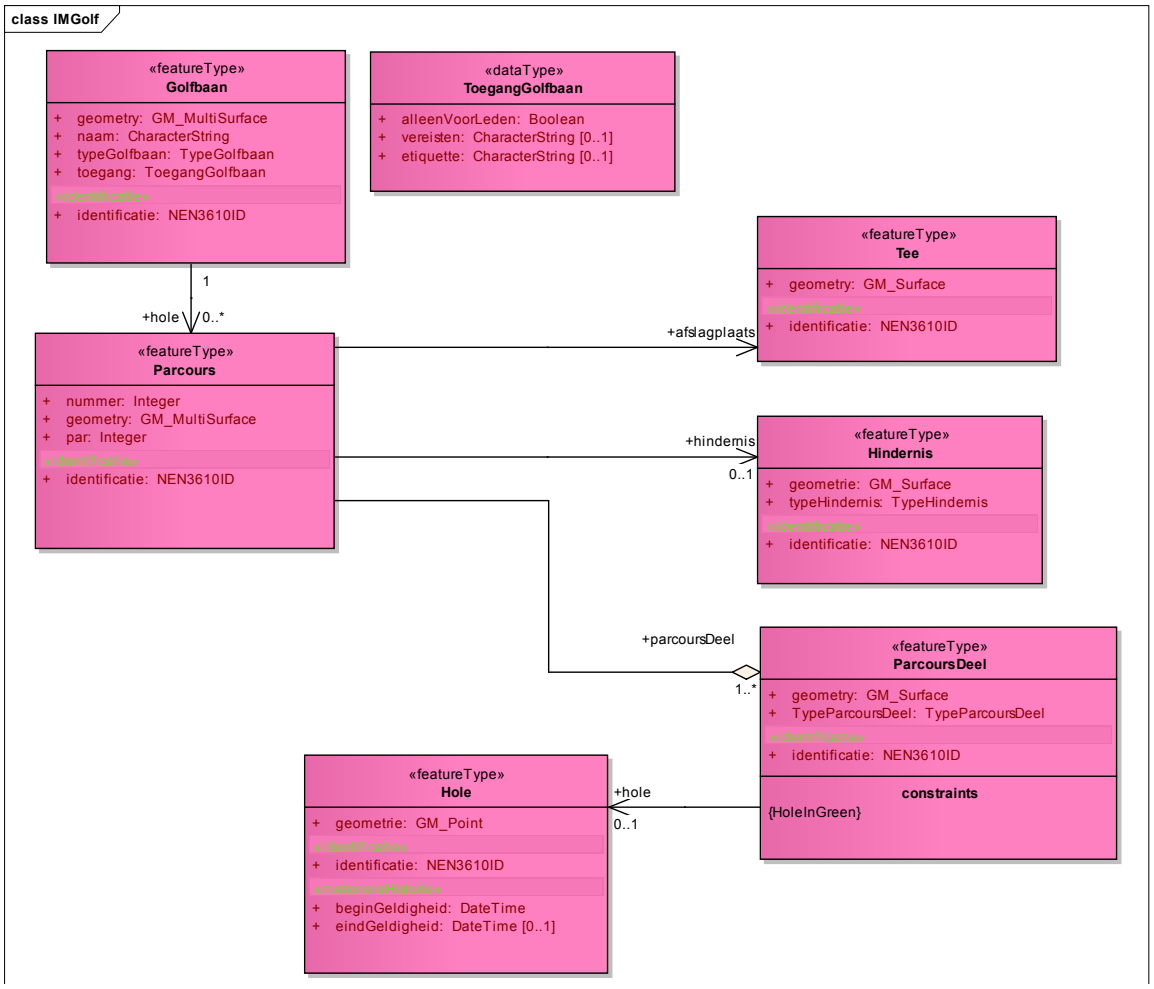
[9] ShapeChange van interactive instruments, zie http://www.interactive-instruments.de/index.php?id=28&L=1

- Clause 21.2.2: Het GML application schema definieert eigen uitbreidingen op het GML XML Schema in een eigen targetNamespace.
- Clause 21.2.3: Het GML application schema importeert het GML XML Schema.
- Clause 21.3.3:
    - Voor elke klasse die een geo-object vertegenwoordigt moet een globaal complexType en een globaal element gedefinieerd zijn.
    - Elk complexType voor een geo-object moet direct of indirect een extensie zijn van het type "gml:AbstractFeatureType". AbstractFeatureType is op zijn beurt een extensie van AbstractGMLType. Van dit type erven alle objecten het verplichte gml:id attribuut.
    - Elk element in het schema voor een geo-object moet lid zijn van de substitutionGroup voor het element "gml:AbstractFeature" (dat wil zeggen: moet zich direct of indirect in de substitutionGroup="gml:AbstractFeature" bevinden).
    - Elk identificeerbaar object dat geen geo-object is, moet lid zijn van de substitutionGroup voor het element "gml:AbstractGML".
- Clause 21.3.4: Alle eigenschappen van een entiteit en alle relaties naar andere entiteiten worden opgenomen als child XML element.
- Clause 21.2.7: Als de waarde van een eigenschap een enkelvoudige waarde is, moet deze als literal value worden opgenomen in het eigenschap element zonder verdere markup. Zie ook 7.2.3.10.
  Voorbeeld:
  GOED:
  ```
  <gml:Integer>5</gml:Integer>
  ```
  FOUT:
  ```
  <gml:Integer>
  <gml:value>5</gml:value>
  </gml:Integer>
  ```
- Clause 21.2.6: Het object property model moet gevolgd worden. Toelichting: GML is een objectenmodel. Dat wil zeggen dat alles is gemodelleerd als ofwel een object, ofwel een eigenschap van een object. Een relatie naar een ander object wordt ook gezien als eigenschap. Een object kan niet direct een ander object bevatten, maar een eigenschap kan wel een object bevatten. Dit wordt het object-property model genoemd. Als een eigenschap dus een complexe waarde heeft (structuur bezit) moet deze complexe waarde als apart object worden gedefinieerd. Een eigenschap mag zelf geen verdere structuur hebben. Het eigenschap element kan een verwijzing naar dit object bevatten of kan het als geneste structuur opnemen. Zie ook Clause 7.2.3.
- Clause 21.2.1/21.3.4: Het gebruik van XML attributen is toegestaan (maar niet voor het opnemen van eigenschappen van objecten). Attributen worden niet in een namespace gedeclareerd met uitzondering van gml:id. Zie ook clause 7.1.3.

