# STOP tekstmodel

## Inleiding en achtergrond

STOP is gericht op het coderen van (informatie in) overheidspublicaties. Deze publicaties zijn semi‑gestructureerd; ze bevatten een combinatie van relatief ongestructureerde leesbare tekst en gestructureerde metadata.

Het coderen van de leesbare tekst van een publicatie is de kern van STOP. De opeenvolging van tekens, geordend in alinea’s, tabellen, lijsten, etc. is de kapstok waaraan additionele informatie zoals metadata, verwijzingen naar afbeeldingen etc. kan worden opgehangen. Het tekstmodel beschrijft welke entiteiten relevant zijn bij het coderen van de tekst en de additionele informatie.

STOP is in principe syntax neutraal. Entiteiten als “hoofdstuk” of “artikel” kunnen op meerdere manieren vastgelegd worden. In de praktijk wordt XML gebruikt als bronformaat voor de publicaties. De structuur van een publicatie wordt beschreven met XML Schema en additionele business rules via Schematron. De beperkingen en mogelijkheden van XML hebben her en der impact op STOP – dit wordt dan aangegeven.

STOP kent een aantal abstracte entiteiten die gebruikt worden voor classificatie van de niet‑abstracte entiteiten:

* container
* block
* inline
* marker
* structuurelement

In de volgende hoofdstukken wordt nader beschreven welke rol de abstracte entiteiten vervullen, welke implementaties beschikbaar zijn, welke eigenschappen ze hebben, en hoe uitbreidingen kunnen plaatsvinden.

## Abstracte entiteiten

In dit hoofdstuk worden de abstracte entiteiten nader beschreven.

### Container

Een container is een groepering van 1 of meer containers en / of 1 of meer structuurelementen. De container bevat geen tekst en heeft geen default vormgeving. Containers zijn over het algemeen grote logische blokken binnen een publicatie. De inhoud van een container varieert sterk.

### Block

Een block is een container met tekst en kan daarnaast 1 of meer inline elementen en / of 1 of meer marker elementen en / of 1 of meer block elementen bevatten. Een block is daarmee een entiteit die fungeert als container op tekstniveau. Het equivalent van een block in HTML is het div element.

Voorbeelden van block entiteiten binnen STOP zijn alinea, tabel.

### Inline

Een inline bevat tekst en / of inline elementen. Het equivalent van een inline in HTML is het span element. Een voorbeeld van een inline is de aanduiding voor vet, italic, etc. Een inline kan ook gebruikt worden voor meer betekenisvolle aanduidingen zoals bouwhoogte.

Er bestaan enkele beperkingen voor het aanmaken van inlines. Deze beperkingen zijn het gevolg van de manier waarop tekst in XML wordt gecodeerd. Het is niet mogelijk om een inline te definiëren die over de grenzen van een block heengaat.

Er kan een onderscheid gemaakt worden tussen inlines die zelf weer (andere) inlines kunnen bevatten en inlines die alleen tekst mogen bevatten.

### Marker

Een marker is een positie in de tekst waaraan informatie gekoppeld is. In tegenstelling tot een inline omvat een marker geen tekst. Een voorbeeld van een marker is een voetnootreferentie of een afbeelding.

### Structuurelement

Een structuurelement is een speciaal soort container waarmee de formele structuur van een regeling kan worden uitgedrukt. Het gaat om hoofdstuk, paragraaf, etc.

## Entiteiten

In dit hoofdstuk wordt een overzicht gegeven van de (niet-abstracte) STOP entiteiten. Er is gekozen voor een logische onderverdeling (in plaats van een groepering per abstracte entiteit). Details als cardinaliteit worden op deze plaats zoveel mogelijk vermeden.

### Documentniveau

Een officiële publicatie is een document dat door een bevoegd gezag wordt bekendgemaakt en daarmee gepubliceerd. De entiteit **officiele-publicatie** is een container die bestaat uit een metadata container en een container die het publicatieblad representeert (bijvoorbeeld **staatscourant**, **gemeenteblad**, etc.).

Sommige officiële publicaties bevatten wijzigingsinstructies die geconsolideerd moeten worden. Het resultaat van een dergelijke consolidatie is een zogeheten toestand. De entiteit **toestand** is een container die bestaat uit een metadata container (verplicht), een regeling container (verplicht)[[1]](#footnote-1).

Een publicatieblad container (zoals **staatscourant**) bevat een **bladaanduiding** container, een **regeling** container en een **nawerk** container. De regeling is in dit geval de “hoofdtekst” van de bekendmaking, terwijl de optionele **nawerk** container ruimte biedt voor nota toelichting, bijlage en dergelijke.

### Regeling

Een **regeling** bestaat uit[[2]](#footnote-2) een opschrift, aanhef, lichaam, slotformulier, ondertekening, en eventuele bijlagen. De volgorde van deze entiteiten is relevant.

Het **opschrift** is een omschrijving (titel) van de regeling. Het komt overeen met wat voorheen werd aangeduid als "intitulé". Alhoewel het opschrift veelal bestaat uit een enkele string, is er modelmatig in STOP een opschrift container, die bestaat uit een (1) alinea blok, en pas daarbinnen tekst.

Een **aanhef** bevat inleidende tekst over de regeling, waaronder de considerans. Aanhef is een container, die een of meer block(s) bevat (alinea of lijst, geen tabel).

Een **lichaam** is een container die de hoofdtekst van de regeling bevat. Dit komt overeen met regeling‑tekst in voorlopers van STOP. Een lichaam bevat of een tekst volgens artikelstructuur (met optioneel groepering via hoofdstuk, paragraaf, etc.) of een vrije tekst (eventueel geordend via een divisiestructuur).

Het **slotformulier** bevat afsluitende (procedurele) tekst over de regeling. De **ondertekening**bevat zaken datum en plaats van publicatie, en rol plus naam van de ondertekenaar(s). Beide voorgaande entiteiten worden gecombineerd tot een entiteit **regeling-sluiting**. Dit is een container.   
In voorgangers van STOP werd de inhoud van regeling-sluiting zo betekenisvol mogelijk gecodeerd. In de praktijk bleek dit een te overmatig niveau van detail. In STOP wordt daarom gekozen voor generieke alinea entiteiten met een optionele rol eigenschap die dan wel moet gevuld wordt met voorgeschreven waarden (zoals “slotformulier”).

De entiteit **bijlage** is een container die 1 of meer blocks bevat. Dit kunnen tabellen, afbeeldingen, of tekst zijn. Let op: er zijn twee soorten bijlagen binnen STOP. Binnen regelingtekst gaat het om bijlage(n) die formeel tot de regeling behoren (en dus ook op hetzelfde niveau bindend zijn). Buiten regeling (dus in **nawerk**) gaat het om toelichtende bijlagen.

Er bestaat een verschil tussen **officiele-publicatie** en **toestand** voor wat betreft de inhoud van regeling. Bij **toestand** is aanhef en regeling-sluiting niet relevant, en binnen het lichaam is informatie over was-wordt niet relevant (die is immers dan reeds gebruikt bij de totstandkoming van de toestand). Binnen **officiele-publicatie** is was-wordt informatie juist wel relevant voor die publicaties die als basis voor een toestand dienen.

### Artikelen en groepering daarvan in hoofdstuk, paragraaf, etc.

Een **artikel** is een bepaling in een regelgevend document, bijvoorbeeld een wet, een regeling of een verdrag. Het artikel is bedoeld om een concrete regel te geven op het 'laagste' niveau van de regeling[[3]](#footnote-3).

Indien het aantal artikelen toeneemt dan kan omwille van de toegankelijkheid gekozen worden voor een groepering van artikelen. In [aanwijzing 97](http://wetten.overheid.nl/jci1.3:c:BWBR0005730&hoofdstuk=3&paragraaf=3.5&aanwijzing=97&z=2011-05-11&g=2011-05-11) van de [Aanwijzingen voor de regelgeving](http://wetten.overheid.nl/BWBR0005730) wordt hier nader op ingegaan:

2 Bij een verdeling op één niveau worden de onderdelen "hoofdstuk" of "paragraaf" genoemd.

3 Bij een verdeling op twee niveaus worden de onderdelen van het eerste niveau "hoofdstuk" en de onderdelen van het tweede niveau "paragraaf" genoemd.