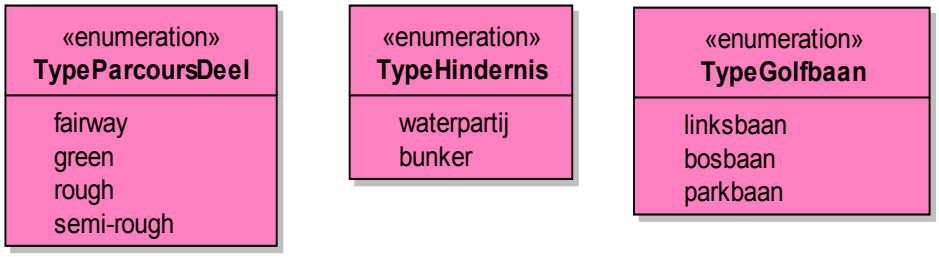
Bijlage 2 Uitgewerkt voorbeeld

In deze bijlage wordt aan de hand van een eenvoudig model de relatie tussen een UML een daarvan afgeleid XML Schema en een concreet GML document weergegeven.

2.1 IMGolf model in UML



Figuur 13 IMGolf klassediagram



Figuur 14 Enumeraties van IM Golf model

2.2 IMGolf Feature Catalogus

Applicatie schema IMGolf

Feature catalogusmetadata

Naam van feature catalogus	IMGolf
Scope	IMGolf
Versienummer	1.0
Versiedatum	2012-02-20
Herkomst Definities	Dataspecificatie IMGolf

Types gedefinieerd in de feature catalogus

Type	Package	Stereotypes	Section
Golfbaan	IMGolf	«featureType»	
Hindernis	IMGolf	«featureType»	
Hole	IMGolf	«featureType»	
NEN3610ID	IMGolf	«dataType»	
Parcours	IMGolf	«featureType»	
ParcoursDeel	IMGolf	«featureType»	
Tee	IMGolf	«featureType»	
ToegangGolfbaan	IMGolf	«dataType»	
TypeGolfbaan	IMGolf	«enumeration»	
TypeHindernis	IMGolf	«enumeration»	
TypeParcoursDeel	IMGolf	«enumeration»	

Geo object types

Golfbaan	
Definitie:	Een golfbaan is een perceel waarop de sport golf wordt gespeeld.
Herkomst:	http://nl.wikipedia.org/wiki/Golfbaan
Omschrijving:	De baan bestaat uit een reeks holes.
Stereotypes:	«featureType»
Attribuut: geometry	
Type:	GM_MultiSurface
Definitie:	Vlakgeometrie van het totale golfterrein.
Omschrijving:	Omvat alle parcoursen (holes) op de golfbaan maar ook de kantine, het parkeerterrein, de oefengreen en de drivingrange.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: naam	
Type:	CharacterString
Definitie:	De benaming van de golfbaan.
Omschrijving:	De naam wordt weergegeven in de lijst van alle golfbanen van nederland.
Multipliciteit:	1
Herkomst:	IMGolf

Golfbaan	
Attribuut: typeGolfbaan	
Type:	TypeGolfbaan
Definitie:	Het soort golfbaan, ingedeeld naar het type terrein waar zij zich bevindt.
Omschrijving:	De indeling is in de loop der tijd ontstaan doordat een golfbaan anders speelt, afhankelijk van de locatie. Een linksbaan bijvoorbeeld bevindt zich vlakbij de kust waardoor de wind een grotere factor is.
Multipliciteit:	1
Herkomst:	http://nl.wikipedia.org/wiki/Golfbaan
Attribuut: toegang	
Type:	ToegangGolfbaan
Definitie:	Informatie over de toegang tot de golfbaan.
Omschrijving:	De informatie bestaat uit meerdere onderdelen.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: identificatie	
Type:	NEN3610ID
Multipliciteit:	1
Stereotypes:	«identificatie»
Association role: hole	
Type:	Parcours
Definitie:	Een van de parcoursen op het golfterrein.
Omschrijving:	Een parcours, meestal Hole genoemd, is een terrein met verschillende onderdelen zoals het afslagpunt, de fairway, de green en de hole.
Multipliciteit:	0..*
Herkomst:	IMGolf

Hindernis	
Definitie:	Een obstakel op het parcours in de vorm van een zand- of waterpartij.
Herkomst:	IMGolf.
Omschrijving:	Om een bal uit een bunker te slaan wordt vaak een speciale club gebruik; als een bal in het water terecht komt ben je verder van huis.
Stereotypes:	«featureType»
Attribuut: geometrie	
Type:	GM_Surface
Definitie:	Vlakgeometrie
Omschrijving:	De coordinaten van het vlak waar het obstakel zich bevindt.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: typeHindernis	
Type:	TypeHindernis
Definitie:	Het soort hindernis.
Omschrijving:	Dit is altijd een zandbunker of een waterpartij. Ook bomen en incidentele waterplassen kunnen een hindernis zijn, maar deze zijn niet gemodelleerd. Bomen worden wellicht in de 2.0 versie opgenomen.
Multipliciteit:	1
Herkomst:	IMGolf.
Attribuut: identificatie	
Type:	NEN3610ID

Hindernis	
Definitie:	Unieke identificatie van het object binnen het domein van NEN 3610.
Multipliciteit:	1
Stereotypes:	«identificatie»
Herkomst:	NEN 3610:2011