4 Bij een verdeling op meer dan twee niveaus worden de onderdelen in volgorde van omvang "deel", " hoofdstuk", "titel", " afdeling" en "paragraaf" genoemd, met dien verstande dat in ieder geval de aanduidingen "hoofdstuk" en "paragraaf" worden gebruikt.

In lijn met bovenstaande aanwijzing zijn er binnen STOP entiteiten aangemaakt voor deel, hoofdstuk, titel, afdeling, en paragraaf gemaakt. Via business rules wordt het toegestane gebruik (“als 1 niveau dan hoofdstuk of paragraaf”) afgedwongen. In de praktijk is hier overigens nog wel eens van afgeweken (vandaar dat dit via business rules wordt afgedwongen, en niet via de definitie van de entiteiten zelf).

Zowel **artikel** als de groeperingen **hoofdstuk**, etc. zijn geclassificeerd als structuurelement. Dit laat onverlet dat er een essentieel verschil qua semantiek tussen beide is. Het artikel is de kern: hier staan de bepalingen (de tekst) en dit is de eenheid van wijziging. De groeperingen zijn optioneel, en worden geacht louter redactionele tekst te bevatten (bijvoorbeeld “vervallen”).

Een artikel bevat een kop en daarnaast een keuze tussen vrije tekst (in de vorm van block(s)) of een of meer **lid** entiteiten. De vrije tekst kan bestaan uit alle soorten blocks. Een **lid** is genummerd en bevat tekst[[4]](#footnote-4). Aangezien vrije tekst binnen een artikel ook een lijst kan bevatten – en de lijstitems ook genummerd zijn – kan verwarring ontstaan met leden. In de praktijk is een opsomming veelal gecombineerd met 1 of meer alinea’s. In dat geval moet het om vrije tekst gaan, omdat buiten de leden geen alinea’s kunnen staan.

### Tabellen

Een tabel is een block. Hierdoor kunnen tabellen (als siblings) gecombineerd worden met alinea’s, lijsten en afbeeldingen binnen een container.

STOP ondersteunt het CALS model voor tabellen[[5]](#footnote-5). Omwille van de interoperabiliteit is gekozen voor het [CALS Exchange Table Model](https://www.oasis-open.org/specs/tr9503.html).

In afwijking op voorlopers wordt binnen STOP ook het HTML tabel model ondersteund. Het betreft hier de implementatie zoals deze is opgenomen binnen XHTML 1.0 Strict. Deze elementen worden net zoals bij Akoma Ntoso niet in de XHTML namespace opgeslagen – formeel zijn het dus pseudo XHTML elementen.

### Lijsten

Een **lijst** is geclassificeerd als een block. Deze keuze is met name gebaseerd op het feit hoe lijsten in de tekst worden gebruikt. Gezien de recursieve aard van lijsten en het feit dat de tekst nooit op lijstniveau zit (maar altijd binnen lijst-items) was een keuze voor container ook verdedigbaar geweest.

Een **definitielijst** is een specifieke vorm van een lijst die gericht is op het coderen van een lijst met definities. Een definitielijst bevat 1 of meer **definitie-item** entiteiten. Deze bestaan uit een volgnummer, een term en een definitie.

### Vormgeving

De XML serialisatie heeft geen default vormgeving, anders dan de weergave van Unicode tekst die eraan ten grondslag ligt.

De [bekendmakingsregeling](http://wetten.overheid.nl/BWBR0025218/2015-07-01) bepaalt dat Staatsblad en Staatscourant als PDF A1 (ISO 19005-1:2005) worden gepubliceerd. De [Regeling elektronische bekendmaking en beschikbaarstelling regelgeving decentrale overheden](http://wetten.overheid.nl/BWBR0024947/2016-07-01) bepaalt dat de publicaties worden uitgegeven en beschikbaar gehouden in het bestandsformaat PDF/A-1a (ISO 19005-1:2005) of PDF/A-2a (ISO 19005-2:2011).

De vertaling van XML naar PDF vindt plaats aan de hand van vormgevingsinstructies. Deze leggen een koppeling tussen gewenste vormgeving in de PDF en element- of attribuutnaam in de XML. Hieruit volgen twee implicaties:

1. Vormgeving speelt op het niveau van een “klasse” en wordt zoveel mogelijk ontkoppeld van de inhoud[[6]](#footnote-6);
2. Er zijn zo min mogelijk specifieke “vormgevings”eigenschappen voor entiteiten.

De reden dat er in 2) “zo min mogelijk” staat in plaats van “geen” is dat er in de praktijk soms eisen worden gesteld die verder gaan dan hetgeen mogelijk is via klasse-gerichte vormgeving. Dit betreft bijvoorbeeld de vormgeving van tabellen, waarbij niet altijd algoritmisch de beste oplossing gevonden kan worden. Voor deze gevallen biedt STOP de optionele eigenschap “style” die gevuld moet worden met een geldige CSS 2.1[[7]](#footnote-7) waarde. In toepassingen kan het bereik van style nader worden ingeperkt.

### Werkingsgebied

Het **werkingsgebied** betreft het (ruimtelijk) gebied waarop een publicatie betrekking heeft. Iedere publicatie heeft een expliciet of impliciet werkingsgebied. Indien niet aangegeven (impliciet) dan komt het werkingsgebied overeen met het gebied waarover het bevoegd gezag dat verantwoordelijk is voor de publicatie iets te zeggen heeft[[8]](#footnote-8). Een impliciet werkingsgebied heeft betrekking op documentniveau.

Werkingsgebied kan ook (binnen het document) op structuurelementen worden gespecificeerd (zie voorgaande tekst voor definitie van structuurelementen). Het laagste niveau van structuurelement is artikel. Werkingsgebied kan tevens worden gespecificeerd voor de delen van het artikel die via een Juriconnect verwijzing kunnen worden geadresseerd. Dit zijn lid, volzin en opsommingsonderdeel.

In alle gevallen geldt voor werkingsgebieden binnen het document dat de hiërarchie van de tekst wordt gevolgd: een onderliggend werkingsgebied moet ruimtelijk gezien gelijk zijn aan of binnen een bovenliggend werkingsgebied liggen.

Indien op structuurelement niveau een werkingsgebied wordt gespecificeerd, dan moet op documentniveau een expliciet werkingsgebied gedeclareerd zijn.

De waarde van een werkingsgebied is een set van 1 of meer **geoNaam,** **geoObject** of **geoReferentie** entiteiten[[9]](#footnote-9).

Een **geoNaam** is een waarde uit een te benoemen (geografische) waardenlijst. Een voorbeeld is “Alkmaar” uit een lijst met gemeenten.

Een **geoObject** is een geldig GM\_Object volgens GML 3.2[[10]](#footnote-10). STOP stelt geen verdere eisen aan het soort geoObject of andersoortige inhoudelijke beperkingen. Een toepassing van STOP kan hier wel nadere beperkingen voor definiëren, bijvoorbeeld alleen GM\_Surface.

Een **geoReferentie** is een interne (binnen de publicatie) verwijzing naar een **geoObject**. Deze eigenschap is bedoeld voor serialisatie in XML.

Als aangegeven volgen werkingsgebieden de hiërarchie van de tekst. In een situatie dat er een groot aantal geoObjecten op een bepaald niveau gedefinieerd is als werkingsgebied, dan kan het wenselijk zijn om uit te drukken dat het werkingsgebied op het niveau daaronder bestaat uit “alle direct bovenliggende geoObjecten structuren behalve ...”. Hiervoor wordt de optionele eigenschap “filter”gebruikt, met als waarde “True”. In dat geval vallen de waarden op dit niveau weg uit het resulterende werkingsgebied.