Hole	
Definitie:	Met een vlag gemarkeerd punt waarin de bal geslagen moet worden.
Herkomst:	IMGolf.
Omschrijving:	De hole wordt regelmatig van plek veranderd, omdat de green dan niet onregelmatig slijt.
Stereotypes:	«featureType»
Attribuut: geometrie	
Type:	GM_Point
Definitie:	Puntgeometrie
Omschrijving:	Wijzigt regelmatig. De nieuwe Hole lokaties worden verspreid onder de leden en deelnemers aan wedstrijden.
Multipliciteit:	1
Herkomst:	
Attribuut: beginGeldigheid	
Type:	DateTime
Definitie:	Start van het tijdstip waarop de hole op deze locatie ligt
Omschrijving:	Dit attribuut implementeert materiele historie volgens NEN3610:2010
Multipliciteit:	1
Stereotypes:	«materieleHistorie»
Attribuut: eindGeldigheid	
Type:	DateTime
Definitie:	Eind van het tijdstip waarop de hole op deze locatie ligt. Wanneer dit de huidige locatie is heeft dit attribuut geen waarde
Omschrijving:	Dit attribuut implementeert materiele historie volgens NEN3610:2010
Multipliciteit:	0..1
Stereotypes:	«materieleHistorie»
Attribuut: identificatie	
Type:	NEN3610ID
Definitie:	Unieke identificatie van het object binnen het domein van NEN 3610.
Multipliciteit:	1
Stereotypes:	«identificatie»
Herkomst:	NEN 3610:2011

Parcours	
Definitie:	Een van de speelvelden van een golfbaan.
Herkomst:	http://nl.wikipedia.org/wiki/Hole_%28golf%29
Omschrijving:	Ook wel 'Hole' genoemd. Hier wordt de term Parcours gebruikt om verwarring met de hole waar de bal ingeslagen moet worden, te voorkomen. Ieder parcours bestaat in de regel uit de volgende onderdelen: Tee-box, Green, Hole, Fairway, Rough, Fringe / Apron, Hindernissen.
Stereotypes:	«featureType»
Attribuut: nummer	
Type:	Integer

Parcours	
Definitie:	De getalaanduiding voor het parcours.
Omschrijving:	Meestal heeft een golfbaan 18 'holes' of parcoursen. De eerste heeft als waarde van dit attribuut 1, de laatste 18.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: geometry	
Type:	GM_MultiSurface
Definitie:	Multi-vlak geometrie
Omschrijving:	Coordinaten van het vlak van het parcours. Het kan een multivlak zijn.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: par	
Type:	Integer
Definitie:	het aantal slagen waarin een gemiddelde professionele golfer een hole zou moeten kunnen spelen.
Omschrijving:	Par staat voor Professional Average Result.
Multipliciteit:	1
Herkomst:	http://nl.wikipedia.org/wiki/Par_(golf)
Attribuut: identificatie	
Type:	NEN3610ID
Definitie:	Unieke identificatie van het object binnen het domein van NEN 3610.
Multipliciteit:	1
Stereotypes:	«identificatie»
Herkomst:	NEN 3610:2011
Association role: afslagplaats	
Type:	Tee
Association role: parcoursDeel	
Type:	ParcoursDeel
Definitie:	Een deelterrein van het parcours.
Multipliciteit:	1..*
Association role: hindernis	
Type:	Hindernis
Definitie:	Een obstakel tussen de tee en de hole in een parcours.
Multipliciteit:	0..1

ParcoursDeel	
Definitie:	Deel van het terrein van de hole (parcours).
Herkomst:	http://nl.wikipedia.org/wiki/Golfbaan#Onderdelen_van_een_golfbaan
Omschrijving:	Elk parcours heeft verschillende onderdelen zoals de green, de fairway, de semi-rough en de rough.
Stereotypes:	«featureType»
Attribuut: geometry	
Type:	GM_Surface
Definitie:	Vlakgeometrie.
Omschrijving:	Voor elk onderdeel van een parcours worden hier de coordinaten van het vlak opgenomen. Als bijvoorbeeld de fairway in twee stukken wordt verdeeld door een waterpartij, worden dit twee aparte parcoursdelen met ieder een vlak.
Multipliciteit:	1

ParcoursDeel	
Herkomst:	IMGolf
Attribuut: TypeParcoursDeel	
Type:	TypeParcoursDeel
Definitie:	Het soort parcoursdeel.
Omschrijving:	Een parcours kan worden verdeeld in een aantal onderdelen, zoals fairway of green.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: identificatie	
Type:	NEN3610ID
Definitie:	Unieke identificatie van het object binnen het domein van NEN 3610.
Multipliciteit:	1
Stereotypes:	«identificatie»
Herkomst:	NEN 3610:2011
Association role: hole	
Type:	Hole
Definitie:	Het punt (gat), gemarkeerd door een vlag, waar de bal in geslagen moet worden.
Multipliciteit:	0..1
Constraint: HoleInGreen	
Natuurlijke taal:	Inside(hole.geometrie,this->geometrie)
OCL:	

Tee	
Definitie:	De afslagplaats aan het begin van een hole.
Herkomst:	http://nl.wikipedia.org/wiki/Tee_(golf)
Omschrijving:	Ook wel tee-box genoemd.
Stereotypes:	«featureType»
Attribuut: geometry	
Type:	GM_Surface
Definitie:	Vlakgeometrie
Omschrijving:	De exacte plaats op de tee waar men moet beginnen wordt aangegeven door een tee-marker, waarvan er meerdere kunnen zijn. Meestal zijn er aparte tee-markers voor dames en heren om te compenseren voor het verschil in kracht. In dit model wordt alleen het vlak opgenomen, waarbinnen alle tee-markers zich bevinden.
Multipliciteit:	1
Herkomst:	http://nl.wikipedia.org/wiki/Golfbaan
Attribuut: identificatie	
Type:	NEN3610ID
Definitie:	Unieke identificatie van het object binnen het domein van NEN 3610.
Multipliciteit:	1
Stereotypes:	«identificatie»
Herkomst:	NEN 3610:2011

Data types

NEN3610ID	
Stereotypes:	«dataType»
Attribuut: namespace	

NEN3610ID	
Type:	CharacterString
Definitie:	unieke verwijzing naar een registratie van objecten
Omschrijving:	Het attribuut 'namespace' is een unieke verwijzing naar de registratie die de identificatie uitdeelt. Deze lijst van registraties wordt beheerd binnen de context van NEN 3610. Binnen Nederland zal deze namespace vrijwel altijd met 'NL.' beginnen. De volgende karakters mogen in een namespace aanduiding voorkomen: {"A"…"Z", "a"…"z", "0"…"9", "_", "- ", ",", "."}
Multipliciteit:	1
Herkomst:	NEN 3610:2011
Attribuut: lokaalID	
Type:	CharacterString
Definitie:	unieke identificatiecode binnen een registratie
Omschrijving:	'LokaalId' is de identificatiecode die een object heeft binnen een (lokale) registratie. De volgende karakters mogen in een lokaalID voorkomen: {"A"…"Z", "a"…"z", "0"…"9", "_", "-", ",", "."}.
Multipliciteit:	1
Herkomst:	NEN 3610:2011
Attribuut: versie	
Type:	CharacterString
Definitie:	versie-aanduiding van een object
Omschrijving:	Het attribuut 'versie' maakt geen deel uit van de identificatie van het object maar kan worden gebruikt om verschillende versies van hetzelfde object te identificeren.
Multipliciteit:	0..1
Herkomst:	NEN 3610:2011

ToegangGolfbaan	
Definitie:	Informatie over de toegang tot de golfbaan.
Herkomst:	IMGolf
Omschrijving:	Bestaat uit verschillende onderdelen. Later worden misschien nog de openingstijden toegevoegd.
Stereotypes:	«dataType»
Attribuut: alleenVoorLeden	
Type:	Boolean
Definitie:	Geeft aan of de golfbaan alleen voor leden van de vereniging toegankelijk is.
Omschrijving:	Als de golfbaan niet alleen voor leden is, is ze dus openbaar en mag iedere golfer er komen spelen.
Multipliciteit:	1
Herkomst:	IMGolf
Attribuut: vereisten	
Type:	CharacterString
Definitie:	Harde vereisten voor toegang tot de golfbaan.
Omschrijving:	Hier kan bijvoorbeeld worden opgenomen dat er toegangsvereisten zijn zoals het hebben van een baanpermissie of GVB.
Multipliciteit:	0..1
Herkomst:	IMGolf
Attribuut: etiquette	
Type:	CharacterString
Definitie:	Beschrijving van de niet-harde regels op de golfbaan.

ToegangGolfbaan	
Omschrijving:	Dit zijn de regels waaraan men zich formeel niet hoeft te houden, maar er wordt wel zeer de prijs op gesteld als je het doet. Hier kan bijvoorbeeld de kledingetiquette worden beschreven.
Multipliciteit:	0..1
Herkomst:	IMGolf

Enumeraties en codelijsten

TypeGolfbaan	
Definitie:	Het soort golfbaan.
Herkomst:	IMGolf
Omschrijving:	Indeling naar terreintype.
Stereotypes:	«enumeration»
Waarde: linksbaan	
Definitie:	Bevindt zich in het gebied vlak langs de kustlijn en kenmerkt zich door het feit dat ze open is (weinig bomen) en relatief hoge rough heeft. Ook zijn de greens en fairways doorgaans vrij hard.
Waarde: bosbaan	
Definitie:	Bevindt zich doorgaans in bosrijke gebieden.
Waarde: parkbaan	
Definitie:	Bevindt zich niet in een bosrijk gebied of aan de kust.

TypeHindernis	
Definitie:	Het soort obstakel.
Herkomst:	IMGolf
Omschrijving:	Een waterpartij of zandbunker.
Stereotypes:	«enumeration»
Waarde: waterpartij	
Definitie:	Een met water gevulde hindernis op een golfbaan.
Waarde: bunker	
Definitie:	Een met zand gevulde hindernis op een golfbaan.

TypeParcoursDeel	
Definitie:	Het soort parcoursdeel.
Herkomst:	IMGolf
Omschrijving:	Een fairway, green, rough, of semi-rough.
Stereotypes:	«enumeration»
Waarde: fairway	
Definitie:	Tussen de green en de tee-box gelegen terrein waar het gras wat korter is dan in het eromheen gelegen terrein.
Waarde: green	
Definitie:	Het laatste deel van het parcours, waar het gras zeer kort wordt gemaaid.
Waarde: rough	
Definitie:	Onderdeel van een Hole bestaande uit het niet speciaal geprepareerd terrein.
Waarde: semi-rough	
Definitie:	Het minder kort geschoren gras naast de fairway.