Een implicatie van de opname van werkingsgebied in STOP is dat de XML serialisatie moet kunnen omgaan met GML fragmenten[[11]](#footnote-11). Dit laat onverlet dat een publicatie op basis van STOP geen GML document volgens IMRO 2012 is, maar het is wel denkbaar dat dit via een standaard XSLT transformatie gegenereerd kan worden.

Zowel werkingsgebied als geoNaam, geoObject of geoReferentie hebben optioneel een (binnen de publicatie unieke) identifier. Er worden in STOP geen nadere eisen gesteld aan het formaat van de identifier – dit kan eventueel wel in toepassingen van STOP gebeuren.

Zowel werkingsgebied als geoNaam, geoObject of geoReferentie hebben optioneel een of meer metadata uitspraken.

In bovenstaand model kan een werkingsgebied uit 1 of meer geoObjecten bestaan. Omgekeerd kan een geoObject tot 1 of meer werkingsgebieden behoren. Dit leidt tot een complexe serialisatie in XML, omdat er een keuze gemaakt moet worden waaraan het “echte” geoObject wordt gehangen en van waaruit verwezen wordt via geoReferenties.

De opname van werkingsgebied in STOP heeft ook als gevolg dat er met consolidatie rekening mee gehouden moet worden. In het hoofdstuk consolidatie wordt dit nader besproken.

### Voorstel Geonovum voor Werkingsgebied

Er is een verschil tussen het Werkingsgebied dat op publicatieniveau geldt en werkingsgebieden die in een omgevingsdocument op lager niveau een rol spelen. Zou het niet helpen als we deze twee uit elkaar trekken.

* Het Werkingsgebied in STOP betreft dan het **gehele** (ruimtelijk) gebied waarop een publicatie betrekking heeft.
* Het Werkingsgebied in STOD betreft de deelgebieden waarvoor meer specifieke regels gelden die in een publicatie zijnde een omgevingsdocument beschreven zijn.

Een betere definitie van de tweede is nodig, alsook een onderscheidende naam. *Werkingsdeelgebied*?

Het werkingsdeelgebied moeten we m.i. niet willen definiëren in het Tekstmodel maar in het ‘geo-model’ zoals ik het maar even noem, cq het IMOD. Het werkt dan als volgt:

* Het Werkingsdeelgebied wordt gedefinieerd in het IMOD in UML door Geonovum
* Dit model definieert in ieder geval het Werkingsdeelgebied en haar eigenschappen (waaronder, maar niet uitsluitend, de geometrie)
* Er moet nog vastgesteld worden in hoeverre (eigenschappen van) Onderwerp, en EigenschapWaarde in het geo-model gedefinieerd dienen te worden. Er is een nieuwe versie van IMOD nodig waarin Tekst misschien helemaal niet meer voorkomt (tenzij een relatie vanuit geo naar tekst nodig is).
* Nadat het geo-model is gedefinieerd wordt vanuit de UML een GML Application Schema automatisch gegenereerd (XML schema dat conform de GML standaard is opgesteld).
* Dit XML schema wordt geïmporteerd in het STOD tekstmodel schema.
* Het element Werkingsdeelgebied dat is gedeclareerd in het XML schema van het geo-model wordt in het tekstmodel schema op die plekken in het content model waar het nodig is (iig Artikel), gebruikt.
* Het IMOD XML Schema kan ook op zichzelf worden gebruikt om GML uit te wisselen.

Disclaimer: dit voorstel is door LvdB opgesteld en niet afgestemd met Geonovum collega’s Paul Janssen enWilko Quak vanwege hun afwezigheid imv vakantie.

### Metadata

In het tekstmodel is de lopende tekst de kern. Op diverse plaatsen kan het nodig zijn om informatie toe te voegen die niet zichtbaar is in de lopende tekst[[12]](#footnote-12). Deze informatie is metadata. Iedere metadata **uitspraak** volgt het patroon <subject> <eigenschap> <object>. Dit is conform de [RDF standaard](https://www.w3.org/RDF/) (waarbij voor “predicate” ook “property” gebruikt mag worden.