2.3 IMGolf XML applicatie schema

```
<?xml version="1.0" encoding="UTF-8"?>
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:IMGolf="http://www.geonovum.nl/IMGolf"
  xmlns:gml="http://www.opengis.net/gml/3.2"
xmlns:gmlsf="http://www.opengis.net/gmlsf/2.0"
  elementFormDefault="qualified" targetNamespace="http://www.geonovum.nl/IMGolf"
version="1.0">
  <import namespace="http://www.opengis.net/gml/3.2"
    schemaLocation="http://schemas.opengis.net/gml/3.2.1/gml.xsd"/>
  <import namespace="http://www.opengis.net/gmlsf/2.0"
    schemaLocation="http://schemas.opengis.net/gmlsfProfile/2.0/gmlsfLevels.xsd"/>
  <!--GML Applicatie Schema voor voorbeeld van NEN3610 Versie 1.0 datum 2012-02-20-->
  <annotation>
    <appinfo source="http://schemas.opengis.net/gmlsfProfile/2.0/gmlsfLevels.xsd">
      <gmlsf:ComplianceLevel>2</gmlsf:ComplianceLevel>
    </appinfo>
  </annotation>
  <element name="FeatureCollectionIMGolf" type="IMGolf:FeatureCollectionIMGolfType"
    substitutionGroup="gml:AbstractGML"/>
  <complexType name="FeatureCollectionIMGolfType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence minOccurs="0" maxOccurs="unbounded">
          <element name="featureMember">
            <complexType>
              <complexContent>
                <extension base="gml:AbstractFeatureMemberType">
                  <sequence>
                    <element ref="gml:AbstractFeature"/>
                  </sequence>
                </extension>
              </complexContent>
            </complexType>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>

  <element name="Hole" substitutionGroup="gml:AbstractFeature" type="IMGolf:HoleType"/>
  <complexType name="HoleType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="geometrie" type="gml:PointPropertyType"/>
          <element name="identificatie" type="IMGolf:NEN3610IDPropertyType"/>
          <element name="beginGeldigheid" type="dateTime"/>
          <element minOccurs="0" name="eindGeldigheid" type="dateTime"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="HolePropertyType">
<!--     <sequence minOccurs="0">
```

```
      <element ref="IMGolf:Hole"/>
    </sequence>-->
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
    <attributeGroup ref="gml:OwnershipAttributeGroup"/>
  </complexType>
  <simpleType name="TypeHindernisType">
    <restriction base="string">
      <enumeration value="waterpartij"/>
      <enumeration value="bunker"/>
    </restriction>
  </simpleType>
  <element name="ToegangGolfbaan" substitutionGroup="gml:AbstractObject"
    type="IMGolf:ToegangGolfbaanType"/>
  <complexType name="ToegangGolfbaanType">
    <sequence>
      <element name="alleenVoorLeden" type="boolean"/>
      <element minOccurs="0" name="vereisten" type="string"/>
      <element minOccurs="0" name="etiquette" type="string"/>
    </sequence>
  </complexType>
  <complexType name="ToegangGolfbaanPropertyType">
    <sequence>
      <element ref="IMGolf:ToegangGolfbaan"/>
    </sequence>
  </complexType>
  <simpleType name="TypeGolfbaanType">
    <restriction base="string">
      <enumeration value="linksbaan"/>
      <enumeration value="bosbaan"/>
      <enumeration value="parkbaan"/>
    </restriction>
  </simpleType>
  <element name="Parcours" substitutionGroup="gml:AbstractFeature"
type="IMGolf:ParcoursType"/>
  <complexType name="ParcoursType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="identificatie" type="IMGolf:NEN3610IDPropertyType"/>
          <element name="nummer" type="integer"/>
          <element name="geometry" type="gml:MultiSurfacePropertyType"/>
          <element name="par" type="integer"/>
          <element maxOccurs="unbounded" minOccurs="0" name="afslagplaats"
            type="IMGolf:TeePropertyType">
            <annotation>
              <appinfo>
                <targetElement
xmlns="http://www.opengis.net/gml/3.2">IMGolf:Tee</targetElement>
              </appinfo>
            </annotation>
          </element>
          <element maxOccurs="unbounded" name="parcoursDeel"
type="IMGolf:ParcoursDeelPropertyType">
            <annotation>
              <appinfo>
                <targetElement
```

```
xmlns="http://www.opengis.net/gml/3.2">IMGolf:ParcoursDeel</targetElement>
              </appinfo>
            </annotation>
          </element>
          <element minOccurs="0" name="hindernis" type="IMGolf:HindernisPropertyType">
            <annotation>
              <appinfo>
                <targetElement
xmlns="http://www.opengis.net/gml/3.2">IMGolf:Hindernis</targetElement>
              </appinfo>
            </annotation>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="ParcoursPropertyType">
    <!--<sequence minOccurs="0">
      <element ref="IMGolf:Parcours"/>
    </sequence>-->
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
    <attributeGroup ref="gml:OwnershipAttributeGroup"/>
  </complexType>
  <element name="ParcoursDeel" substitutionGroup="gml:AbstractFeature"
    type="IMGolf:ParcoursDeelType"/>
  <complexType name="ParcoursDeelType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="geometry" type="gml:SurfacePropertyType"/>
          <element name="identificatie" type="IMGolf:NEN3610IDPropertyType"/>
          <element name="TypeParcoursDeel" type="IMGolf:TypeParcoursDeelType"/>
          <element name="hole" type="IMGolf:HolePropertyType" minOccurs="0">
            <annotation>
              <appinfo>
                <targetElement
xmlns="http://www.opengis.net/gml/3.2">IMGolf:Hole</targetElement>
              </appinfo>
            </annotation>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="ParcoursDeelPropertyType">
<!--    <sequence minOccurs="0">
      <element ref="IMGolf:ParcoursDeel"/>
    </sequence>-->
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
    <attributeGroup ref="gml:OwnershipAttributeGroup"/>
  </complexType>
  <element name="Tee" substitutionGroup="gml:AbstractFeature" type="IMGolf:TeeType"/>
  <complexType name="TeeType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
```

```
          <element name="geometry" type="gml:SurfacePropertyType"/>
          <element name="identificatie" type="IMGolf:NEN3610IDPropertyType"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="TeePropertyType">
<!--    <sequence minOccurs="0">
      <element ref="IMGolf:Tee"/>
    </sequence>-->
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
    <attributeGroup ref="gml:OwnershipAttributeGroup"/>
  </complexType>
  <element name="NEN3610ID" substitutionGroup="gml:AbstractObject"
type="IMGolf:NEN3610IDType"/>
  <complexType name="NEN3610IDType">
    <sequence>
      <element name="namespace" type="string"/>
      <element name="lokaalID" type="string"/>
      <element minOccurs="0" name="versie" type="string"/>
    </sequence>
  </complexType>
  <complexType name="NEN3610IDPropertyType">
    <sequence>
      <element ref="IMGolf:NEN3610ID"/>
    </sequence>
  </complexType>
  <simpleType name="TypeParcoursDeelType">
    <restriction base="string">
      <enumeration value="fairway"/>
      <enumeration value="green"/>
      <enumeration value="rough"/>
      <enumeration value="semi-rough"/>
    </restriction>
  </simpleType>
  <element name="Hindernis" substitutionGroup="gml:AbstractFeature"
type="IMGolf:HindernisType"/>
  <complexType name="HindernisType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="geometrie" type="gml:SurfacePropertyType"/>
          <element name="identificatie" type="IMGolf:NEN3610IDPropertyType"/>
          <element name="typeHindernis" type="IMGolf:TypeHindernisType"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="HindernisPropertyType">
<!--    <sequence minOccurs="0">
      <element ref="IMGolf:Hindernis"/>
    </sequence>-->
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
    <attributeGroup ref="gml:OwnershipAttributeGroup"/>
  </complexType>
  <element name="Golfbaan" substitutionGroup="gml:AbstractFeature"
```

```
type="IMGolf:GolfbaanType"/>
  <complexType name="GolfbaanType">
    <complexContent>
      <extension base="gml:AbstractFeatureType">
        <sequence>
          <element name="geometry" type="gml:MultiSurfacePropertyType"/>
          <element name="naam" type="string"/>
          <element name="typeGolfbaan" type="IMGolf:TypeGolfbaanType"/>
          <element name="toegang" type="IMGolf:ToegangGolfbaanPropertyType"/>
          <element name="identificatie" type="IMGolf:NEN3610IDPropertyType"/>
          <element maxOccurs="unbounded" minOccurs="0" name="hole"
            type="IMGolf:ParcoursPropertyType">
            <annotation>
              <appinfo>
                <targetElement
xmlns="http://www.opengis.net/gml/3.2">IMGolf:Parcours</targetElement>
              </appinfo>
            </annotation>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="GolfbaanPropertyType">
<!--    <sequence minOccurs="0">
      <element ref="IMGolf:Golfbaan"/>
    </sequence>-->
    <attributeGroup ref="gml:AssociationAttributeGroup"/>
    <attributeGroup ref="gml:OwnershipAttributeGroup"/>
  </complexType>
</schema>
```

2.4 IMGolf GML document

```
<?xml version="1.0" encoding="UTF-8"?>
<FeatureCollectionIMGolf xmlns:xlink="http://www.w3.org/1999/xlink"
   xmlns:gml="http://www.opengis.net/gml/3.2"
   xmlns="http://www.geonovum.nl/IMGolf" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance"
   xsi:schemaLocation="http://www.geonovum.nl/IMGolf IMGolf.xsd" gml:id="ID001">
   <gml:boundedBy>
      <gml:Envelope srsDimension="2" srsName="urn:opengis:def:crs:EPSG::xxxxx">
         <gml:lowerCorner>10 10</gml:lowerCorner>
         <gml:upperCorner>20 20</gml:upperCorner>
      </gml:Envelope>
   </gml:boundedBy>
   <featureMember>
      <Golfbaan gml:id="G11">
         <geometry>
            <gml:MultiSurface gml:id="x00">
               <gml:surfaceMember>
                  <gml:Polygon gml:id="x2">
                     <gml:exterior>
                        <gml:LinearRing>
                           <gml:posList>10 10 20 10 20 20 10 20 10 10</gml:posList>
```

```
                    </gml:LinearRing>
                 </gml:exterior>
              </gml:Polygon>
           </gml:surfaceMember>
        </gml:MultiSurface>
     </geometry>
     <naam>Openbare golfbaan Westlaren</naam>
     <typeGolfbaan>bosbaan</typeGolfbaan>
     <toegang>
        <ToegangGolfbaan>
           <alleenVoorLeden>false</alleenVoorLeden>
        </ToegangGolfbaan>
     </toegang>
     <identificatie>
        <NEN3610ID>
           <namespace>NL.IMGOLF</namespace>
           <lokaalID>G1</lokaalID>
        </NEN3610ID>
     </identificatie>
     <hole xlink:href="#Hin1v1"/>
  </Golfbaan>
</featureMember>
<featureMember>
  <Parcours gml:id="Hin1v1">
     <identificatie>
        <NEN3610ID>
           <namespace>NL.IMGOLF</namespace>
           <lokaalID>Hin1</lokaalID>
           <versie>1</versie>
        </NEN3610ID>
     </identificatie>
     <nummer>1</nummer>
     <geometry>
        <gml:MultiSurface gml:id="x01">
           <gml:surfaceMember>
              <gml:Polygon gml:id="x3">
                 <gml:exterior>
                    <gml:LinearRing>
                       <gml:posList>11 11 15 11 15 19 11 19 11 11</gml:posList>
                    </gml:LinearRing>
                 </gml:exterior>
              </gml:Polygon>
           </gml:surfaceMember>
        </gml:MultiSurface>
     </geometry>
     <par>5</par>
     <afslagplaats xlink:href="#t01"/>
     <parcoursDeel xlink:href="#pd01"/>
     <parcoursDeel xlink:href="pd02"/>
  </Parcours>
</featureMember>
<featureMember>
  <Tee gml:id="t01">
     <geometry>
        <gml:Polygon gml:id="x4">
           <gml:exterior>
```

```
                <gml:LinearRing>
                   <gml:posList>12 12 14 12 14 13 12 13 12 12</gml:posList>
                </gml:LinearRing>
             </gml:exterior>
          </gml:Polygon>
       </geometry>
       <identificatie>
          <NEN3610ID>
             <namespace>NL.IMGOLF</namespace>
             <lokaalID>t01</lokaalID>
          </NEN3610ID>
       </identificatie>
    </Tee>
 </featureMember>
 <featureMember>
    <ParcoursDeel gml:id="pd01">
       <geometry>
          <gml:Polygon gml:id="x06">
             <gml:exterior>
                <gml:LinearRing>
                   <gml:posList>12 13 14 13 14 17 12 17 12 13</gml:posList>
                </gml:LinearRing>
             </gml:exterior>
          </gml:Polygon>
       </geometry>
       <identificatie>
          <NEN3610ID>
             <namespace>NL.IMGOLF</namespace>
             <lokaalID>pd01</lokaalID>
          </NEN3610ID>
       </identificatie>
       <TypeParcoursDeel>fairway</TypeParcoursDeel>
    </ParcoursDeel>
 </featureMember>
 <featureMember>
    <ParcoursDeel gml:id="pd02">
       <geometry>
          <gml:Polygon gml:id="x07">
             <gml:exterior>
                <gml:LinearRing>
                   <gml:posList>12 17 14 17 14 18.5 12 18.5 12 17</gml:posList>
                </gml:LinearRing>
             </gml:exterior>
          </gml:Polygon>
       </geometry>
       <identificatie>
          <NEN3610ID>
             <namespace>NL.IMGOLF</namespace>
             <lokaalID>pd01</lokaalID>
          </NEN3610ID>
       </identificatie>
       <TypeParcoursDeel>green</TypeParcoursDeel>
       <hole xlink:href="#h01"/>
    </ParcoursDeel>
 </featureMember>
 <featureMember>
```

```
      <Hole gml:id="h01">
         <geometrie>
            <gml:Point gml:id="x5">
               <gml:pos>13 18</gml:pos>
            </gml:Point>
         </geometrie>
         <identificatie>
            <NEN3610ID>
               <namespace>NL.IMGOLF</namespace>
               <lokaalID>h01</lokaalID>
            </NEN3610ID>
         </identificatie>
         <beginGeldigheid>2011-12-01T00:00:00</beginGeldigheid>
      </Hole>
   </featureMember>
</FeatureCollectionIMGolf>
```