Het subject bevat een identifier van de entiteit waarover de uitspraak gaat. Bij serialisatie als XML is het subject is veelal impliciet omdat het uit de plaatsing van de metadata container volgt. Alleen wanneer metadata uitspraken fysiek ontkoppeld worden van de publicatie (en daarmee hun context) dan moet het subject expliciet gemaakt worden in de metadata uitspraak.

Eigenschap is bij voorkeur een URI die aangeeft welke relatie tussen subject en object bestaat[[13]](#footnote-13).

Het object is de waarde van een metadata uitspraak. Het betreft een getypeerde waarde. Binnen STOP worden de [XML Schema primitive datatypes](https://www.w3.org/TR/xmlschema-2/#built-in-primitive-datatypes) toegestaan[[14]](#footnote-14). Daarnaast worden ook list en hashmap ondersteund. Een list[[15]](#footnote-15) is een geordende verzameling van 1 of meer waarden[[16]](#footnote-16). Een hashmap is een ongeordende set met sleutel – waarde paren, waarbij de sleutel een string zonder spaties is. De waarde van een list kan zelf weer een list zijn (recursief). Dit geldt ook voor de hashmap.

Er is binnen STOP een **metadata container**. Deze bestaat uit 1 of meer metadata uitspraken. Implicatie van de keuze om metadata te definiëren als container is dat deze niet op het niveau van block of inline geplaatst kan worden. Indien een metadata uitspraak over een block of inline gaat dan moet dit block of deze inline expliciet als object benoemd worden in de metadata uitspraak (via een referentie naar het id).

Vergeleken met voorgangers van STOP is metadata niet langer iets dat alleen op documentniveau speelt. Dit heeft als voordeel dat meer informatie vastgelegd kan worden en dat minder snel specifieke eigenschappen toegevoegd hoeven te worden.

### Identificatie

Alle entiteiten binnen STOP hebben een optionele identifier. In een toepassing kan het gebruik verplicht gesteld worden voor een of meer entiteiten. Ook kunnen in een toepassing eisen worden gesteld aan de vorm van de identifier (bijvoorbeeld een GUID).

Via de metadata container kunnen additionele uitspraken worden gedaan over entiteiten. Voorbeelden zijnde toekenning van een Juriconnect 1.31 verwijzing of een stam-id aan een structuurelement.

### Consolidatie

Consolidatie betreft het doorvoeren van wijzigingsinstructies, resulterende in 1 of meer nieuwe toestanden.

De wijzigingsinstructie wordt in een officiële-publicatie opgenomen in een **wijzig-artikel** entiteit. De inhoud van wijzig-artikel is in STOP zo generiek mogelijk. In toepassingen zal dit nader ingevuld worden.

Wijzigingsinstructies kunnen betrekking hebben op de tekst, de metadata-uitspraken en / of het werkingsgebied.

Een wijzigingsinstructie kan louter wijzigen wat adresseerbaar is. Dit kan via een “id” eigenschap en / of via een (logische) Juriconnect verwijzing (voor tekst).

Er zijn nog veel vragen rond consolidatie en het informatiemodel, zoals bijvoorbeeld:

* wat is de granulariteit van wijzigingsinstructies op werkingsgebied?;
* hoe ziet een wijzigingsinstructie op werkingsgebied eruit?;
* is het wel de bedoeling dat een metadata-uitspraak gewijzigd wordt via een wijzigingsinstructie? Zo nee, hoe dan wel?
* Wat gebeurt er als een geoObject gewijzigd wordt dat door meerdere werkingsgebieden wordt gebruikt?

### Bijlage

In een bijlage kan vrije tekst worden opgenomen volgens een divisie-structuur. De divisie-structuur is een recursieve hiërarchische structuur zonder de semantische betekenis van de eerder genoemde structuurelementen (paragraaf; hoofdstuk; etc.).

De bijlage is tevens bedoeld voor geoObjecten die meervoudig gebruikt worden.

# Toepassing IMOD

## Onderwerp

De entiteit Onderwerp kan gerealiseerd worden door gebruik van de structuurelementen hoofdstuk en / of paragraaf, waarbij eventueel benodigde eigenschappen als metadata worden gerealiseerd. Dit laatste ziet er in XML als volgt uit:

<metadata>  
 <uitspraak betreft="imor:onderwerp">  
 <hashmap>  
 <eigenschap naam="imor:normadressaat">  
 <waarde uit="http://cop.overheid.nl/imod/1/normadressaat">bevoegdGezagGemeente</waarde>   
 </eigenschap>   
 <eigenschap naam="imor:thema">  
 <waarde uit="http://cop.overheid.nl/imod/1/thema">landschap</waarde>  
 </eigenschap>  
 </hashmap>   
 </uitspraak>  
</metadata>

## Werkingsgebied

Werkingsgebied is in IMOD 0.5 een specifieke vorm van geoObject (namelijk GM\_Surface + eigenschap begrenzingIndicatief). In tekstmodel STOP kan een werkingsgebied uit meer dan 1 geoObject bestaan.

### EigenschapWaarde

In IMOD 0.5 is EigenschapWaarde enerzijds (verplicht en herhaalbaar) gekoppeld vanuit Werkingsgebied en anderzijds (optioneel en herhaalbaar) aan Tekstelement.

Als je EigenschapWaarde interpreteert als geannoteerde leesbare tekst, dan is het logischer om de koppeling altijd via structuurelement of artikelonderdeel te laten lopen (en dus indirect aan werkingsgebied). Het is dan nog wel mogelijk om via metadata uitspraken te doen die rechtstreeks over een werkingsgebied of geoObject gaan.

<!-- nu is er in 0.5 een directe relatie tussen eigenschap-waarde (en daarmee Maatvoering) en werkingsgebied; in Boekel IMRO 2012 is de relatie eigenlijk naar bestemmingsvlak. Probleem?>

<!-- alle enkelbestemmingen opgeteld worden het totale gebied; ze mogen dus ook niet overlappen. Dubbelbestemmingen verdwijnen –->

### Tekstpatroon

In IMOD 0.5 is een tekstpatroon een verzameling vaste zinstructuren waarmee zinnen kunnen worden opgebouwd. Een tekstpatroon bestaat uit een of meer TekstPhrase en / of een of meer Invulveld entiteiten. Een Invulveld is dan een deel van een tekstpatroon waar een term of waarde wordt ingevuld, en een TekstPhrase is een deel van een zin wat kan worden getagged.

In STOP kan regel(ing)tekst geplaatst worden in een block (bijvoorbeeld alinea) dat terecht komt in een artikel of een lid (binnen artikel). Dit zijn de standaard (identificeerbare) entiteiten. Binnen een block kan via een inline een bereik worden gedefinieerd. Aan dit bereik kan een identifier worden toegekend, zodat er via de metadata sectie van het artikel uitspraken over gedaan kunnen worden.

Voor de realisatie van STOD in STOP moeten de volgende vragen beantwoord worden:

* zijn er specifieke invullingen van inline nodig voor STOD?
  + In STOP bestaat een generieke inline waar via @type eigen betekenis aan toegekend kan worden. Is dit voldoende?
* in hoeverre is het ook wenselijk om geannoteerde tekst te kunnen definiëren voor tekst buiten structuurelementen (bijvoorbeeld in een nota-toelichting)?

Voor het aanmaken van inlines gelden de beperkingen zoals aangegeven in het tekstmodel STOP.

Conditie wordt gerealiseerd in TekstPhrase

Voorbeeld: “De raad van de gemeente <<IMOD.bevoegdgezag>> gelet op ...”

Subtype van block? Semantisch

Weergaveformule

### Nota-toelichting

Binnen STOD is nota-toelichting nodig. Nog bepalen in hoeverre er toelichting op artikelen (artikelsgewijs) nodig is, in aanvulling op algemene nota-toelichting.