Bijlage 3 GML 3.2.1 en GML 3.1.1

In deze bijlage wordt ingegaan op de verschillen tussen de GML versies 3.2.1 en 3.1.1.

3.1 Inleiding

In de praktijk zijn er momenteel drie versies van GML in gebruik:
- GML 2.1.2 (2002)
- GML 3.1.1 (2004/2005)
- GML 3.2.1 (2007)

GML 3.3 is op het moment van schrijven net uitgekomen (begin 2012). Deze versie is volledig backwards compatible met GML 3.2.1 en bestaat uit enkele uitbreidingen die in nieuwe namespaces zijn gedefinieerd. GML 3.3 wordt hier verder buiten beschouwing gelaten.

In Nederland is tot nu toe met name gestandaardiseerd op GML 3.1.1. De overstap naar GML 3.2.1 wordt pas de laatste tijd gemaakt, omdat eerder deze versie nog niet altijd ondersteund werd door software applicaties. Sommige nieuwe standaarden maken nog steeds gebruik van GML 3.1.1: dit geldt bijvoorbeeld voor IMGeo 2.0 vanwege de aansluiting met CityGML[10], waarin een afhankelijkheid met GML 3.1.1 zit.

Dit appendix beschrijft de verschillen tussen 3.1.1 en 3.2.1, zodat:
- De afweging tussen beide versies goed gemaakt kan worden;
- De impact duidelijk wordt voor zij die de overstap van 3.1.1 naar 3.2.1 overwegen;
- Gebruikers van GML 3.1.1 de standaard zo kunnen toepassen dat de stap naar 3.2.1 als het moment daar is zo klein mogelijk wordt.

Alleen de verschillen die GML onderdelen raken die in Nederland waarschijnlijk of mogelijk gebruikt worden, zijn hier opgesomd. Een volledige beschrijving van de verschillen is te vinden in de officiële Revision Notes voor GML 3.2.1 (OGC documentnummer 07-061)[11], of in het hierop gebaseerde document Analysis of changes between GML 3.1.1 and GML 3.2.1, te vinden op de website van Geonovum.

Hieronder volgt een opsomming en beschrijving van de belangrijkste verschillen.

3.2 Waar iedereen mee te maken krijgt

Importeren van het GML schema en GML profielen
Vanaf GML 3.2.1 moet een GML application schema het volledige GML schema importeren. In GML 3.1.1 was het nog toegestaan een gestript GML schema te importeren. Dit onderwerp is al behandeld in paragraaf 3.7.

XML namespace
De XML namespace van het GML schema ("http://www.opengis.net/gml") is gewijzigd naar "http://www.opengis.net/gml/3.2".

Dit betekent dat alle componenten van GML nu in een andere namespace staan en dus andere componenten zijn.
- Dit heeft een kleine of grote impact op software applicaties die GML 3.1.1 ondersteunen, afhankelijk van de wijze van implementeren.
- GML bestanden in 3.1.1 kunnen eenvoudig handmatig worden aangepast als de namespace declaratie op één plek in de header staat. Als de wijziging op meer plekken moet worden doorgevoerd kan dit geautomatiseerd gebeuren.

---

[10] OGC standaard voor 3D geo-informatie. http://www.opengeospatial.org/standards/citygml
[11] http://portal.opengeospatial.org/files/?artifact_id=26765

- GML application schemas moeten (uiteraard) het GML 3.2.1 schema met de nieuwe namespace importeren in plaats van het GML 3.1.1 schema met de oude namespace. Dit heeft alleen impact op de header van het schema.

Abstracte elementen hernoemd
De "_" (underscore) in element namen om aan te geven dat ze abstract zijn, is vervangen door het beginnen van de naam met "Abstract".

Dit betekent dat bijvoorbeeld het GML element _Feature is hernoemd naar AbstractFeature.
- In GML application schemas die handmatig van 3.1.1 naar 3.2.1 gemigreerd worden, moet bij alle elementen die afgeleid zijn van een abstract GML element, de naam van het abstracte GML element worden gewijzigd.
- Kan impact hebben op software.
- Geen impact op GML documenten.

gml:id verplicht
GML objecten hebben identiteit en daarom is het attribuut gml:id verplicht geworden. Ook geometrieën zoals gml:Point, gml:Curve en gml:Surface worden in GML gezien als objecten en krijgen nu een verplicht gml:id.