Het is de verwachting dat nota toelichting consolideerbaar moet zijn. Dit veronderstelt identificatie (geen probleem).

Mogelijk scenario’s qua consolidatie zijn:

* + nieuw artikel – dan ook mogelijk nieuwe toelichting gekoppeld aan artikel
  + gewijzigde toelichting zonder wijziging in artikel

1. Omdat een toestand niet in een publicatieblad wordt gepubliceerd, is er hier geen noodzaak voor een publicatieblad entiteit. [↑](#footnote-ref-1)
2. Zie [Aanwijzingen voor de regelgeving, aanwijzing 94](http://wetten.overheid.nl/jci1.3:c:BWBR0005730&hoofdstuk=3&paragraaf=3.5&aanwijzing=94&z=2011-05-11&g=2011-05-11) [↑](#footnote-ref-2)
3. Zie <https://nl.wikipedia.org/wiki/Artikel_(recht>). [↑](#footnote-ref-3)
4. De aanwijzingen voor de regelgeving zijn aanmerkelijk beperkter dan de praktijk: <http://wetten.overheid.nl/jci1.3:c:BWBR0005730&hoofdstuk=3&paragraaf=3.5&aanwijzing=99&z=2011-05-11&g=2011-05-11>. STOP laat meer toe dan 1 alinea, maar kan via business rules tot conformiteit met aanwijzing 97 komen. [↑](#footnote-ref-4)
5. Zie <https://en.wikipedia.org/wiki/CALS_Table_Model> [↑](#footnote-ref-5)
6. Het is bijvoorbeeld niet de bedoeling dat auteurs zelf allerlei vormgeving gaan bedenken in Word. Dit wordt na aanlevering eerst verwijderd, en dan worden de algemene regels voor het documenttype toegepast. [↑](#footnote-ref-6)
7. Zie Cascading Style Sheets Level 2 Revision 1 (CSS 2.1) Specification – <https://www.w3.org/TR/CSS21/>. [↑](#footnote-ref-7)
8. In voorlopers van STOP was het idee van werkingsgebied niet expliciet benoemd. Door in deze gevallen uit te gaan van (het gebied van) bevoegd gezag kan hiermee omgegaan worden. [↑](#footnote-ref-8)
9. Formeel: (geoNaam | geoVlak | geoReferentie)+ [↑](#footnote-ref-9)
10. Bijvoorbeeld Surface; Polygon; CompositeSurface; OrientableSurface (hierop vinden mogelijk nog wijzigingen plaats). Met geoObject wordt de Geometry root bedoeld uit INSPIRE consolidated UML model. [↑](#footnote-ref-10)
11. In de http://www.opengis.net/gml/3.2 namespace. In voorgangers van STOP werd geen gebruik gemaakt van XML namespaces, maar in de STOP XML serialisatie zal minimaal 1 namespace gebruikt moeten worden. [↑](#footnote-ref-11)
12. De grens tussen XML elementen en metadata is soms vaag. Een XML element <b>vet</b> is pas voor de gebruiker leesbaar indien er een vormgevingsinstructie aan gekoppeld wordt. Dit kan ook uitgedrukt worden als XML attribuut via <span style=”font-weight: bold;”>vet</span> of zelfs <span id=”a1b2bc3”>vet</span> waarbij ergens anders een de koppeling wordt gelegd tussen (het id van) de span en het gewenste gedrag. Het laatste geval komt het dichtst bij metadata zoals het vaak wordt gebruikt, maar in de kern zijn het op UML niveau eigenschappen. [↑](#footnote-ref-12)
13. De terugvaloptie is een string. Vanuit ondersteuning legacy biedt STOP beide mogelijkheden maar via business rules kan binnen een bepaalde toepassing gebruik van URI’s als metadata eigenschap verplicht gesteld worden. [↑](#footnote-ref-13)
14. Plus integer. [↑](#footnote-ref-14)
15. in veel programmeertalen ook wel Array genoemd. [↑](#footnote-ref-15)
16. Let op: een lijst kan dus niet leeg zijn. [↑](#footnote-ref-16)