Gevolgen:
- Software die wordt gebruikt om GML documenten te maken, moet nu een gml:id aanmaken, met een binnen het document unieke waarde, voor elk GML object. In GML 3.1.1 hoefde dit niet en vaak werd dit dan ook niet gedaan voor objecten waarnaar niet werd verwezen, bijvoorbeeld voor een geometrie object binnen een feature.
- GML documenten die van 3.1.1 naar 3.2.1 worden gemigreerd, moeten bij elk object dat nog geen gml:id had, er een toevoegen met een unieke waarde. Dit kan geautomatiseerd gedaan worden.
- Geen gevolgen voor GML application schemas.

Aanbeveling: wie nu nog GML 3.1.1 gebruikt maar binnen afzienbare tijd naar GML 3.2.1 (of GML 3.3) wil overstappen, kan het beste vast beginnen het gml:id overal op te nemen waar dit in GML 3.2.1 verplicht wordt.

Feature collection en andere collection types geschrapt
Alle in GML 3.1.1 aanwezige collection typen, inclusief die voor feature collections, zijn geschrapt. Het is nu de bedoeling date en GML application schema zelf een collection definieert als dit nodig is, met als basis de nieuwe typen gml:AbstractMemberType en gml:AbstractFeatureMemberType.

Dit heeft gevolgen voor alle GML toepassingen die collecties gebruiken. Zowel de software die collecties maakt of leest, als het GML applicatie schema en GML documenten moeten worden aangepast.

3.3 Andere verschillen met mogelijke gevolgen

Normatieve referenties naar andere standaarden
Een aantal van de normatieve verwijzingen naar aan GML gerelateerde standaard documenten verwijzen nu naar een nieuwere versie daarvan. Hiervan zijn er slechts drie mogelijk relevant voor NEN 3610 gebruikers:

- *ISO 19111:2007, Geographic Information — Spatial referencing by coordinates*. De 2007 revisie van deze standaard is significant gewijzigd.
- *ISO/TS 19139:2007, Geographic Information — Metadata — XML schema implementation*: Metadata elementen uit ISO 19115 worden nu geïmplementeerd door gebruik te maken van ISO/TS 19139.
- *ISO/IEC 19757-3:2006, Document Schema Definition Languages (DSDL) — Part 3: Rule-based validation — Schematron*. Er zijn verschillen tussen de hiervoor gebruikte versie en de in GML 3.2.1 gebruikte door ISO goedgekeurde versie.

Metadata eigenschappen
gml:_MetaData en gml:metaDataProperty zijn geschrapt en vervangen door een abstract property type als basis waarop men eigen metadata eigenschappen kan definiëren. Men kan nu daardoor in het eigen GML applicatie schema afdwingen dat metadata wordt ingevuld. Door het eigen metadata element aft e leiden van het daarvoor bestemde abstracte GML element kan software de metadata herkennen.

Gevolgen als deze GML 3.1.1 elementen gebruikt werden:
- GML applicatie schema moet worden aangepast door het definiëren van een eigen metadata element dat volgens de GML 3.2.1 voorschriften is afgeleid van het abstracte GML type;
- De GML 3.1.1 metadata property moet waar deze is gebruikt in GML documenten worden vervangen door het zelf gedefinieerde metadata element;
- Software die van het GML 3.1.1 metadata element gebruik maakte moet gewijzigd worden om het zelf gedefinieerde metadata element te ondersteunen.

Hernoemde / vervangen eigenschappen
- Alle "*Name" elementen in het coordinate reference system schema zijn vervangen door gml:name en gebruiken nu dus gml:CodeType als type.
    - gml:axisName >gml:axisLabel
    - gml:fileName >gml:fileReference
    - gml:SequenceRulesNames >gml:SequenceRuleEnumeration.
- gml:StringOrRefType is geschrapt. Eigenschappen die een string of een verwijzing als inhoud hebben worden nu gecodeerd als twee aparte eigenschappen in plaats van één.

Wijzigingen in geometrietypen
- Een aantal technische correcties zijn uitgevoerd op geometrietypen. Deze kunnen mogelijk gevolgen hebben voor software:
    - Gml:Envelope is expliciet gedefinieerd als zijnde geen geometrie door het in de substitutionGroup AbstractObject te zetten.
    - gml:CircleByCenterPointType is nu gedefinieerd als een restriction van ArcByCenterPointType.
    - gml:Shell is toegevoegd als implementatie van ISO geometrietype GM_Shell.
    - Het interpolationType attribuut ontbrak op sommige segment typen en is daar toegevoegd.

Wijziging in coordinate reference system dictionaries
De verwijzing in GML naar het gebruikte coördinaatreferentiesysteem (CRS) wordt meestal gedaan door de URN of EPSG code te gebruiken waaronder het bekend staat. Het is echter ook mogelijk de CRS expliciet te definiëren in een GML dictionary. Wie dit in GML 3.1.1 heeft gedaan, krijgt te maken met een wijziging die niet backwards compatible is.

De encoding is ingrijpend gewijzigd om te voldoen aan ISO 19111:2007 en de GML coderingsconventies.

Ondersteuning voor Void (ISO/IEC 11404)
Het weglaten van een kenmerk (middels minOccurs="0" in het schema) is niet hetzelfde als Void zoals gedefinieerd in ISO/IEC 11404:1996. GML 3.2.1 is aangepast om Void correct te implementeren:
- GML 3.2.1 is uitgebreid met een uitleg hoe xsi:nil moet worden gebruikt als implementatie van Void.
- Gml:Null is geschrapt.
- Het type gml:NilReasonType is toegevoegd voor het kunnen opnemen van metadata over de reden van een ontbrekende waarde.

Dit heeft gevolgen voor GML application schemas die gebruik maakten van gml:Null of een eigen implementatie van Void hanteerden. Het GML application schema, GML documenten en software die het GML application schema ondersteunt, moeten aangepast worden.